LA-8277-MS

Informal Report

C.3

FITPULS

A Code for Obtaining Analytic Fits to Aggregate Fission-Product Decay-Energy Spectra

University of California

LOS ALAMOS SCIENTIFIC LABORATORY
Post Office Box 1663 Los Alamos, New Mexico 87545

This report was not edited by the Technical Information staff.

This work was supported by the US Department of Energy, Division of Energy Reactor Research and Technology.

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Department of Energy, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

UNITED STATES
DEPARTMENT OF ENERGY
CONTRACT W-7405-ENG. 36

LA-8277-MS
Informal Report

UC-34c
Issued: March 1980

FITPULS

A Code for Obtaining Analytic Fits to Aggregate Fission-Product Decay-Energy Spectra

R. J. LaBauve
D. C. George
T. R. England

FITPULS
A CODE FOR OBTAINING ANALYTIC FITS TO AGGREGATE FISSION-PRODUCT DECAY-ENERGY SPECTRA

by

R. J. LaBauve, D. C. George, and T. R. England

ABSTRACT

This report describes the operation and input to the FITPULS code, recently updated to utilize interactive graphics. The code is designed to retrieve data from a library containing aggregate fine-group spectra (150 energy groups) from fission products, collapse the data to few groups (up to 25), and fit the resulting spectra along the cooling time axis with a linear combination of exponential functions. Also given in this report are useful results for aggregate gamma and beta spectra from the decay of fission products released from ^{235}U irradiated with a pulse (10^{-4} s irradiation time) of thermal neutrons. These fits are given in 22 energy groups that are the first 22 groups of the LASL 25-group decay-energy group structure, and the data are expressed both as MeV per fission second and particles per fission second; these pulse functions are readily folded into finite fission histories.

I. INTRODUCTION

Aggregate fission-product beta and gamma spectra in 150 energy groups (the PEFPYD library)[1] have been generated from ENDF/B-IV using the CINDER-10 summation code and other Los Alamos Scientific Laboratory (LASL) processing codes as described in Ref. 1. The FITPULS code is designed to collapse the 150 energy-group fission-product spectra to few groups (up to 25) and fit the resulting spectra along the cooling time axis with a linear combination of exponential functions. An earlier version of FITPULS is described in Ref. 1, including 11-group fits, but recently the code has been updated to improve resulting fits by using interactive graphics. It is the purpose of this report to describe the operation of and input to this version of FITPULS and to provide specific 22-group spectrum fits for the ^{235}U thermal fission products. Familiarity with Ref. 1 may be of help to the reader, particularly in definition of units and utility of the fitted data. The energy boundaries of the 11 groups used in Ref. 1 correspond to boundaries of the 22-group structure used in this report. Reference 1, however, provides fits for several fissioning nuclides.

The PEFPYD library contains data generated for a pulse, that is, for an irradiation time of 10^{-4} s using ENDF/B-IV data. (NOTE: 10^{-4} s is arbitrary but is small compared to fission-product lifetimes, hence adequate for defining a pulse.) Analytic fits to this pulse data are very convenient for design applications, as these functions can be folded with a reactor power history, for example, so that decay spectra for irradiated fuel can be calculated as a function of cooling time. Simple approximations can be added to account for neutron absorption.[1-3] Such fitted functions have been provided for the American Nuclear Society standard for decay-heat power in light water reactors.[4]

II. OPERATION OF FITPULS

The FITPULS code can either take the aggregate of data for various cooling times t after a pulse irradiation and fit it with a linear combination of functions fc(t),

$$fc(t) = \sum_{i=1}^{n} \alpha_i e^{-\lambda_i t} \quad , \qquad \text{(MeV /fission-second)} \qquad (1)$$

where the α_i and λ_i are determined by the code, or the code can take data following one or more finite irradiation times, reduce these to a single equivalent pulse, and obtain the α_i and λ_i for the result. [By an equivalent pulse we mean a function, as in Eq. (1), that will duplicate to great accuracy the initial input values when folded into the finite fission history. The function is not unique even for the case of a single input experiment.] Pulse fits are usually done for calculated data, as from the CINDER-10 code, whereas fits for experimental data generally require the finite irradiation time option. In fact, data from several experiments with different irradiation times can be entered into the code together; the resulting fit will converge to a minimum chi-square, and percent deviations from the use of the pulse will be obtained for each experimental point.

Normally, experimental decay data are for relatively large energy bins or even for a single total summed over all energies, so that rebinning is unnecessary. For the calculated data, however, the 150-group energy bins (in equal widths of 0.05 MeV) need to be reduced to 25 or less by rebinning into wider bins in order to make the fits useful. Routines for doing this are contained in FITPULS in which the bounds of the subset groups need not coincide with those for the 150 fine groups. An option is also available for expressing the regrouped data in either MeV per fission-second or numbers of particles per fission-second.

FITPULS contains three options for obtaining the parameters for the fits to the rebinned decay-energy spectra. In the first option, both the α_i's and λ_i's are obtained by a simple stripping method beginning at the long cooling time end. Semi-log slopes (ΔlogFX/Δt) where (t,FX) are the points to be fitted, and coefficients are calculated for pairs of points beginning with the two extreme cooling times. The contriubitons of the α and λ from a particular pair of points are removed from the remaining points before proceeding to the next pair. If a particular point is within 5% of the value calculated by using the current α and λ, it is considered to be "on the same slope" and is not used in a subsequent α, λ calculation. This stripping process continues automatically in the code until the data points are exhaused, but regions of data requiring negative coefficeints as, for example, regions where the slope reverses, are skipped. These regions are subsequently handled with interactive graphics in another routine in the code as described in the example problem below.

The first fitting option does not require initial guesses for either α_i or λ_i, but input values for the λ_i are required for the second option. The method used in this single-parameter fit, that is, a least-squares fit of the α_i's given adequately chosen λ_i's, is described in Ref. 5. The λ_i's can be taken from previously obtained fits for a similar problem; gamma-decay energy spectra for a different set of yields, for example; or the reciprocals of a number of cooling times taken over the range of interest can be used for input λ_i's. The second option must be used if the data to be fit is for a finite irradiation time, as the first option only works for a pulse. Note that option 2 is not too efficient; and, as a matter of fact, this option should only be used if it is impossible to use either options 1 or 3.

