

AD-A054 642

ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND ABERD--ETC F/G 9/2
GENFIT - A GENERALIZED, INTERACTIVE, ARBITRARY FUNCTION-TO-DATA--ETC(U)
APR 78 J T KLOPCIC

UNCLASSIFIED

ARBRL-MR-02832

SBIE-AD-E430 035

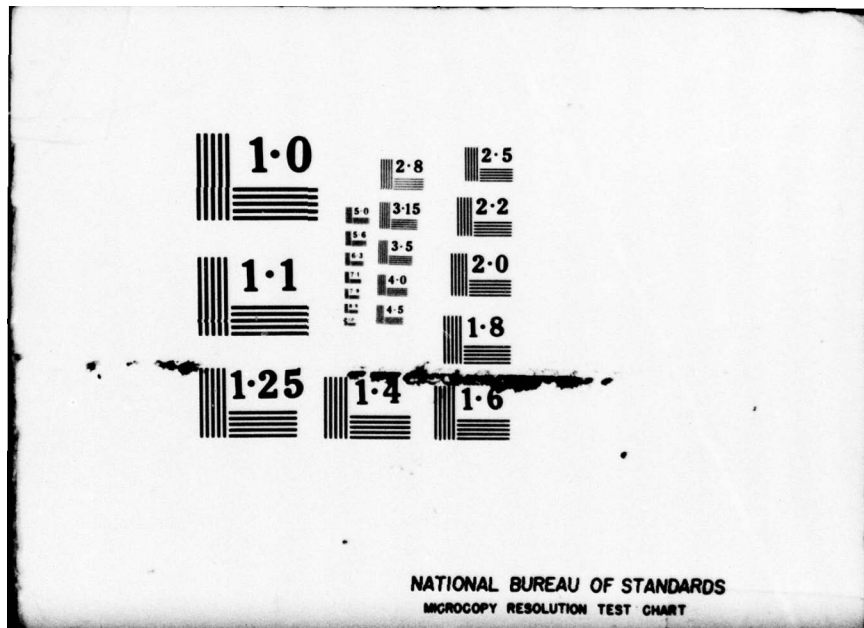
NL

1 OF 1
ADA
054642

12



END
DATE
FILMED
6-78
DDC



AD A 054642

FOR FURTHER TRAN

Handwritten initials and a circled stamp with "b.s."

5/18

AD-E430 035

MEMORANDUM REPORT ARBRL-MR-02832

GENFIT - A GENERALIZED, INTERACTIVE,
ARBITRARY FUNCTION-TO-DATA FITTING ROUTINE

J. Terrence Kloplic

April 1978

DDC
RECEIVED
JUN 7 1978
B

AD No. _____
DDC FILE COPY



US ARMY ARMAMENT RESEARCH AND DEVELOPMENT COMMAND
BALLISTIC RESEARCH LABORATORY
ABERDEEN PROVING GROUND, MARYLAND

Approved for public release; distribution unlimited.

Destroy this report when it is no longer needed.
Do not return it to the originator.

Secondary distribution of this report by originating
or sponsoring activity is prohibited.

Additional copies of this report may be obtained
from the National Technical Information Service,
U.S. Department of Commerce, Springfield, Virginia
22161.

The findings in this report are not to be construed as
an official Department of the Army position, unless
so designated by other authorized documents.

*The use of trade names or manufacturers' names in this report
does not constitute indorsement of any commercial product.*

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

The available output options include an easy-to-use plotting package which yields graphical comparisons between function and data.

The KWIT code is an extremely detailed, multi-variant function to data fitting routine. Because of its high-level computer interface and many details, the program is also easy-to-use. Initial queries are reported, however, the search is sensitive to both guesses only. The data-producing experiment was ill-conditioned in the parameters being determined.

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	5
II. THEORY OF OPERATION	6
III. COMPUTER QUERIES; USER RESPONSES	12
IV. SUBSEQUENT FITS	15
V. CALCOMP PLOTS OF FITS	16
VI. A GENFIT EXAMPLE	16
VII. SUMMARY	24
REFERENCES	27
APPENDIX A - EXAMPLE RESPONSES AND OUTPUT.	29
DISTRIBUTION LIST	39

ACCESSION		
NTIS	NTIS Section	<input checked="" type="checkbox"/>
DOC	DOC Section	<input type="checkbox"/>
UNCLASSIFIED		<input type="checkbox"/>
DISTRIBUTION		
BY		
DISTRIBUTION/ACQUISITION UNIT		
Dist.	Order for	CR/INT
A		

I. INTRODUCTION

In light of the increased emphasis on analysis and modeling, the need to extract the maximum amount of information from difficult-to-augment, scattered data bases, and the growing availability of powerful computers, it is not surprising that data-fitting has become an important facet of Army research and development. Of course, the fitting of standard functions is a well developed field, and "canned" programs are available at any major computing facility. Examples of these are linear regression and Fourier analysis programs. However, the standard functions are linear in at least the fitting parameters, or can be made linear by a simple transformation. For example, a Fourier expansion

$$Y = C_0 + \sum_{k=1}^{\infty} C_k \cdot \cos(\omega t + \phi_k) = C_0 + \sum_{k=1}^{\infty} [A_k \cos(\omega t) + B_k \sin(\omega t)]$$

is linear in the coefficients A, B, and C_0 (but not ω); and the function

$$Y = A_1 \exp(-A_2 \cdot x)$$

can be made linear by taking logarithms

$$\ln Y = C_1 - A_2 x; \quad C_1 \equiv \ln A_1.$$

Unfortunately, the functional forms of the expressions which best reflect an analyst's understanding of data-producing phenomena are often not linear. For such cases, "canned" programs are usually not available. The analyst is then faced with three alternatives: he can attempt to data-fit by hand or by eye; he can develop his own "canned" program; or he can use a "canned", quasi-linear iterative routine. The first alternative is prohibitive if more than a few data points or more than a few parameters are involved. The second alternative is possible, and often preferred, if very large amounts of data are being fit by a well-established, unchanging functional form. However, the development of a fitting program is time consuming, and is thus impractical when less than very large amounts of data are involved, or when the functional form is experimental and is often changed (e.g. in establishing an optimum form).

The third alternative has widespread acceptance, as attested to by the large number of users of the BRL program FNFIT.¹ Since its original development for use in the BRL nuclear dosimetry program, FNFIT has been extensively used in laser vulnerability, laser effects,

¹J.T. Klopco, "FNFIT: An Easy-to-Use, Arbitrary Function-to-Data Fitting Routine (A User's Manual)," BRL-MR-2402. (Aug 74)
(AD #A000655)

radiation shielding, nuclear pulse spectral analysis, and laser beam diagnostics, as well as in many "one-shot" laboratory utility applications. Outside users have included the U.S. Forestry Service, U.S. Army Chemical Systems Laboratory, and private industry.

As originally published, however, the FNFIT routine had shortcomings which limited its usefulness. First, in spite of the title of reference 1, FNFIT proved not to be "easy-to-use" unless the user had initial instruction; and even with experience, a manual was needed to recall data order. Secondly, FNFIT allowed only one independent variable. Hence functions such as the Fourier analysis shown above, in which the dependent variable (Y) is measured as a function of one independent variable (t), could be fit. However, a general expression for the time required for a laser to penetrate a plate, which is a function of such independent variables as plate thickness, laser intensity and power, and wind velocity, could not be fit. Shortcomings also existed in the available outputs, especially the incompleteness of the Calcomp package, and in the available documentation.

In view of the widespread usage of the technique and the improvement in local computer facilities, it was decided to upgrade the FNFIT program. The result is GENFIT, a fully interactive program to fit arbitrary functions of multiple parameters and independent variables to data, and plot the results as a function of any independent variable.

II. THEORY OF OPERATION

The theory of operation of GENFIT is, in part, identical to that of FNFIT (described in reference 1). However, differences do exist. Therefore, a restatement will be presented here.

For statistical reasons,^{2,3,4} the goodness of fit is given by the statistic χ^2 :

$$\chi^2 = \sum_{I=1}^{NDATA} W(I) K^* [Y(I) - F(I)]^2 / (NDATA - NP) \quad (1)$$

where $\sum_{I=1}^{NDATA} W(I) = 1.$

²Applied Optimal Estimation, ed. A. Gelb. M.I.T. Press, Cambridge, MA. (1974)

³P.R. Bevington, Data Reduction and Error Analysis for the Physical Sciences, McGraw - Hill Book Co., N.Y., N.Y. (1969)

⁴Y. Beers, Introduction to the Theory of Error, Addison - Wesley Publishing Co., Reading, MA. (1957)

Here, the fit is to NDATA data points, each point being of the form,

$$X_1, X_2, X_3, \dots, X_{NI}, Y, W$$

where X_1, X_2, \dots = independent variables (e.g. pre-set experimental conditions)

NI = number of independent variables

Y = dependent variable (measured result)

W = weight to be given to each datum

$F(I) = F[X_1(I), X_2(I), \dots; A(1), A(2), \dots, A(NPARAM)]$
is the value of the fitting functions, evaluated at the Ith data point

A(J) = the parameters of the fitting function

NPARAM = number of parameters [A(J)] in F and

NP = NPARAM minus the number of parameters held constant (see NHOLD, below).