In the third and final option, a non-linear least-squares routine[6] is used. This routine requires fairly good guesses for both the α_i's and λ_i's in order to get good results. These can be obtained either from previous fits with similar shapes or from first running options one or perhaps two. As would be expected, running time for this option is an order of magnitude or more greater than the running time for options one and two, incurring that much greater expense. It is necessary, however, to exercise this option if fits to within a few per cent over large cooling time ranges (over about ten decades) are desired.

III. INPUT AND SAMPLE PROBLEMS

A description of the overlay structure and a listing of the FITPULS code is contained in Appendix A. A description of the FITPULS input specifications is given in Table I. Note that the input description given in Ref. 1 is now obsolete, as also is the description of running a sample problem.

As the first sample problem, consider a multigroup collapse of the PEFPYD data for the aggregate decay-energy gamma spectra of fission products produced by a pulse (10^{-4}s irradiation time) of thermal neutrons on ^{235}U. (MAT = 1261, MF=80, MT=803, see Table II for definition of MAT, MF, and MT numbers in PEFPYD.) The PEFPYD data used in this problem were processed from a CINDER-10 output run for a cooling time range from 0.0 to 10^{10} s in six time steps per decade. We shall specify a cooling time range of 0.1 to 10^9 s for this example and a broad-group structure containing groups 1-22 of the LASL decay energy-group structure shown in Table III.[7] The group structure is truncated at group 22 since the PEFPYD library contains insufficient spectral data above 7 MeV to obtain realistic fits.

In this first example, we choose the units of MeV per fission-second and choose to use the first fitting option for the first pass, which calls for the use of interactive graphics. This is done by setting the input as follows.

```
Input to Program FITPULS

     Card 1       NPUN = 20
                  IRAD = 0
                  NCORS = 0
                  NIAPL = 1

Input to Subroutine CORSBIN

     Card 1A      MAT1 = 1261
                  MF1 = 80
                  MT1 = 803
                  MEVU = 0
```

TABLE I

FITPULS INPUT SPECIFICATIONS

Card No.	Format	Variable	Comment
1	12I6	NPUN	Set NPUN = 7 if rebinned data file wanted; otherwise set to 20.
		IRAD	Set IRAD = 0 for regular pulse fit. Set IRAD = 1 to reduce finite irradiation data to pulse.
		NCORS	Set NCORS = 0 to call subroutine CORSBIN. Set NCORS = 1 for no call, i.e, if input data is not to be rebinned, which is usually the case for fitting experimental data (IRAD = 1).
		NIAPL	Set NIAPL = 1 if first fitting option using interactive graphics is desired. Set NIAPL = 0 if second or third option using file 7 output from previous problem is desired. Otherwise, set NIAPL = -1.
(Input continued in OVRLAY1)			
2	8A10	TITL(I)	80 character title, if TITL(1) = SELECT, subroutine SELECT is called and this input goes here (see SELECT input). If TITL(1) = DO NOT GO, program stops.
3	12I6	IPROB	Problem No. Make negative if fit is to be made in segments. See conditional input below.
		NTOTER	Option to read experimental data. See subroutine RUNTOTS.
		NPUN	Flag for file 7 output. Set equal to 7 if file 7 output is desired, equal to 20 if file 7 output is not desired. In general file 7 output is needed for subsequent runs.
		NSTEP	Flag to call subroutine DHFIT, which calls the routine STEPIT that performs a non-linear least-squares fit. Routine usually not called until a couple of passes are made to get a coarse adjustment of the parameters using more rapid options. Set equal to zero if DHFIT is not desired,

			otherwise set to 1. Also note below option for calling DHFIT by group.
		NFINL	Flag for option to read all parameters for all groups from previous problem. Used when striving for final convergence. Set equal to 1 to activate, otherwise set equal to zero. See conditional input below.
4	6E12.5	DIFLIM	Maximum per cent deviation allowed in STEPIT. Set high on initial passes and tighten up as desired convergence is approached.
		RUNTIM	Running time. Make fraction of second less than execution time to get file 7 for subsequent run.
		TMIN	Minimum cooling time desired. If set to zero, code will choose minimum cooling time available on data file.
		TMAX	Maximum cooling time desired. If set to zero, code will choose maximum cooling time available on data file.
		GXMIN	Minimum allowed value of decay energy. Set so fit is limited to about 15 decades.
5	12I6	KKN(I)	Flags for calling for fits by group. If call for a particular group, say Group IG, is desired, set KKN(IG) = IG. If call is not desired, set KKN(IG) = zero.

FITPULS Conditional Input

(Input for subroutine CORSBIN, after card 1 if NCORS = 0)

1A	6I11	MAT1	MAT-No. of desired fissioning nuclide.
		MF1	MF desired (incident energy type).
		MT1	MT desired (particle/photon data type).
		MEVU	MEVU = 0, units are in MeV/fiss-s. MEVU = 1, units are in particles/fiss-s.
2A	6I11	NE	No. of desired broad groups + 1.
3A	6E11.4	EB(I)	Energy bounds in MeV, including lower and upper bounds. Read low to high energy.

(Input for subroutine SELECT, after card 2 if TITL(1) = SELECT)

1B	12I6	ITS,ITP	No. of time steps desired, indexes of desired time steps. SELECT used if one wishes to fit a subset of particular data file, and this input follows the title card.

(Input for subroutine RUNTOTS, after card 2 if NTOTER = 1)

The read statements in this subroutine should be changed to conform with format in which data is received. See subroutine.

If NIAPL = 0 or NFIN = 1, file 7 output from a previous problem must be attached after card 5. This file, in an ENDF-like format, contains alphas and lambdas for all groups.