It is worthwhile to emphasize that, although the 'X's may have been unknown, measured quantities in the data-taking process, they are not the unknowns in the data fitting. The unknowns are the 'A's, the parameters of the function, whereas the 'X's, 'Y's, and 'W's are known data.

The problem can be succinctly written in matrix notation by writing χ^2 as

$$\chi^2 = (\underline{Y} - \underline{F})^T W (\underline{Y} - \underline{F}) \quad (2)$$

where \underline{Y} and \underline{F} are NDATA-dimensional vectors, each element being one data point, and W a diagonal matrix whose elements are the respective weights of each data point divided by (NDATA-NP).

(The convention adopted here is to define a vector as an N x 1 column matrix; thus the χ^2 defined in equation 2 is the same scalar as defined in equation 1.)

The desired solution of the problem is the NPARAM-dimensional vector \underline{A}^* which minimizes χ^2 . From the calculus, the minimum of a scalar with respect to a vector is achieved when

$$\frac{\partial \chi^2}{\partial \underline{A}} = 0 \quad (3)$$

If F were linear in the A 's, one could write

$$\underline{F} \text{ linear} = \underline{H}\underline{A} + \underline{C} \quad (4)$$

where \underline{C} is an N_{DATA} - dimensional vector independent of \underline{A} , and \underline{H} is an $N_{DATA} \times N_{PARAM}$ dimensional matrix.¹ The solution for the vector \underline{A}^* which satisfies equation 3 is straight forward.²

$$\underline{A}^* \text{ linear} = (\underline{H}^T \underline{W} \underline{H})^{-1} \underline{H}^T \underline{W} \underline{Y} \quad (5)$$

This is the case of "canned" programs discussed previously.

The method used in GENFIT is to force the problem into a linear (equation 4) form. This is accomplished by expanding \underline{F} to first order in a Taylor series about a starting value \underline{A}_0 .

$$\underline{F} = \underline{F}(\underline{A}_0) + \left(\frac{\partial \underline{F}}{\partial \underline{A}} \right) \bigg|_{\underline{A} = \underline{A}_0} \cdot \underline{\Delta A} \quad (6)$$

This is seen to be of the form of equation 4 where

$$\underline{C} = \underline{F}(\underline{A}_0) + \underline{H} \cdot \underline{\Delta A}$$

$$H_{ij} = \partial F_j / \partial A_i \quad (7)$$

and

$$\underline{\Delta A} = \underline{A} - \underline{A}_0$$

Since equation 3 is equivalent to

$$\frac{\partial \chi^2}{\partial \underline{A}} = 0$$

the solution which satisfies equation 3' is given by equation 5 as

$$\underline{\Delta A} = (\underline{H}^T \underline{W} \underline{H})^{-1} \underline{H}^T \underline{W} \underline{Y} \quad (8)$$

where \underline{H} is given by equation 7.

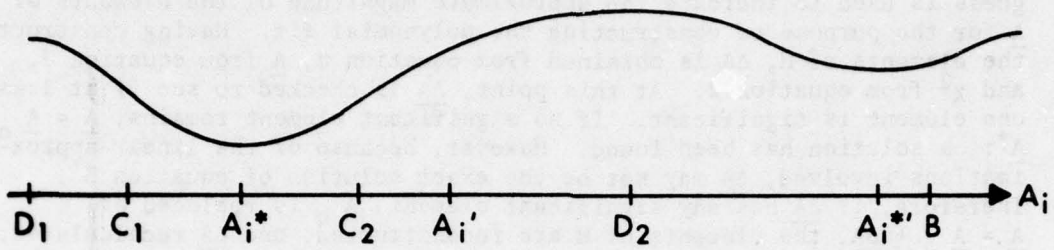
The GENFIT algorithm is straight forward. An initial guess for \underline{A} is input by the user. GENFIT takes this guess as \underline{A}_0 . The elements of

H are evaluated by fitting F in a fifth order polynomial in A at each data point and evaluating the derivative in closed form. The initial guess is used to indicate the approximate magnitude of the elements of A for the purpose of constructing the polynomial fit. Having constructed the elements of H, ΔA is obtained from equation 8, A from equation 7, and χ^2 from equation 2. At this point, ΔA is checked to see if at least one element is significant. If no significant element remains, $A = A_0 = A^*$: a solution has been found. However, because of the linear approximations involved, ΔA may not be the exact solution of equation 3. Therefore, if ΔA has any significant element, A_0 is replaced by $A = A_0 + \Delta A$, the elements of H are reconstructed, and ΔA recalculated. This procedure continues until ΔA becomes insignificant, or the specified maximum number of iterations have been taken.

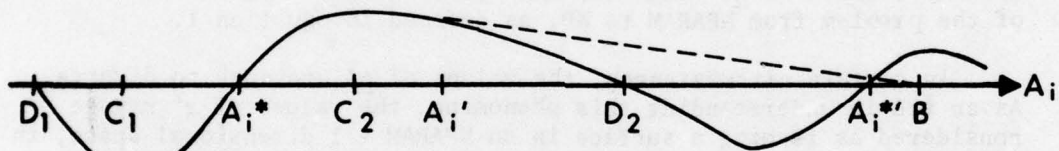
Because of the non-linear nature of the problem, it is possible for successive values of ΔA to diverge or oscillate. In order to control the search, give all elements of A an opportunity to adjust, and prevent elements of A from passing through impossible or divergence-causing values, GENFIT includes the COOLIT and HOLD options. COOLIT allows placing upper limits on the elements of ΔA , and upper and lower limits on the elements of A. HOLD allows the user to specify a fixed value for any element in A. This has the effect of lowering the A-dimensionality of the problem from NPARAM to NP, as defined in equation 1.

In certain circumstances, the values of ΔA are sure to diverge. As an aid in understanding this phenomena, the values of χ^2 may be considered as forming a surface in an NPARAM + 1 dimensional space, in which the abscissae are the elements of A, and the ordinate is the corresponding value of χ^2 . For simplicity, consider an NPARAM-dimensional cross-section through the surface parallel to the A_i axis, as shown in Figure 1a. The desired value of A_i to minimize χ^2 is shown as A_i^* . The χ^2 minimization algorithm, however, does not deal with the χ^2 plot directly, but rather with $\partial\chi^2/\partial A_i$. The cross-section through the corresponding $\partial\chi^2/\partial A_i$ surface is shown in Figure 1b. Effectively, the algorithm finds the slope of the $\partial\chi^2/\partial A_i$ surface at the starting point A_{i0} , and extrapolates to $\partial\chi^2/\partial A_i = 0$. As can be seen in Figure 1, this procedure can only work if the second derivative of χ^2 at the point A_{i0} is positive (χ^2 is concave). In Figure 1, this condition pertains in the interval $C_1 \leq A_i \leq C_2$. However, if the starting guess were the point marked A_i , the extrapolation toward $\partial\chi^2/\partial A_i = 0$ would give the point B in Figure 1b as the next guess for A_{i0} . The final result might be "local minimum" A_i^* , or might be further divergence, depending upon the slope of $\partial\chi^2/\partial A_i$ at B. There is little chance, however, that the point A_i^* would be found.

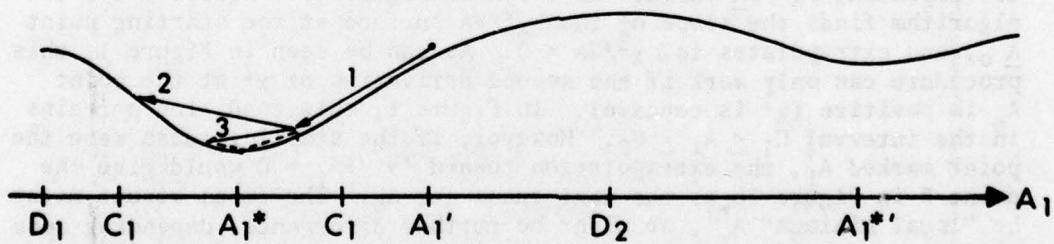
To mollify this effect, GENFIT begins with an optional call to GRID, a modified grid search routine.



a. X^2



b. $\partial X^2 / \partial A_i$



c. X^2

Figure 1. Cross-Sections of X^2 (a and c) and $\partial X^2 / \partial A_i$ (b) vs. A_i
(Dashed Lines Indicate Linear Interpolation)

A grid search is basically a fairly simple, iterative technique. Again, an initial point, \underline{A}_0 , is used. One-by-one, the elements of \underline{A} are selected, the gradient of χ^2 in the selected dimension is calculated, and a series of steps taken toward decreasing χ^2 in the selected dimension.

The modifications to the basic technique which were developed for GRID deal with the determination of step size, and the step logic. The initial step size is based on the parameter magnitudes discussed above. The series of steps is taken according to the following scheme:

1) a step of pre-selected length is taken, where the step direction is found from the gradient of χ^2 at the starting point. If χ^2 increases, the search returns to the starting point, the step size is reduced, and the step retaken. This procedure is repeated until the χ^2 decreases (or an iteration counter reaches its maximum). 2) Upon taking a step that decreases χ^2 , the gradient at the new point is compared to that at the starting point: A) if the two gradients are of the same sign, the minimum has not yet been reached. The step size is increased, and the algorithm returns to step 1. B) If the two gradients are of different sign, the χ^2 minimum lies between the two points. A linear interpolation is done, and the algorithm moves on to the next element of \underline{A} .

Having taken the series of steps in all the (active) elements of \underline{A} , the output subroutine is called and the algorithm is repeated. In GRID, the process continues until no element of \underline{A} is being changed significantly, or until the user-dictated maximum number of steps have been taken. Since only χ^2 reducing steps are taken, GRID exits with its lowest χ^2 , and returns to the main GENFIT routine.

For purposes of illustration, one iteration of a GRID search in one element-dimension is shown in Figure 1c. It is apparent that there is a significantly larger interval (D_1, D_2) within which the starting guess can fall and still result in a convergent search. Since the GRID search operates directly on the χ^2 surface, the surface needs only to slope toward \underline{A}^* ; concavity is not required.

We note here that the GRID search is unique to GENFIT. FNFIT used a STEEPEST GRADIENT search, which, although more efficient in some cases, was too undependable for general use. As a final observation, notice, in Figure 1, that a starting guess to the right of D_2 is more likely to end at A_1^* than A_1^* , regardless of the search technique. For the common case in which the shape of the χ^2 surface is unknown and there is no a-priori knowledge upon which to base the starting guess, the data analyst has no choice but to "map out" the χ^2 surface. Techniques such as the ravine search³ have been developed to help systematize such a mapping; however, experience and knowledge of the system remain valuable quantities.

III. COMPUTER QUERIES; USER RESPONSES

Upon execution (@XQT AMUCK*GENFIT. FUNFIT) the user is welcomed, warned against using file (logical unit) 2, and offered an example expression. Then, the following data is solicited:

1. EXPRESSION - FORTRAN statements to create the fitting expression. Must result in $G = \text{expression}$, where expression involves independent variables $[X(1), X(2), \dots]$ and parameters $[A(1), A(2), \dots]$. End this input with @EOF.
2. HEADING - Up to 10 lines of Hollerith characters to be reproduced at the top of each output page.
3. MODIFIED GRID ITERATIONS - Number of iterations of the GRID search algorithm.
4. CHI-SQ STEPS - Number of iterations of the (main) quasi-linear routine.
5. OUTPUT FORMAT - 0 = output only at end
1 = parameters and χ^2 each iteration
2 = acts like 3 (below) if χ^2 improves, else like 1
3 = option 1 plus full data output on every iteration.
6. OUTPUT ONTO 7? - Unit 7 is a designated alternate print file. Program output can be to any file designated as unit 7 for subsequent examination and multiple copy printing. Respond YES or NO.
7. BINARY FILE FOR CALCOMP - Information for Calcomp plots (optional) is written on a user designated file. Responding 0 indicates no plot desired. The plot option can be turned on and off during a run, but only one file may be used. See section V for Calcomp instructions.
8. NUMBER OF INDEPENDENT VARIABLES - NI as defined in equation 1.
9. OUTPUT VS. WHICH VARIABLE - Orders and lists data points by chosen variable. Also computes derivative of function with respect to chosen variable for output option 4.
10. NUMBER OF PARAMETERS ('A's) - NPARAM as defined below equation 1.
11. PARAMETERS HELD CONSTANT - Although appearing as fitting parameters in the user expression, the user may designate any parameters to be given fixed values for a given fit. Subsequent fits may release or change these parameters (HOLD option as discussed in section II).
- 11A. SUBSCRIPTS - If response to 11 is not 0, the subscripts of the held parameters are solicited.

12. STARTING VALUES (, MAGNITUDES) - Initial values for the parameters, used to start the iterative routines. The magnitudes are used to derive initial step-size GRID, and to generate points for the polynomial approximations in the derivative routines. If the starting value is non-zero, omission of the magnitude results in the starting value being taken as the magnitude. In a well determined problem, the results are fairly insensitive to this entry. See Section VI.
13. SEARCH CONTROLS - Are controls on the parameters desired?
- 13A. CONTROLS - If the answer to 13 is YES, control specifications are solicited in three categories: (1) control of change size during any single iteration, (2) absolute minimum allowed value, and (3) absolute maximum allowed value for specified parameters.
1. Change size - (χ^2 minimization only.) If solution of the quasi-linear equation 3' results in elements of ΔA exceeding specified bounds, the elements are reset to the specified bounds. Special feature - typing ALL bounds changes in all parameters to $0.5 * \text{MAGNITUDE}$.
 2. Minimum - (GRID and χ^2 minimization.) If the new $A (= A + \Delta A)$ has elements that are lower than specified minimum bounds, the elements are reset to the specified bounds. Special feature - Bounds can be set relative to the current value of another parameter. Hence, if A(5) is always to be 0.4 greater than A(4), the response 5, 0.4, A(4) will use the current value of A(4) in assessing the bound on A(5).
 3. Maximum - Works like (2) above on elements of A that exceed specified maxima. Special feature - as in (2), relative bounds may be specified. Note that the lower bound, (2), is always checked first. Hence, if two relative parameters are too close together, the one with the relative minimum will be increased. In order to "share" the change, the relative minimum bound on the larger parameter should be one half of the desired difference, and a relative maximum equal to the whole difference should be impressed on the smaller parameter. Thus, if A(4) is to be 0.5 less than A(5), and it is desired to control both, the proper sequence is to respond 5, 0.25, A(4) to the request for minimum bounds, and 4, 0.5, A(5) to the request for maxima. Then if an iteration resulted in A(4) = 1.6, and A(5) = 1.7. the minimum bound would first change A(5) to $1.6 + 0.25 = 1.85$; and following that, the maximum bound would change A(4) to $1.85 - 0.50 = 1.35$.
- "NOTE ON THE CONTROLS": Since the new values of all parameters (A) in the CHI-SO minimization search are interdependent, the "artificial" alteration of A or ΔA by one of the controls could introduce instabilities in the search. In order to avoid such

instabilities, any parameters which are artificially changed are placed on a temporary hold stack and the dimensionality of the problem reduced for the following iteration as in item 11 above. If a following iteration results in more parameters being artificially changed, those parameters are added to the stack. When an iteration occurs in which no parameter is artificially changed, one parameter is released from the stack on a first-on-first-off basis. In this way, instabilities are damped out by adjustments in the stable parameters. For the convenience of the analyst, any parameter which is held is shown with an * after its value in the output for output options greater than 0.

14. DATA WEIGHTING - Data points can be assigned different weights (the diagonal matrix W in equation 1). Three options are available:

1. Data can be weighted as a power of Y (the dependent variable). Hence, responding -1 will weight each data point inversely as the magnitude of Y (Data points with large Y values are discounted relative to small Y). For many processes in which the relative standard deviation improves as the square root of the number of counts, responding 0.5 properly weights such data.
2. As a corollary, the response 0 weights all data equally.
3. Each datum can include its own relative weight. The DATA INPUT (item 15, below) allows for such input. It should be noted that the action of options (1) and (2) is to ignore any prior weights, and reweight all data. To avoid this action on re-runs, option 3 must be used. Further, all weights are relative. After input, all weights are normalized such that $\sum W(I) = 1$, as specified following equation 1.

15. DATA INPUT - The data is input in free field format, one datum per record. The order is

$$X_1, X_2, X_3, \dots, X_{NI}, Y, W$$

where the X_i correspond to the independent variables in the EXPRESSION (item 1, above), and W, the weight for each datum, is read only if option (3) is selected in item 14 above. The final data record must be:

@EOF

This end-of-file mark initiates execution of the fit.

IV. SUBSEQUENT FITS

After completing a fit, control of the program returns to the user. Since the EXPRESSION has been compiled into the GENFIT program (available as an absolute element called FIT in TPF\$.), it is efficient to repeat the fitting process at this time if desired. GENFIT asks if another fit is requested. If the response is YES, GENFIT allows selective changes of the data input under any of the items of section III except item 1, EXPRESSION. Old data can be left in, augmented or replaced as desired. Reweighting of data is accomplished as described in section III, item 14. Initial values for the parameters are automatically updated to the BEST FIT values of the previous fit; however, the initial magnitudes remain. Both of these can be changed by the user.

V. CALCOMP PLOTS OF FITS

If a non-zero response is given to item 7 in section III, selected data points and a smooth curve of EXPRESSION values encompassing the data points are written on a binary file by GENFIT. The following solicitations are involved in writing the file.