(Input for subroutine TRMSEE, after file 7 output and for each group IG where KKN(IG) = -IG)

1C	12I6	MLT	Number of parameters to be changed.
2C	I6	L	Time step number of parameters to be changed.
	E12.5	ALF(K,L)	New value of alpha
	E12.5	ALAM(K,L)	New value of lambda
3C	12I6	LT	No. of terms to be removed.
		LTM(I)	Term nos. of terms removed.

IPROB set negative allows the data for the groups to be fitted in several segments, and card 6 is read after card 5.

6	12I6	NSEG	No. of segments plus 1.
		NS(I)	Breakpoints of segments.

(Input for subroutine PULSFIT, after card 5 or 6 if NIAPL = -1.

1D	12I6	LWT	Weight function desired in single parameter fit. If LWT = 0, weight function = 1. If LWT = 2, weight function = 1/FX. If LWT = 2, weight function = $1/FX^2$. If LWT = 3, weight function = $1/FX^{1.5}$. (Here, FX = MeV/fiss-s values to be fitted.)
		KTRM	No. of lambdas to be read in.
		IPRT	Print flag; if IPRT = 1, print A matrix. If IPRT = 0, no print.

2D	12I6	IWANT	Select lambdas wanted by position number. If all are to be retained, as in a first pass, set equal to zero.
		KCAL(L)	Position nos. of lambdas to be kept. Do not enter if IWANT = 0.

TABLE II

DEFINITION OF MAT, MF, MT NUMBERS USED IN PEFPYD

MAT: Mat-No. of target nucleus, same as in ENDF/B.

MF: File No., used to identify energy type of incident neutron, defined as follows:

MF=80 - fission induced by thermal neutrons.
MF=81 - fission induced by fast neutrons.
MF=82 - fission induced by high-energy (14-MeV) neutrons.

MT: Section number used to describe data contents of the sections. MT numbers are as follows:

MT=801 - delayed energy/fission for $\beta^- + \gamma$ summed over all fission products.

MT=802 - delayed energy/fission for β^- summed over all fission products.

MT=803 - delayed energy/fission for γ summed over all fission products.

MT=811 - delayed energy/fission for $\beta^- + \gamma$ summed over all gaseous fission products (halogens plus noble gases).

MT=812 - delayed energy/fission for β^- summed over all gaseous fission products.

MT=813 - delayed energy/fission for γ summed over all gaseous fission products.

MT=821 - delayed energy/fission for $\beta^- + \gamma$ summed over the noble gas fission products.

MT=822 - delayed energy/fission for β^- summed over noble gas fission products.

MT=823 - delayed energy/fission for γ summed over noble gas fission products.

MT=831 - delayed energy/fission for $\beta^- + \gamma$ summed over halogen fission products.

MT=832 - delayed energy/fission for β^- summed over halogen fission products.

MT=833 - delayed energy/fission for γ summed over halogen fission products.

TABLE III

LASL GROUP STRUCTURE FOR DECAY ENERGY

Group No.	E-Low (MeV)	E-High (MeV)	Delta-E (MeV)
1	0.00000	0.01000	0.01000
2	0.01000	0.10000	0.09000
3	0.10000	0.30000	0.20000
4	0.30000	0.40000	0.10000
5	0.40000	0.50000	0.10000
6	0.50000	0.70000	0.20000
7	0.70000	0.90000	0.20000
8	0.90000	1.00000	0.10000
9	1.00000	1.12500	0.12500
10	1.12500	1.33000	0.20500
11	1.33000	1.50000	0.17000
12	1.50000	1.66000	0.16000
13	1.66000	1.87500	0.21500
14	1.87500	2.00000	0.12500
15	2.00000	2.33300	0.33300
16	2.33300	2.66600	0.33300
17	2.66600	3.00000	0.33400
18	3.00000	3.50000	0.50000
19	3.50000	4.00000	0.50000
20	4.00000	5.00000	1.00000
21	5.00000	6.00000	1.00000
22	6.00000	7.00000	1.00000
23	7.00000	8.00000	1.00000
24	8.00000	9.00000	1.00000
25	9.00000	10.00000	1.00000

Card 2A	NE = 23
Card 3A	(Group bounds from Table III)

Input to OVLAY1

Card 2	Title Card

Card 3	IPROB = 1
	NTOTER = 0
	NPUN = 7
	NSTEP = 0
	NFINL = 0

Card 4	DIFLIM = 5.0 (used in second pass)
	RUNTIM = 295.0 (used in second pass)
	TMIN = 0.1
	TMAX = 1,0E+9
	GXMAX = 1.0E-21

Card 5	KKN(K) = group numbers, 1-22

If the first fitting option is specified as in this example (NIAPL=1), the code must be run from a graphics terminal so that displayed intermediate results can be viewed and acted upon. Although all groups are treated interactively, for purposes of illustration we will discuss the interactive graphics operation for only two groups, namely 12 and 16.

First, consider the fitting for group 12. Figure 1 shows the result of a fit in which only positive coefficients are allowed, which can be done by not entering data points requested from interactive graphics. In the figure, six point per decade data from the PEFPYD library are represented by asterisks; the curve computed from the calculated parameters is represented by a solid line. Note that the fit overshoots in the region 6×10^4 to 4×10^5 s cooling time. As is evident from the figure, negative coefficients are needed in this region to correctly represent the reverse in slope. Table IV, taken from the code output File 20, shows the actual numerical values plotted in Fig. 1, and in addition, the per-cent difference between the computed and original values. As can be seen from this table, the difference exceeds 100% at a cooling time of 1×10^5 s. Note also in Table IV that the first three points overshoot by more than 10%. Parameters calculated using the first fitting option but bypassing the interactive graphics feature are the first 12 in Table V under the heading "Option I" values.

The first figure that appears on the graphics terminal screen for this group (12) is reproduced as Fig. 2. This is for the first "region" that contains at least three contiguous points for which differences between computed and original values exceed 10%, and these differences are plotted on a log scale against the representative cooling times on a linear scale. A vector is drawn between the last two points to aid the eye in selecting those points to be used in computing the parameters. Note in Fig. 2 that points 1 and 3 were selected. The code then computes the slope [$\Delta(\log FX)/\Delta t$] between the selected points (1 and 3), and using this slope next computes the coefficients for the selected points (1,3) and each point in between (2). Finally, an average coefficient is calculated for this set by simply averaging the coefficients. The resulting α and λ are number 13 in Table V.

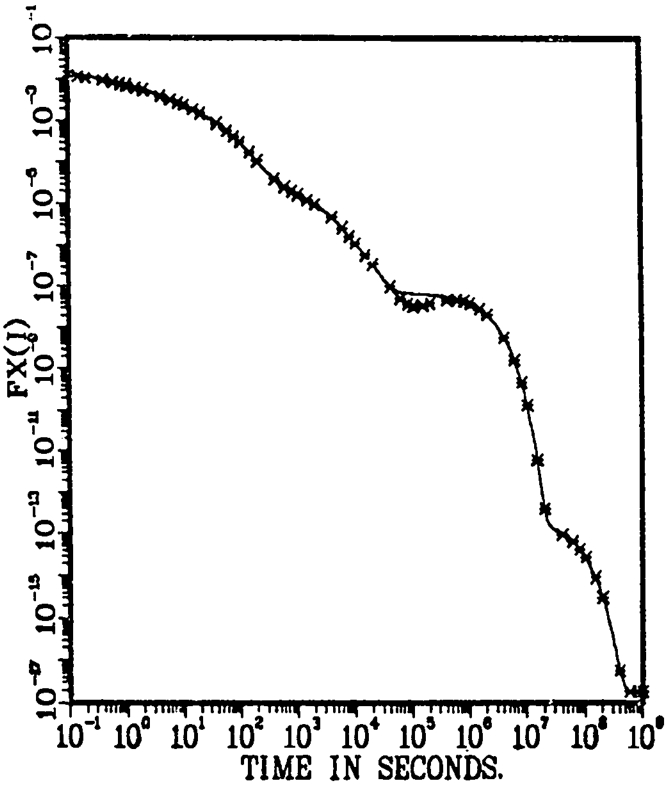
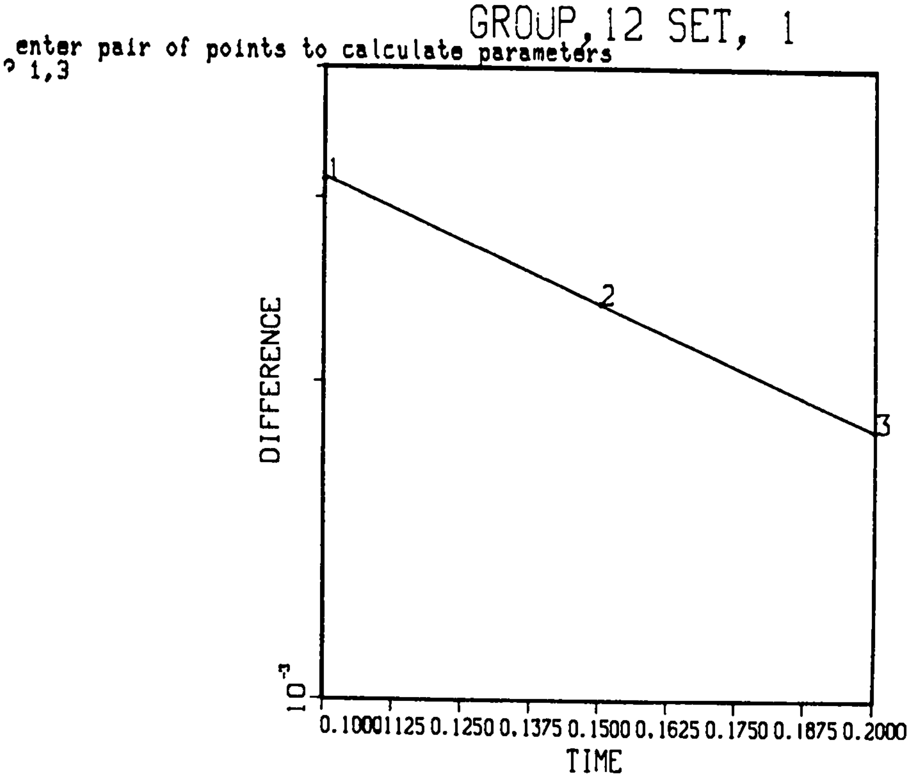


Fig. 1.
Fit without interactively generated parameters for group 12.



```
enter pair of points to calculate parameters
? 1,3
```

Fig. 2.
First display for group 12.