1. INDEPENDENT VARIABLE FOR PLOT ABSCISSA - Since the plot produced is 2-dimensional (Y vs. one X) the subscript of the desired X must be input.
2. CALCOMP SYMBOL - An integer (0 through 127) indicating the system defined plotter symbol used for data points on the plot. If the integer is preceded by a minus sign, the plot program will not create a new set of axes for the current plot. Rather, the current plot will be combined with the preceding plot(s), jointly scaled, and superimposed onto one plot.
3. VALUE FOR OTHER INDEPENDENT VARIABLES - Since the plot corresponds to a slice in the (item 1 - chosen) direction through a multi-dimensional function space, fixed values for the other independent variables must be specified. Thus, in the example of lasers penetrating plates, the dependent variable (penetration time) is plotted versus the chosen independent variable (e.g. laser intensity) while the other independent variables (plate thickness, total power, airflow velocity) are held at the values specified by this input. This input also solicits the ranges within which data points must fall to be included on the plot. Hence, if the laser penetration time is to be plotted versus intensity for 1. cm. plates, the response 1., 0.9, 1.2 would evaluate and plot the expression for a thickness of 1., and would place on the curve those data points for which the thickness fell between 0.9 and 1.2.

The program responds to the above inputs by telling the user how many data points fall within the specified ranges. If at least one

datum qualifies, the user is asked if he wishes to have the plot information written on his previously specified binary file. If response is YES, the information is blocked and written.

Whether the above response is YES or NO, the program asks if more plots are desired. If this response is YES, the program returns to item 1 of this section. Thus, the user can generate several plots of a given fit, overlaying whichever he wishes, selecting various independent variables as abscissae, and fixing different values for the non-selected independent variables.

Upon termination of the fitting program, the plot information binary file is closed. To create a calcomp plot tape from this file, the user must execute the plot program.

@XQT AMUCK*GENFIT. CALCOMP

The CALCOMP program sends the following queries:

- A. INPUT UNIT AND OUTPUT UNIT - The INPUT UNIT is the plot information binary file described above. The OUTPUT UNIT is the Calcomp tape.
- B. IS SCALE LIMIT INPUT DESIRED? - If answered YES, the program will ask for XMIN and XMAX, the desired extent of the plot abscissa, and YMIN and YMAX, the desired extent of the plot ordinate, before each plot.
- C. TITLES FOR GRAPHS - For each new set of axes (see item 2 of this section) a TITLE of less than 73 characters is solicited. If no (further) titles are desired, typing @EOF suppresses this solicitation.
- D. XMIN, XMAX, YMIN, YMAX, OR @EOF - If the answer to B. was YES, the program asks for the desired axes extrema. However, the current plot can be scaled as if B were answered NO by now answering @EOF.

The CALCOMP program ends with a report of how many data sets were drawn on how many plots (sets of axes). The Calcomp tape is closed, and may be plotted in accord with system standard operating procedure.

VI. A GENFIT EXAMPLE

Situation: An unknown, very strong radioactive source is discovered at a storage site being decommissioned. A monitor (monitor A) is set up at a safe distance, and the US Army RADCON team is called in to perform a detailed survey. Two hours later, the RADCON team advance party has already arrived and set up a monitor (monitor B) closer than

monitor A. An hour later, the full team is present, and a series of measurements are taken to within the 500 R/hr. level.

It is hypothesized that the package contains two rather short-lived isotopes. The meter readings are calibrated, and the data fit to a curve of the form

$$\text{RATE} = \left[I_1 * \exp(-\lambda_1 * t) + I_2 * \exp(-\lambda_2 * t) \right] / R^2 \quad (9)$$

The data are given in Table I.

Looking at equation 9, a few facts are easily noted. All the parameters must remain greater than 0. However, even with that restriction, the value of the function could be very sensitive to small changes in λ_1 , and λ_2 . A more subtle difficulty could be the following: the two terms in the numerator, subscripted 1 and 2, appear identical to GENFIT, as they stand. It is possible that the fit will not be able to decide which of the two describes the shorter-lived isotope and will fall into an oscillation between two identical minima in the χ^2 surface.

The COOLIT option provides a mechanism for alleviating the above problems. As listed by GENFIT on the top of the output, non-negative minima and change limits were applied to both A(3) and A(4). Furthermore, A(3) was chosen to be the time constant of the short-lived isotope; therefore, the minimum of A(3) was set relative to A(4). This provides the needed asymmetry between the two terms.

The results of the fitting runs are given in Table II. The different sets of results were the result of changing the initial conditions, data weights, and parameter controls. The first set was run with A(2) held at the value 8.0E3, which was the value used to generate the data. (Note: The data for this example was generated using the values listed in Table III and introducing random round-off errors.)

The second set of results used the results of set #1 as starting values. The lower CHI-SQ resulted from having more independent parameters. The third set of results came from weighting all data evenly, instead of a 10:1 weighting toward the Radcon data in sets 1 and 2.

Figures 2-5 are plots of the fit of answer set #1. However, the plots of the other answer sets are identical in this range. Notice that, although actually exponential, the range of the independent variable time [X(1)] is so short relative to the half-lives that the curve appears linear.

Situation Continues: After the initial RADCON survey, it is decided to transfer the unknown source to a private laboratory for positive identification. However, because of the time required for administrative functions, it is six months to the minute when the

TABLE I. DATA FOR EXAMPLE PROBLEM

MONITOR A (DISTANCE 100m) MONITOR B (DISTANCE 10m)

<u>TIME (hr.)</u>	<u>RATE (cts)</u>	<u>RATE (cts)</u>
0.0	1.80	
0.5	1.79	
1.0	1.79	
1.5	1.78	
2.0	1.78	177.8443
2.5	1.77	177.3127
3.0	1.77	176.7840
3.5	1.76	176.2582
4.0	1.76	175.7352

SURVEY

TIME = 3.0 hr.

<u>DISTANCE (m.)</u>	<u>RATE (cts)</u>
100.	1.7678
50.	7.0714
25.	28.2854
15.	78.5707
12.	122.7667
10.	176.7840
8.	276.2250
7.	360.7837
6.	491.0667
5.5	584.4099

TABLE II. FITTING PARAMETERS FROM TABLE I DATA

DATA SET	χ^2	A(1)	A(2)	A(3)	A(4)
1	2.602-6	1.0+4	8.0+3	1.088-2	1.341-5
2	2.511-6	1.0+4	8.0+3	1.088-2	1.345-5
3	1.661-5	1.0+4	8.0+3	1.088-2	1.353-5
4	2.675-6	5.809+3	1.219+4	1.394-2	2.296-3
5	8.384-5	3.807+3	1.420+4	2.973-2	1.329-4
6	1.675-5	4.414+3	1.359+4	1.536-2	3.022-3
7	3.171-7	6.005+3	1.2+4	1.356-2	2.289-3
8	5.876-7	7.791+3	1.021+4	1.228-2	1.304-3
9	4.311-7	9.206+3	8.794+3	1.092-2	9.278-4