TABLE IV

GROUP 12 RESULTS WITH INTERACTIVE GRAPHICS BYPASSED

Step No.	Time	Original Value	Computed Value	Per cent Difference
1	1.00000E-01	1.18510E-02	1.49678E-02	-2.63002E+01
2	1.50000E-01	1.13322E-02	1.37127E-02	-2.10069E+01
3	2.00000E-01	1.08687E-02	1.26723E-02	-1.65949E+01
4	4.00000E-01	9.43458E-03	9.96329E-03	-5.60400E+00
5	6.00000E-01	8.45178E-03	8.55738E-03	-1.24948E+00
6	8.00000E-01	7.73742E-03	7.73742E-03	0.
7	1.00000E+00	7.18978E-03	7.18978E-03	0.
8	1.50000E+00	6.22239E-03	6.28343E-03	-9.80949E-01
9	2.00000E+00	5.55506E-03	5.63502E-03	-1.43944E+00
10	4.00000E+00	4.02138E-03	4.04420E-03	-5.67343E-01
11	6.00000E+00	3.22970E-03	3.22970E-03	-3.93727E-09
12	8.00000E+00	2.73656E-03	2.73656E-03	-5.07125E-13
13	1.00000E+01	2.39237E-03	2.39485E-03	-1.03424E-01
14	1.50000E+01	1.84014E-03	1.84179E-03	-8.93647E-02
15	2.00000E+01	1.49834E-03	1.49846E-03	-7.90612E-03
16	4.00000E+01	8.43702E-04	8.46443E-04	-3.24903E-01
17	6.00000E+01	5.63688E-04	5.64056E-04	-6.52599E-02
18	8.00000E+01	4.04582E-04	4.04602E-04	-4.90849E-03
19	1.00000E+02	3.03230E-04	3.04000E-04	-2.53772E-01
20	1.50000E+02	1.67961E-04	1.68900E-04	-5.58601E-01
21	2.00000E+02	1.06700E-04	1.06783E-04	-7.76052E-02
22	4.00000E+02	3.86301E-05	3.92374E-05	-1.57219E+00
23	6.00000E+02	2.47972E-05	2.50615E-05	-1.06582E+00
24	8.00000E+02	1.91685E-05	1.91705E-05	-1.04180E-02
25	1.00000E+03	1.60403E-05	1.60404E-05	-4.68689E-04
26	1.50000E+03	1.18872E-05	1.19483E-05	-5.14086E-01
27	2.00000E+03	9.51152E-06	9.51636E-06	-5.09075E-02
28	4.00000E+03	4.62256E-06	4.57756E-06	9.73388E-01
29	6.00000E+03	2.57195E-06	2.56523E-06	2.61617E-01
30	8.00000E+03	1.61457E-06	1.61471E-06	-8.79782E-03
31	1.00000E+04	1.11886E-06	1.11887E-06	-7.91176E-04
32	1.50000E+04	5.72884E-07	5.76982E-07	-7.15375E-01
33	2.00000E+04	3.46863E-07	3.49321E-07	-7.08610E-01
34	4.00000E+04	9.85013E-08	9.85017E-08	-3.75236E-04
35	6.00000E+04	5.10354E-08	7.12370E-08	-3.95835E+01
36	8.00000E+04	3.71013E-08	6.75729E-08	-8.21309E+01
37	1.00000E+05	3.30575E-08	6.64363E-08	-1.00972E+02a
38	1.50000E+05	3.42913E-08	6.43559E-08	-8.76739E+01
39	2.00000E+05	3.85599E-08	6.23740E-08	-6.17589E+01
40	4.00000E+05	4.80559E-08	5.50389E-08	-1.45310E+01
41	6.00000E+05	4.75911E-08	4.85664E-08	-2.04917E+00
42	8.00000E+05	4.37704E-08	4.28550E-08	2.09144E+00
43	1.00000E+06	3.91830E-08	3.78153E-08	3.49055E+00
44	1.50000E+06	2.87345E-08	2.76587E-08	3.74380E+00
45	2.00000E+06	2.09018E-08	2.02301E-08	3.21381E+00
46	4.00000E+06	5.90419E-09	5.78983E-09	1.93694E+00
47	6.00000E+06	1.67887E-09	1.65715E-09	1.29384E+00
48	8.00000E+06	4.78413E-10	4.74402E-10	8.38398E-01
49	1.00000E+07	1.36585E-10	1.35904E-10	4.98015E-01
50	1.50000E+07	6.11894E-12	6.11894E-12	-1.68979E-12
51	2.00000E+07	4.15253E-13	4.15253E-13	-3.89061E-12
52	4.00000E+07	9.67216E-14	1.00144E-13	-3.53809E+00
53	6.00000E+07	6.55882E-14	6.48295E-14	1.15672E+00
54	8.00000E+07	4.28427E-14	4.19710E-14	2.03464E+00
55	1.00000E+08	2.75448E-14	2.71745E-14	1.34410E+00
56	1.50000E+08	9.19704E-15	9.17292E-15	2.62293E-01
57	2.00000E+08	3.10432E-15	3.10432E-15	-4.06587E-13
58	4.00000E+08	5.79643E-17	5.79643E-17	0.
59	6.00000E+08	1.85988E-17	1.86113E-17	-6.73276E-02
60	8.00000E+08	1.80882E-17	1.80948E-17	-3.67576E-02
61	1.00000E+09	1.80799E-17	1.80800E-17	-4.74956E-04

a - maximum difference

TABLE V

EXAMPLE 1: FIT TO SIX PT/DECADE DATA

	Parameters from Option I		Parameters from Option III	
	Alpha	Lambda	Alpha	Lambda
1	1.8122E-17	2.3004E-12	1.8116E-17	2.2465E-12
2	2.3896E-13	2.1747E-08	2.3935E-13	2.1747E-08
3	7.0687E-08	6.2555E-07	7.1024E-08	6.2555E-07
4	2.5966E-06	1.1189E-04	2.0730E-06	1.1517E-04
5	1.6345E-05	4.4012E-04	1.6952E-05	4.3706E-04
6	9.4182E-06	1.3878E-03	8.3936E-06	1.3878E-03
7	1.2285E-04	5.0707E-03	1.2247E-04	5.0707E-03
8	9.9422E-04	1.6398E-02	9.9433E-04	1.6398E-02
9	1.3648E-03	4.8550E-02	1.3623E-03	4.8550E-02
10	2.3382E-03	1.4595E-01	2.3394E-03	1.4595E-01
11	4.3989E-03	5.2611E-01	4.4170E-03	5.3718E-01
12	9.0820E-03	4.1653E+00	9.1106E-03	4.1350E+00
13	-5.3861E-03	5.4701E+00	-5.4596E-03	5.5756E+00
14	-6.0494E-08	4.6613E-06	-6.0839E-08	4.7414E-06
15	3.0316E-07	4.1238E-05	3.0316E-07	4.1238E-05

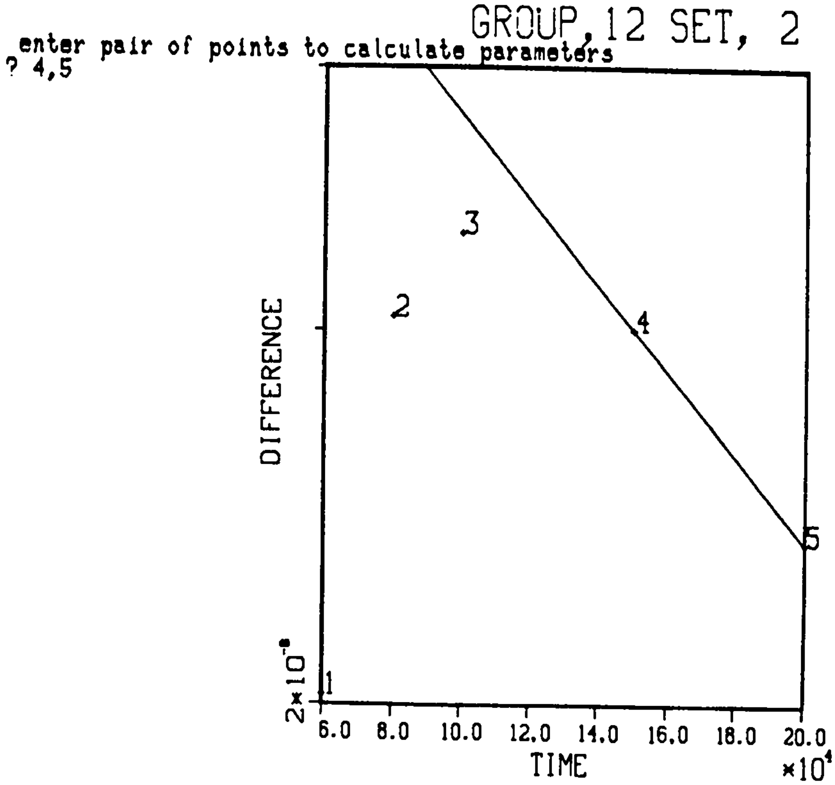
Since the first region contained only three points, the next picture displayed on the graphics terminal for group 12 is for "region" or "set" number 2, which is shown in Fig. 3. This set contains five points from 6 x 10^4 to 2 x 10^5 s cooling time. Note that points 4 and 5 were selected for the calculation, and the resulting α and λ are numbered 14 in Table V under Option I values.

In the next picture displayed, the remaining three points from set 2 are plotted after removal of the contributions from the α and λ computed from points 4 and 5. This display is reproduced as Fig. 4. Although not evident from the figure, the negative coefficient calculated from points 4 and 5 will cause the fit to undershoot, so the coefficient calculated from points selected in Fig. 4 will be positive. Note that points 1 and 2 were selected, and the resulting parameters are numbered 15 in Table V under Option I values.

When in the interactive graphics mode, the code will automatically display the results of the fit compared with the original input data. This graphical comparison is shown in Fig. 5, and numerical values as well as per-cent differences are given in Table VI. Note that the maximum difference has been reduced from 101 to 23%, and that it now occurs at 2 x 10^4 s cooling time. These results are accurate enough to proceed to the second pass in which the third option, that for a non-linear least-squares fit, is specified.

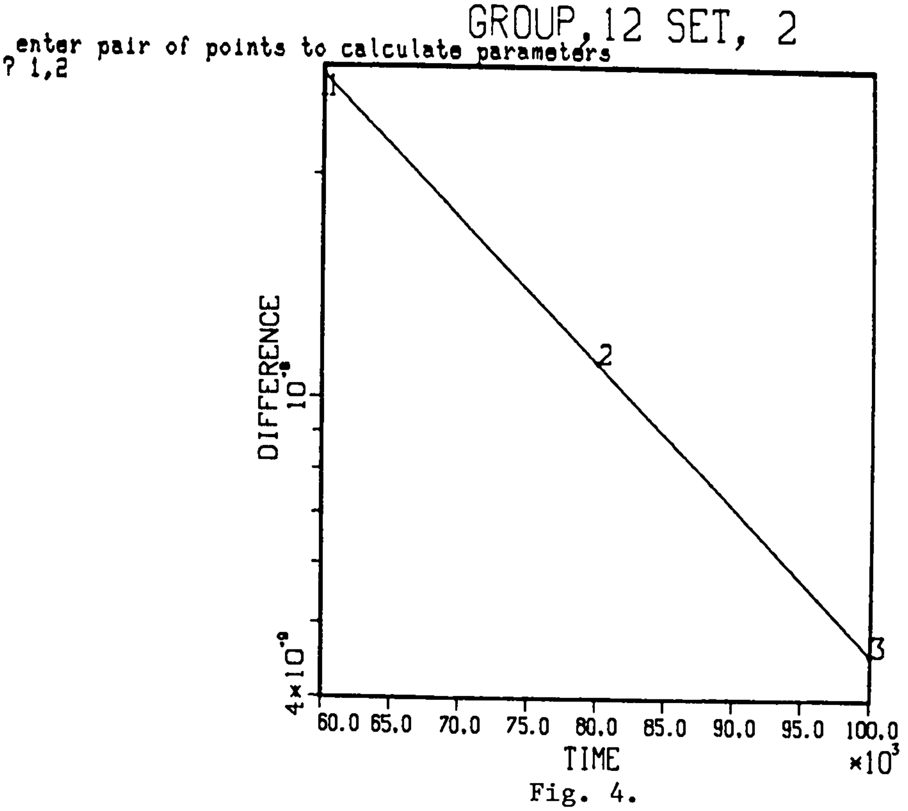
Only minor input changes are needed to run the second pass for the non-linear least-squares fit, namely, set NIAPL=0, NSTEP=1, and NFINL=1. Also File 7 from the first pass must be attached to the input after card 5. Note that for this pass DIFLIM = 5.0 was specified. This means that the non-linear least-squares routine STEPIT will attempt to adjust the α_i and λ_i for a particular group until no calculated point differs from the input data point by more than 5% before proceeding to the next group. If the input guesses for α_i and λ_i supplied by the user are inadequate, STEPIT will continue to adjust the parameters until a minimum chi-square is reached, after which the calculation will proceed to the next group.

Figure 6 shows the per-cent deviation of the fitted data from the original input data. Note that it is within the input specification of 5% over the whole of the fitted cooling-time range. Figure 7 is a final plot of the fit for group 12.



```
enter pair of points to calculate parameters
? 4,5
```

Fig. 3.
Second display for group 12.



```
enter pair of points to calculate parameters
? 1,2
```

Fig. 4.
Third display for group 12.