TABLE III. THE GENERATOR SET

A(1)	A(2)	A(3)	A(4)
1.0+4	8.0+3	1.0896-2	7.1209-8

EXAMPLE PROBLEM FOR GENFIT - BOTH MONITORS

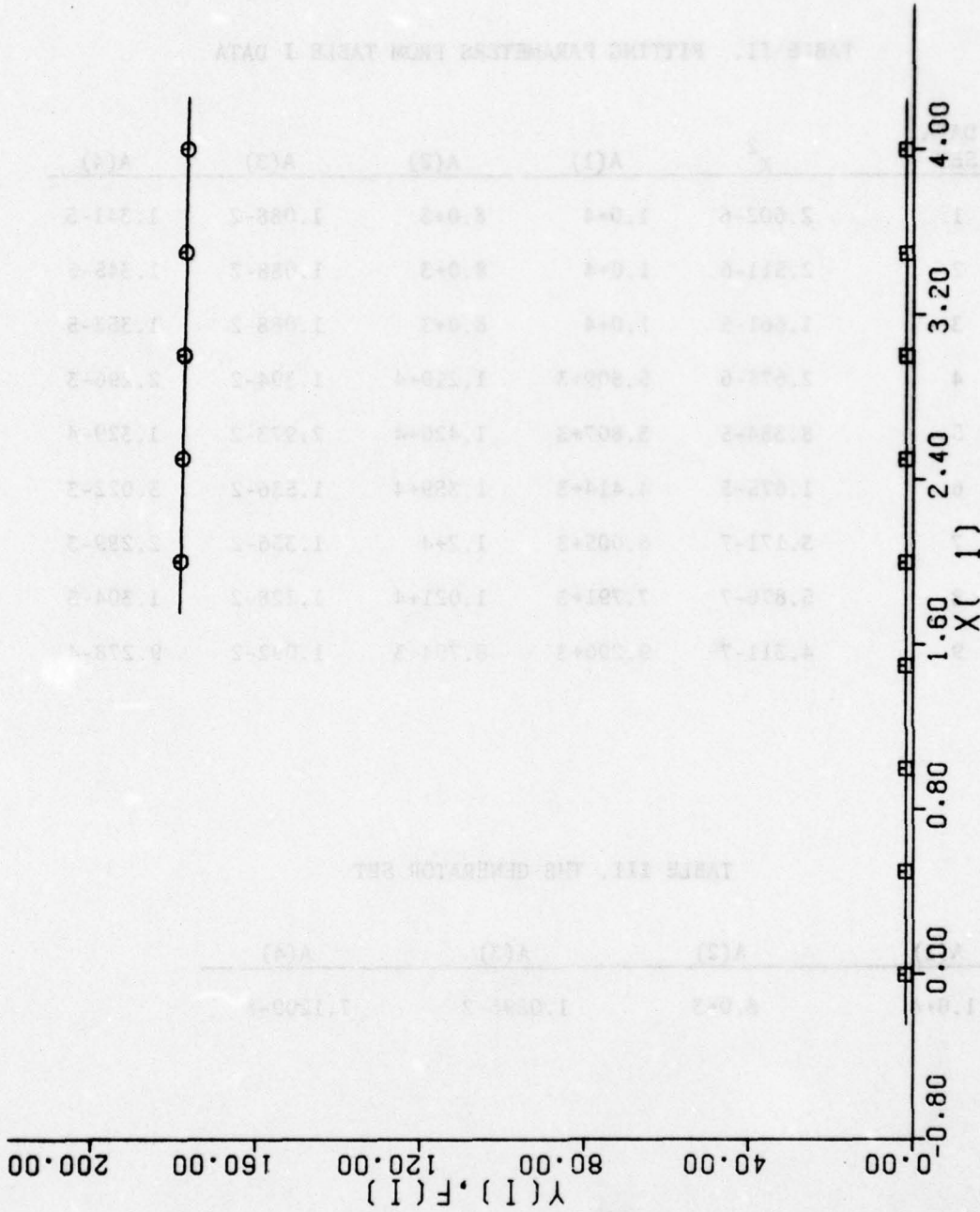


Figure 2. Count Rate Versus Time for Monitor A (squares) and Radcon Monitor (octagons).

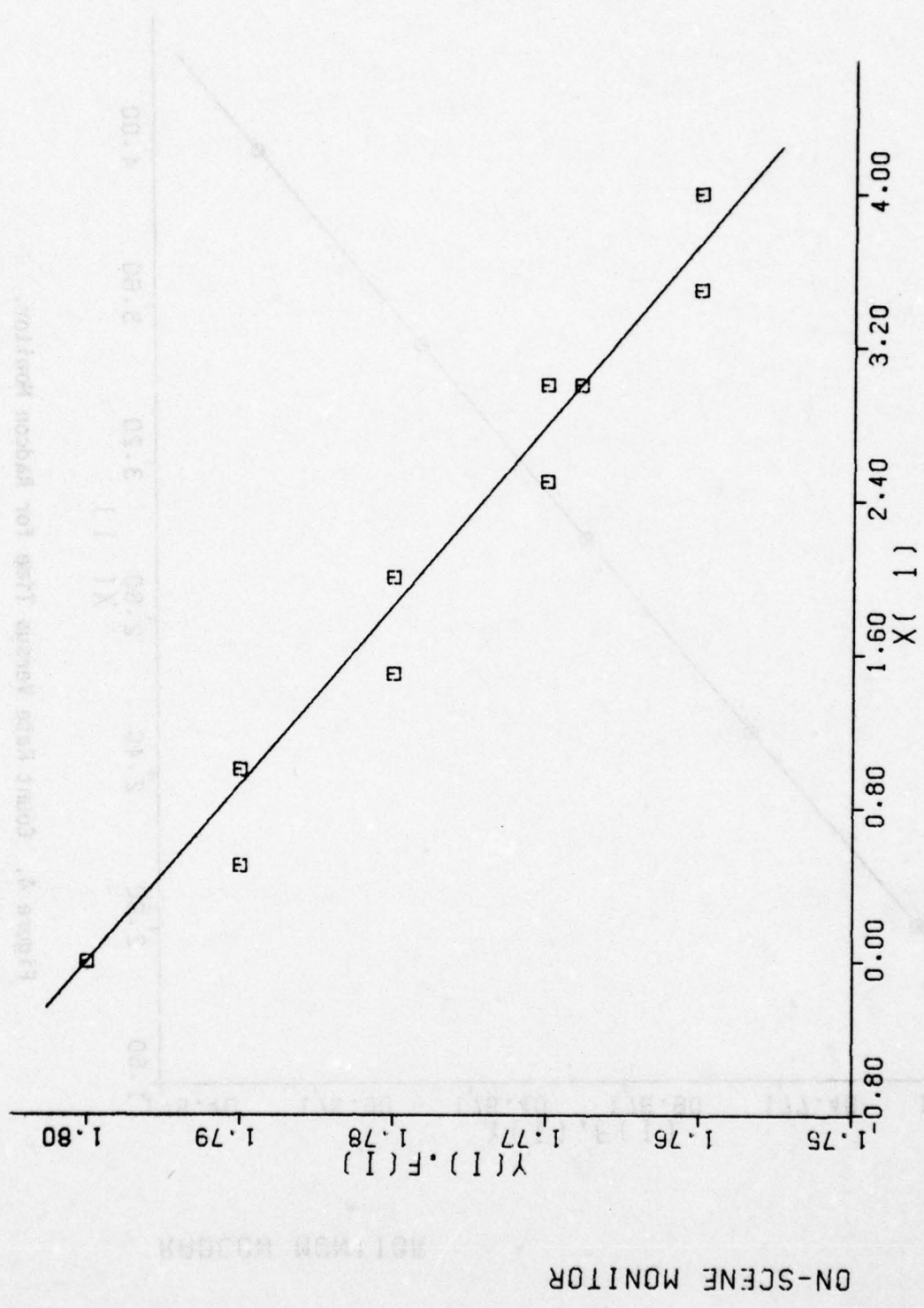


Figure 3. Count Rate Versus Time for Monitor A.

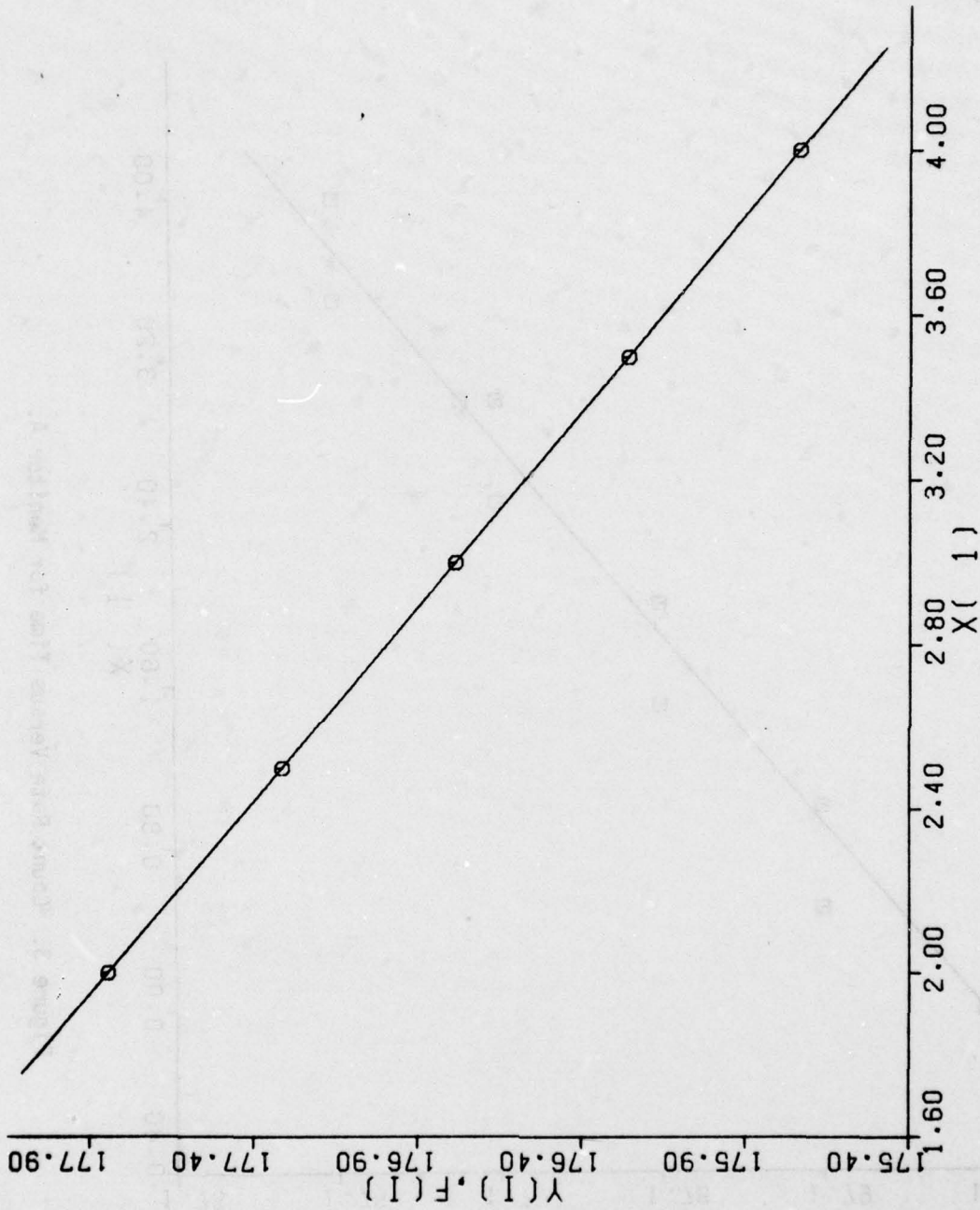


Figure 4. Count Rate Versus Time for Radcon Monitor.