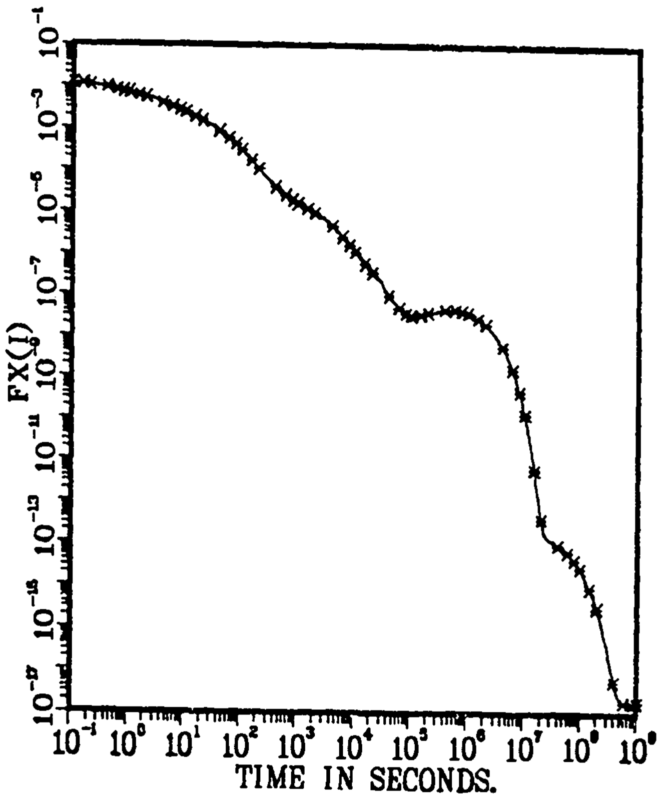


Fig. 5.
First pass fit for group 12.

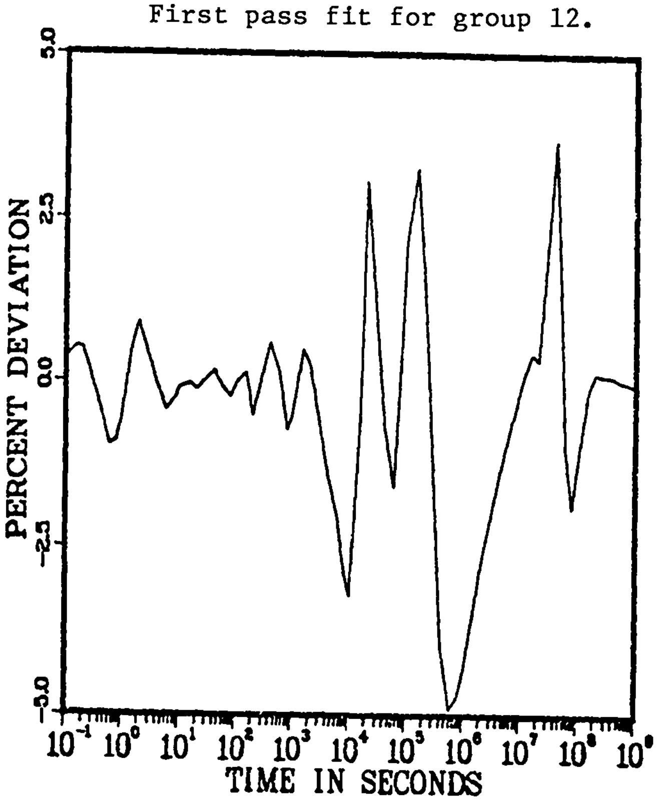


Fig. 6 .
Deviation from input for group 12.