RADCON MONITOR

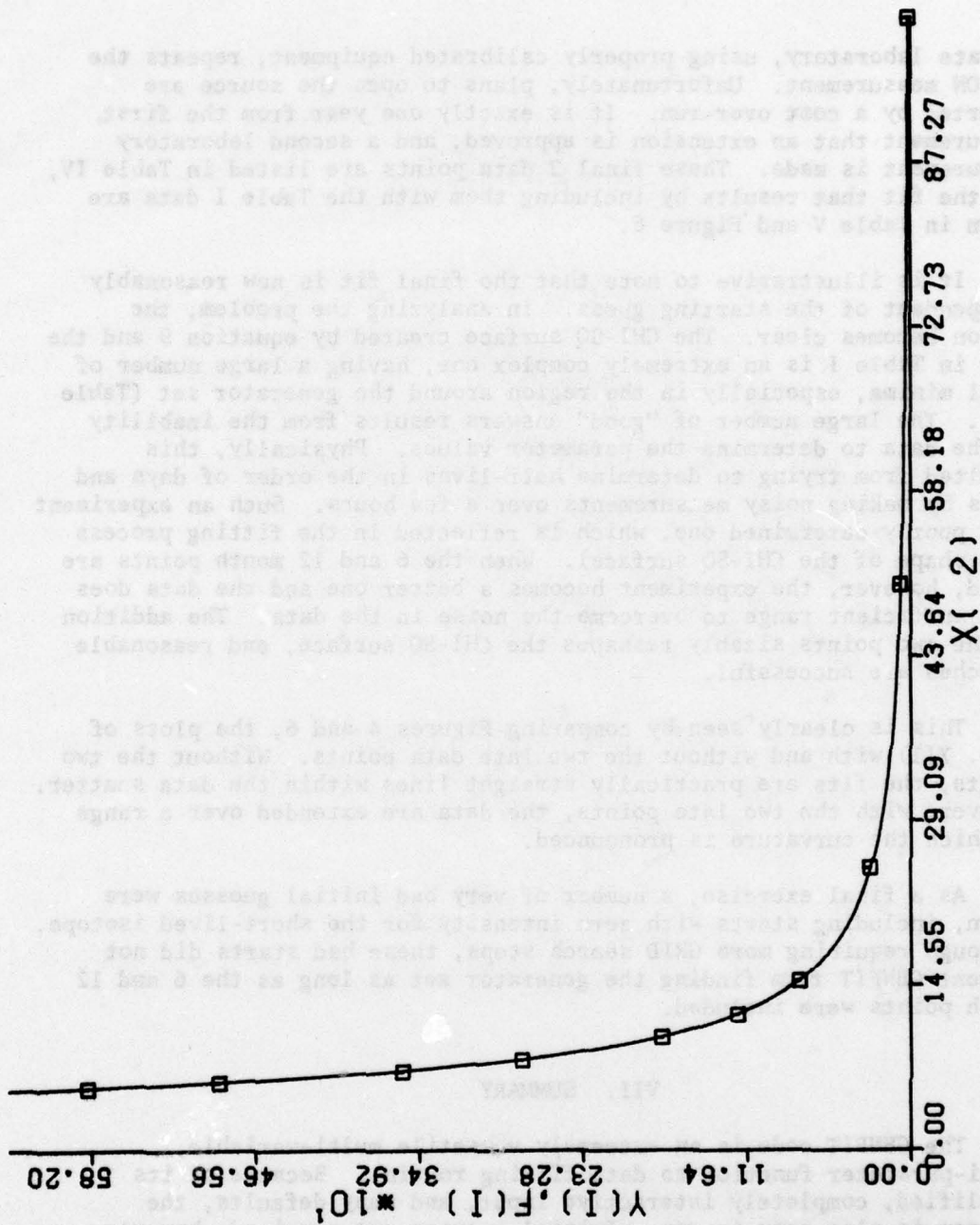


Figure 5. Count Rate Versus Distance At t = 3 hr.

RATE VS. DISTANCE .. TIME = 3 HR.

private laboratory, using properly calibrated equipment, repeats the RADCON measurement. Unfortunately, plans to open the source are thwarted by a cost over-run. It is exactly one year from the first measurement that an extension is approved, and a second laboratory measurement is made. These final 2 data points are listed in Table IV, and the fit that results by including them with the Table I data are shown in Table V and Figure 6.

It is illustrative to note that the final fit is now reasonably independent of the starting guess. In analyzing the problem, the reason becomes clear. The CHI-SQ surface created by equation 9 and the data in Table I is an extremely complex one, having a large number of local minima, especially in the region around the generator set (Table III). The large number of "good" answers results from the inability of the data to determine the parameter values. Physically, this resulted from trying to determine half-lives in the order of days and years by making noisy measurements over a few hours. Such an experiment is a poorly determined one, which is reflected in the fitting process (the shape of the CHI-SQ surface). When the 6 and 12 month points are added, however, the experiment becomes a better one and the data does span sufficient range to overcome the noise in the data. The addition of the two points sizably reshapes the CHI-SQ surface, and reasonable searches are successful.

This is clearly seen by comparing Figures 4 and 6, the plots of Y vs. X(1) with and without the two late data points. Without the two points, the fits are practically straight lines within the data scatter. However, with the two late points, the data are extended over a range in which the curvature is pronounced.

As a final exercise, a number of very bad initial guesses were taken, including starts with zero intensity for the short-lived isotope. Although requiring more GRID search steps, these bad starts did not prevent GENFIT from finding the generator set as long as the 6 and 12 month points were included.

VII. SUMMARY

The GENFIT code is an extremely versatile multi-variable, multi-parameter function to data fitting routine. Because of its simplified, completely interactive input, and many defaults, the program is also easy-to-use. Initial guesses are required; however, the search is sensitive to these guesses only if the data-producing experiment was ill-conditioned to the parameters being determined.

The available output options include an easy-to-use plotting package which yields graphical comparisons between function and data.

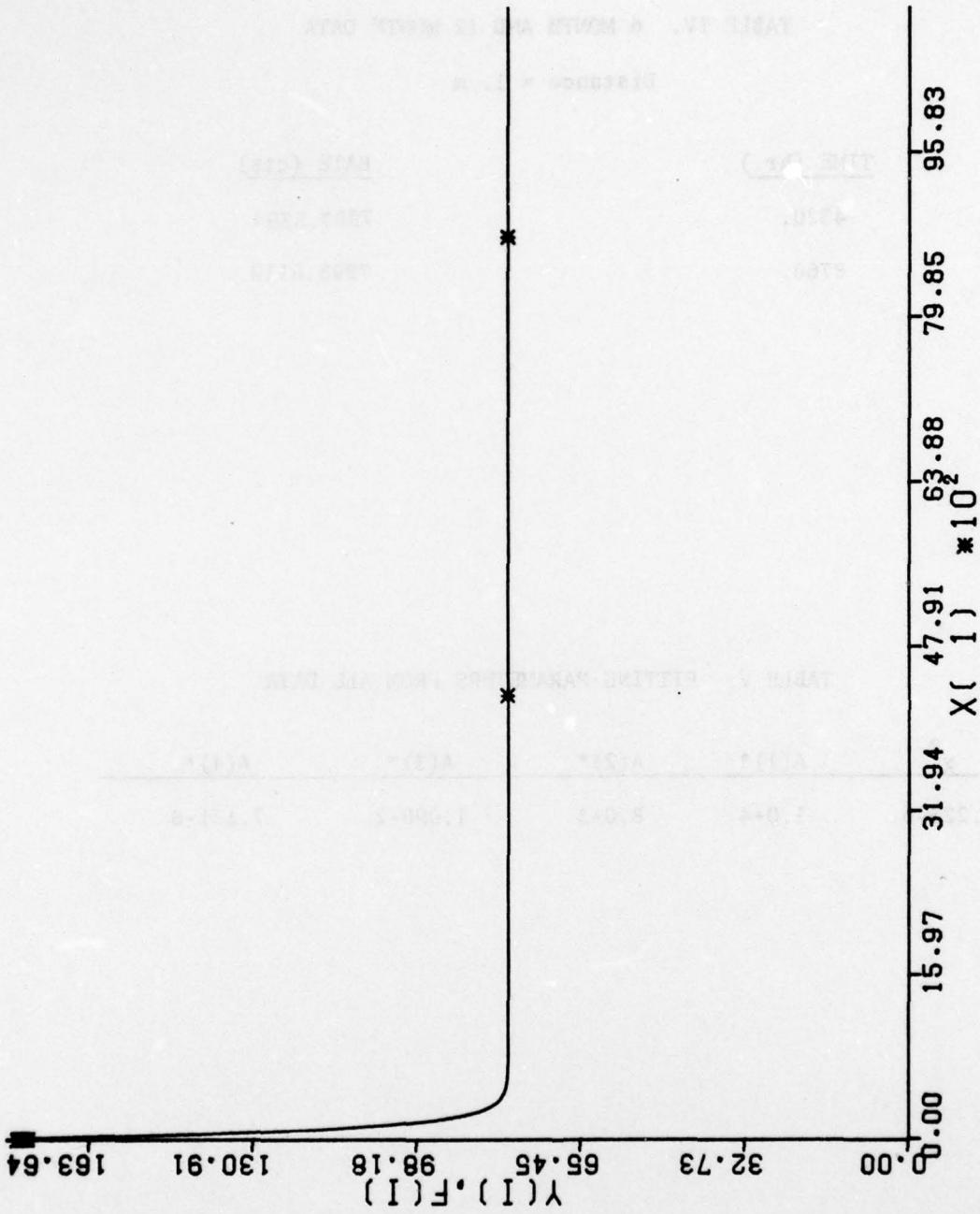
TABLE IV. 6 MONTH AND 12 MONTH DATA

Distance = 1. m

<u>TIME (hr.)</u>	<u>RATE (cts)</u>
4320.	7997.5394
8760.	7995.0112

TABLE V. FITTING PARAMETERS FROM ALL DATA

<u>x^2</u>	<u>A(1)*</u>	<u>A(2)*</u>	<u>A(3)*</u>	<u>A(4)*</u>
1.223-6	1.0+4	8.0+3	1.090-2	7.121-8



RATE VS. TIME .. LATE DATA SCALED TO 10 M.

Figure 6. Count Rate Versus Time for Radcon Monitor Plus
6 Month and 12 Month Points.

REFERENCES

1. J.T. Klopčič, "FNFIT: An Easy-to-Use, Arbitrary Function-to-Data Fitting Routine (A User's Manual)," BRL-MR-2402. (Aug 74)
(AD #A000655)
2. Applied Optimal Estimation, ed. A. Gelb. M.I.T. Press, Cambridge, MA. (1974)
3. P.R. Bevington, Data Reduction and Error Analysis for the Physical Sciences, McGraw - Hill Book Co., N.Y., N.Y. (1969)
4. Y. Beers, Introduction to the Theory of Error, Addison - Wesley Publishing Co., Reading, MA. (1957)

BEST AVAILABLE COPY

```
QASG,UP PRINTFILE.
READY
QUSE 7.,PRINTFILE.
READY
QASG,AZ GENFITDATA.
READY
QXGT AMUCK*GENFIT.FUNFIT
WELCOME TO THE ROUNDEST ( LEAST-SQUARED ) WORLD OF FUNFIT -
*** A 'FITTING' PLACE TO BE**
FUNFIT WILL HELP YOU MAKE UP A AMUCK*GENFIT RUNSTREAM

THE ONLY RULE IS ... DO NOT USE FILE 2 ...
TO PROCEED, TYPE GO
→ GO
FIRST, IN FORTRAN, ENTER YOUR FITTING EXPRESSION
MANY LINES AND LOOPS AND ANYTHING FORTRAN IS O.K.
MUST RESULT WITH: G = FUNCTION( A(K),X(J) )
WHERE 'A'S ARE FITTING PARAMETERS ( K=1,NPARAM )
      'X'S ARE INDEPENDENT VARIABLES
      ( JE1,NUMBER OF INDEPENDENT VARIABLES )
END INPUT WITH GEOF
DO YOU WANT AN EXAMPLE?
→ NO
O.K. NOW ENTER YOUR EXPRESSION AND GEOF.
I'LL COMPILE, MAP, AND THEN CALL YOU
→ G = ( A(1)*EXP(-A(2)*X(1)) + A(3)*EXP(-A(4)*X(1)) ) / X(2)**2
GEOF
FOR 50E3-03/06/78-08:18:23 (7,0)

STORAGE USED: CODE(1) 0001161 DATA(0) 0000161 BLANK COMMON(2) 0000000
END FOR
MAP28R1 RL71-3 03/06/78 08:18:27 (1, )
END MAP
```

Figure A-1. I/O for example run.

*** READY TO FIT ***
 WRITE ANY HEADINGS YOU WISH, UP TO TEN LINES.
 WHEN DONE, TYPE GO
 ANY HEADINGS OR MESSAGES GO HERE
 IT IS OFTEN CONVENIENT TO WRITE THE FITTING EXPRESSION
 $G = (A(1)*EXP(-A(2)*X(1)) + A(3)*EXP(-A(4)*X(1))) / X(2)**2$
 UP TO 10 LINES ARE ALLOWED(A LOT OF GAB!)
 GO
 HOW MANY MODIFIED GRID ITERATIONS? (10 NOMINAL)
 10
 HOW MANY CHI-SQ STEPS?(100 NOMINAL)
 100
 OUTPUT OPTION. 0 IS MINIMAL. 1 IS NOMINAL. 2 IS ALOT
 1
 SEND OUTPUT TO UNIT 7 (ALTERNATE PRINTFILE)?
 YES (I WANT COPIES FOR REPORT. 'NO' SENDS OUTPUT TO TERMINAL)
 TO WRITE A BINARY FILE OF RESULTS FOR CALCOMP,
 TYPE FILE NUMBER. ELSE 0
 15
 TO GET CALCOMP PLOTS, QXGT GENFIT.CALCOMP WHEN DONE FITTING
 HOW MANY INDEPENDENT VARIABLES IN DATA?
 (X(1)=TIME, X(2)=DISTANCE)
 2
 OUTPUT DATA VERSUS WHICH INDEPENDENT VARIABLE (USUALLY 1)
 1
 HOW MANY PARAMETERS ('A'S) ARE IN THE FITTING EXPRESSION?
 4
 HOW MANY PARAMETERS HELD CONSTANT(USUALLY 0)
 0
 NOW YOU'LL HAVE TO THINK. I NEED STARTING GUESSES FOR THE PARAMETERS
 AND, FOR ANY GUESS = 0.0, I NEED AN 'ORDER OF MAGNITUDE' - I.F:
 IS IT BETWEEN -10. AND 10.?(TYPE 0.,10.)
 OR BETWEEN -0.001 AND 0.001?(TYPE 0.,0.001)

Figure A-1 Continued

BEST AVAILABLE COPY

ETC. NOT NEEDED IF GUESS IS NOT 0.0
STARTING GUESS(,MAGNITUDE) FOR A(1)

→ 1.0E4

STARTING GUESS(,MAGNITUDE) FOR A(2)

→ 1.0E-2

STARTING GUESS(,MAGNITUDE) FOR A(3)

→ 1.0E4

STARTING GUESS(,MAGNITUDE) FOR A(4)

→ 1.0E-4

COOLIT OPTION. ARE LIMITS ON PARAMETERS AND/OR PARAMETER CHANGES WANTED?
YES

O.K. THREE CATEGORIES OF SEARCH CONTROLS WILL BE SOLICITED:

(1)PARAMETER CHANGE SIZE, (2)PARAMETER MINIMUM, AND (3)PARAMETER MAXIMUM

(1) TO LIMIT THE SIZE OF CHANGE OF ANY PARAMETER IN ANY ONE ITERATION,
TYPE PARAMETER NUMBER, CHANGE SIZE

TYPE A(1) TO LIMIT A(1), CHANGES TO 0.5*MAGNITUDE
WHEN DONE, TYPE GO

→ ALL

(2) TO PUT A MINIMUM ON ANY PARAMETER, TYPE PARAMETER NUMBER, MINIMUM

TO GIVE A MINIMUM RELATIVE TO PARAMETER K, TYPE PARA.NO., MIN., A(K)

(E.G. 5, 6., A(3) WILL KEEP A(5) .GE. A(3)+6.)

TO CONTINUE, TYPE GO

→ 1,0.

→ 3,0.

→ 4,0.

→ 2,1.0E-3,A(4)

→ GO

(3) TO PUT A MAXIMUM ON ANY PARAMETER, TYPE PARAMETER NUMBER, MAXIMUM

TO GIVE A MAXIMUM RELATIVE TO PARAMETER K, TYPE PARA.NO., MAX., A(K)

(E.G. 5, 6., A(3) WILL KEEP A(5) .LE. A(3)+6.)

Figure A1. Continued

→ TO CONTINUE, TYPE GO

*** FINALLY, THE DATA GOES IN ***
 HOW DO YOU WANT THE DATA WEIGHTED?
 OPTIONS: 1) TO WEIGHT EACH AS A POWER OF Y, ENTER POWER
 F.G. -0.5 WEIGHTS EACH INVERSELY AS SORT(Y)
 (Y REFERS TO THE DEPENDENT VARIABLE)
 2) TO WEIGHT ALL EVENLY, TYPE 0.
 3) TO ENTER WEIGHT WITH EACH DATA POINT, TYPE \$

→ 1. NOW ENTER OR ADD DATA IN THE FORM:
 X(1,1),X(1,2),...X(1,NINDVR),Y(1)
 X(2,1),X(2,2),...X(2,NINDVR),Y(2)
 ...
 THEN ENTER QWOF TO EXECUTE FIT

→ GOOD LUCK!
 QADD,E GENFITDATA.

BEST FIT

6	1.109-06	1.000+04	1.090-02	8.000+03	7.121-08
---	----------	----------	----------	----------	----------

→ *** DO YOU WANT CALCOMP PLOT(S) OF THIS FIT? ***

→ YES

Figure A1. Continued

BEST AVAILABLE COPY

↑ 1 SUBSCRIPT OF INDEPENDENT VARIABLE FOR PLOT ABSCISSA?

↑ 1 CALCOMP SYMBOL (0 - 14). -SYMBOL CAUSES OVERLAY ON PREVIOUS PLOT

↑ 1 NOW NEED FIXED VALUES FOR THE OTHER INDEPENDENT VARIABLES AT WHICH TO EVALUATE AND PLOT THE FIT VERSUS X(1) ALSO NEED RANGE OF VALUES (LOWER AND UPPER LIMIT) WITHIN WHICH OTHER IND. VAR. MUST LIE TO INCLUDE A DATA POINT ON PLOT. VALUE, LOWER, UPPER LIMIT FOR X(2)

↑ 10.,9.,11.

↑ 1 8 DATA POINTS ARE INCLUDED IN PLOT. TO PROCEED, TYPE GO. ELSE NO

↑ GO

↑ DO YOU WANT ANOTHER PLOT OF THIS FIT?

↑ NO

↑ DO YOU WANT TO FIT AGAIN WITH THE SAME EXPRESSION?

↑ NO

NOW, THAT WASN'T SO BAD, WAS IT?

BY THE WAY, YOUR (ABSOLUTE) PROGRAM IS IN TPF\$.FIT.

TO DO A FIT USING THE SAME FITTING EXPRESSION, SAVE IT,

AND JUST XGT IT AT WILL. ALSO, FORTRAN (SOURCE) FOR FITTING EXPRESSION IS IN TPF\$.FCN

BYE

Figure A1. Continued

BEST AVAILABLE COPY

'MAGNITUDE' USED FOR A(1) = 1.000+04
 'MAGNITUDE' USED FOR A(2) = 1.000-02
 'MAGNITUDE' USED FOR A(3) = 1.000+04
 'MAGNITUDE' USED FOR A(4) = 1.000-04

CHANGE LIMITED FOR PARAMETERS ...

A(1) ... 5.000+03
 A(2) ... 5.000-03
 A(3) ... 5.000+03
 A(4) ... 5.000-05

MINIMUM LIMITED FOR PARAMETERS ...

A(1) ... 0.000
 A(3) ... 0.000
 A(4) ... 0.000
 A(2) ... 1.000-03 + A(4)

* = PARAMETER AT LIMIT

MODIFIED GRID SEARCH

ITER	SQ	A(1)	A(2)	A(3)	A(4)
0	1.153+04	1.000+04	1.000-02	1.000+04	1.000-04
1	1.518+01	7.647+03	1.100-03*	1.004+04	3.361-05
2	1.324+01	7.657+03	1.383-03	1.005+04	3.257-05
3	1.300+01	7.659+03	1.526-03	1.005+04	3.231-05
4	1.285+01	7.662+03	1.740-03	1.006+04	3.244-05
5	1.280+01	7.664+03	1.847-03	1.006+04	3.237-05
6	1.278+01	7.666+03	1.900-03	1.005+04	3.242-05
7	1.274+01	7.668+03	2.007-03	1.006+04	3.237-05
8	1.273+01	7.669+03	2.061-03	1.005+04	3.242-05
9	1.271+01	7.672+03	2.141-03	1.005+04	3.237-05
10	1.268+01	7.674+03	2.182-03	1.005+04	3.232-05

USED MAX STEPS, IF NO FINAL FIT, TRY MORE GRID STEPS
 (START WITH LAST GRID VALUES)

Figure A-2. Printout (File 7) from example run.

BEST AVAILABLE COPY

BEST FIT

6 1.109-06 1.000+04 1.090-02 8.000+03 7.121-08

I	X(1)	X(2)	Y	F	Y-F	W	DF/DX(1)
1	0.0000	1.0000+02	1.8000+00	1.8000+00	3.4720-06	1.4682-02	-1.0896-02
2	5.0000-01	1.0000+02	1.7900+00	1.7946+00	-4.5636-03	1.4600-02	-1.0836-02
3	1.0000+00	1.0000+02	1.7900+00	1.7892+00	8.3990-04	1.4600-02	-1.0778-02
4	1.5000+00	1.0000+02	1.7800+00	1.7838+00	-3.7860-03	1.4519-02	-1.0719-02
5	2.0000+00	1.0000+02	1.7800+00	1.7784+00	1.5589-03	1.4519-02	-1.0661-02
6	2.0000+00	1.0000+01	1.7784+02	1.7784+02	1.8883-04	1.4506+00	-1.0661+00
7	2.5000+00	1.0000+02	1.7700+00	1.7731+00	-3.1252-03	1.4437-02	-1.0603-02
8	2.5000+00	1.0000+01	1.7731+02	1.7731+02	1.7738-04	1.4463+00	-1.0603+00
9	3.0000+00	1.0000+02	1.7700+00	1.7678+00	2.1617-03	1.4437-02	-1.0545-02
10	3.0000+00	1.0000+01	1.7678+02	1.7678+02	1.7548-04	1.4420+00	-1.0545+00
11	3.0000+00	1.0000+02	1.7678+00	1.7678+00	-3.8251-05	1.4419-02	-1.0545-02
12	3.0000+00	5.0000+01	7.0714+00	7.0714+00	4.6968-05	5.7679-02	-4.2181-02
13	3.0000+00	2.5000+01	2.8285+01	2.8285+01	-1.2159-05	2.3071-01	-1.6872-01
14	3.0000+00	1.5000+01	7.8571+01	7.8571+01	1.1063-04	6.4087-01	-4.6868-01
15	3.0000+00	1.2000+01	1.2277+02	1.2277+02	1.5450-04	1.0014+00	-7.3230-01
16	3.0000+00	1.0000+01	1.7678+02	1.7678+02	1.7548-04	1.4420+00	-1.0545+00
17	3.0000+00	8.0000+00	2.7622+02	2.7622+02	2.7084-04	2.2531+00	-1.6477+00
18	3.0000+00	7.0000+00	3.6078+02	3.6078+02	3.8147-04	2.9428+00	-2.1521+00
19	3.0000+00	6.0000+00	4.9107+02	4.9107+02	5.1880-04	4.0054+00	-2.9292+00
20	3.0000+00	5.5000+00	5.8441+02	5.8441+02	5.5695-04	4.7668+00	-3.4860+00
21	3.5000+00	1.0000+02	1.7600+00	1.7626+00	-2.5800-03	1.4356-02	-1.0488-02
22	3.5000+00	1.0000+01	1.7626+02	1.7626+02	2.0027-04	1.4377+00	-1.0488+00
23	4.0000+00	1.0000+02	1.7574+00	1.7574+00	2.6497-03	1.4356-02	-1.0431-02
24	4.0000+00	1.0000+01	1.7574+02	1.7574+02	1.7166-04	1.4334+00	-1.0431+00
25	4.3200+03	1.0000+01	7.9975+01	7.9975+01	6.3896-05	6.5233-01	-5.6949-06
26	8.7600+03	1.0000+01	7.9950+01	7.9950+01	6.2943-05	6.5212-01	-5.6931-06

Figure A-2. Printout (File 7) from example run. (Continued)

BEST AVAILABLE COPY

CHISQ MINIMIZATION SEARCH

ITER	SQ	A(1)	A(2)	A(3)	A(4)
0	1.268+01	7.674+03	2.182-03	1.005+04	3.232-05
1	2.384+01	1.013+04	7.182-03*	7.859+03	1.215-06
2	1.262-01	9.894+03	7.182-03*	7.994+03	0.000 *
3	1.163-01	9.894+03	7.182-03*	7.996+03	0.000 *
4	8.190-05	1.000+04	1.091-02	7.996+03	0.000 *
5	5.306-06	1.000+04	1.090-02	8.000+03	7.121-08
6	1.109-06	1.000+04	1.090-02	8.000+03	7.121-08
7	1.154-06	1.000+04	1.090-02	8.000+03	7.121-08
8	1.154-06	1.000+04	1.090-02	8.000+03	7.121-08
9	1.154-06	1.000+04	1.090-02	8.000+03	7.121-08
10	1.154-06	1.000+04	1.090-02	8.000+03	7.121-08
11	1.154-06	1.000+04	1.090-02	8.000+03	7.121-08
12	1.154-06	1.000+04	1.090-02	8.000+03	7.121-08

Figure A-2. Printout (File 7) from example run. (Continued)