TABLE VI

GROUP 12 RESULTS AFTER USING INTERACTIVE GRAPHICS

Step No.	Time	Original Value	Computed Value	Per cent Difference
1	1.00000E-01	1.18510E-02	1.18512E-02	-2.04759E-03
2	1.50000E-01	1.13322E-02	1.13420E-02	-8.63541E-02
3	2.00000E-01	1.08687E-02	1.08689E-02	-2.23264E-03
4	4.00000E-01	9.43458E-03	9.35955E-03	7.95284E-01
5	6.00000E-01	8.45178E-03	8.35537E-03	1.14073E+00
6	8.00000E-01	7.73742E-03	7.66993E-03	8.72224E-01
7	1.00000E+0C	7.18978E-03	7.16734E-03	3.12085E-01
8	1.50000E+00	6.22239E-03	6.28220E-03	-9.61195E-01
9	2.00000E+00	5.55506E-03	5.63517E-03	-1.44209E+00
10	4.00000E+00	4.02138E-03	4.04444E-03	-5.73376E-01
11	6.00000E+00	3.22970E-03	3.22994E-03	-7.51114E-03
12	8.00000E+00	2.73656E-03	2.73680E-03	-8.86378E-03
13	1.00000E+01	2.39237E-03	2.39509E-03	-1.13562E-01
14	1.50000E+01	1.84014E-03	1.84203E-03	-1.02542E-01
15	2.00000E+01	1.49834E-03	1.49870E-03	-2.40851E-02
16	4.00000E+01	8.43702E-04	8.46685E-04	-3.53606E-01
17	6.00000E+01	5.63688E-04	5.64298E-04	-1.08179E-01
18	8.00000E+01	4.04582E-04	4.04843E-04	-6.46454E-02
19	1.00000E+02	3.03230E-04	3.04241E-04	-3.33395E-01
20	1.50000E+02	1.67961E-04	1.69140E-04	-7.01988E-01
21	2.00000E+02	1.06700E-04	1.07023E-04	-3.02747E-01
22	4.00000E+02	3.86301E-05	3.94753E-05	-2.18780E+00
23	6.00000E+02	2.47972E-05	2.52969E-05	-2.01520E+00
24	8.00000E+02	1.91685E-05	1.94035E-05	-1.22620E+00
25	1.00000E+03	1.60403E-05	1.62711E-05	-1.43869E+00
26	1.50000E+03	1.18872E-05	1.21732E-05	-2.40603E+00
27	2.00000E+03	9.51152E-06	9.73558E-06	-2.35572E+00
28	4.00000E+03	4.62256E-06	4.77524E-06	-3.30301E+00
29	6.00000E+03	2.57195E-06	2.74311E-06	-6.65453E+00
30	8.00000E+03	1.61457E-06	1.77440E-06	-9.89917E+00
31	1.00000E+04	1.11886E-06	1.26184E-06	-1.27792E+01
32	1.50000E+04	5.72884E-07	6.83888E-07	-1.93764E+01
33	2.00000E+04	3.46863E-07	4.27098E-07	-2.31315E+01a
34	4.00000E+04	9.85013E-08	1.06548E-07	-8.16881E+00
35	6.00000E+04	5.10354E-08	5.10354E-08	0.
36	8.00000E+04	3.71013E-08	3.71013E-08	0.
37	1.00000E+05	3.30575E-08	3.33870E-08	-9.96824E-01
38	1.50000E+05	3.42913E-08	3.49155E-08	-1.82011E+00
39	2.00000E+05	3.85599E-08	3.86393E-08	-2.05913E-01
40	4.00000E+05	4.80559E-08	4.56642E-08	4.97680E+00
41	6.00000E+05	4.75911E-08	4.48759E-08	5.70529E+00
42	8.00000E+05	4.37704E-08	4.14022E-08	5.41053E+00
43	1.00000E+06	3.91830E-08	3.72434E-08	4.95012E+00
44	1.50000E+06	2.87345E-08	2.76031E-08	3.93732E+00
45	2.00000E+06	2.09018E-08	2.02247E-08	3.23968E+00
46	4.00000E+06	5.90419E-09	5.78983E-09	1.93694E+00
47	6.00000E+06	1.67887E-09	1.65715E-09	1.29384E+00
48	8.00000E+06	4.78413E-10	4.74402E-10	8.38398E-01
49	1.00000E+07	1.36585E-10	1.35904E-10	4.98015E-01
50	1.50000E+07	6.11894E-12	6.11894E-12	-1.68979E-12
51	2.00000E+07	4.15253E-13	4.15253E-13	-3.89061E-13
52	4.00000E+07	9.67216E-14	1.00144E-13	-3.53809E+00
53	6.00000E+07	6.55882E-14	6.48295E-14	1.15672E+00
54	8.00000E+07	4.28427E-14	4.19710E-14	2.03464E+00
55	1.00000E+08	2.75448E-14	2.71745E-14	1.34410E+00
56	1.50000E+08	9.19704E-15	9.17292E-15	2.62293E-01
57	2.00000E+08	3.10432E-15	3.10432E-15	-4.06587E-13
58	4.00000E+08	5.79643E-17	5.79643E-17	0.
59	6.00000E+08	1.85988E-17	1.86113E-17	-6.73276E-02
60	8.00000E+08	1.80882E-17	1.80948E-17	-3.67576E-02
61	1.00000E+09	1.80799E-17	1.80800E-17	-4.74956E-04

a - maximum difference

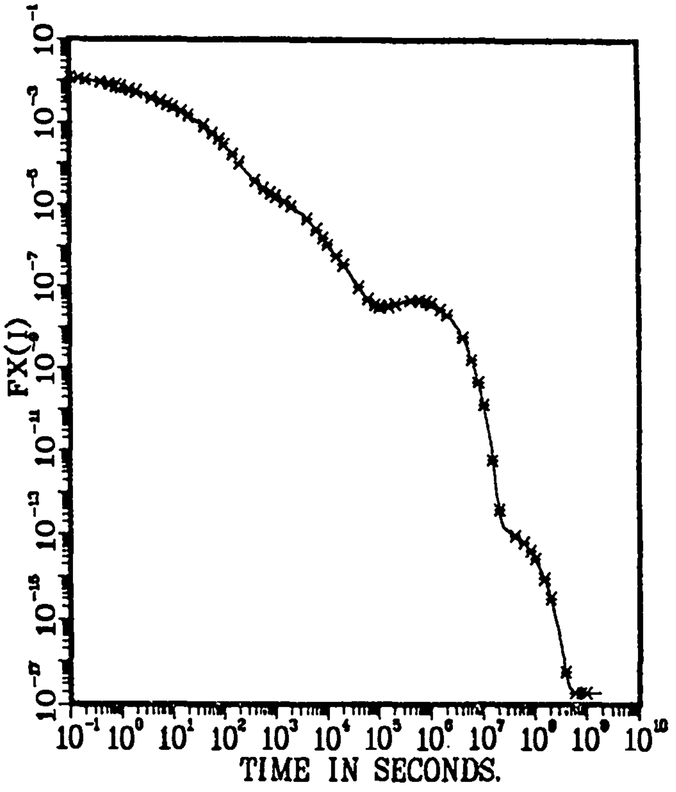


Fig. 7.
Five percent fit for group 12.

It would seem that from the above description of the fitting for group 12 that interactive graphics are not needed and that the whole process could be easily automated. The example of the fitting for group 16, however, makes the need for the interactive graphics much clearer. Figure 8 shows the fit with the interactive graphics bypassed. At first sight, it appears that the fitting in the region 10^5 to 10^7 s cooling time would be similar to that for group 12 except that the set contains 13 rather than 5 points. If we proceed as we did in the fitting for group 5, we would choose the last two points for the parameter calculation that appear in each display of the set (Fig. 9-12). The result is seen in Fig. 13, which could not be fit with sufficient accuracy in STEPIT before this routine converged to minimum (high) chi-square.

Now consider a second attempt at obtaining a fit for group 16 (Fig. 14-17). Instead of choosing points 12 and 13 in the first display, we choose points 11 and 12. As a result, we obtain a first pass fit (Fig. 18) that can easily be brought to within the desired 5% accuracy by the non-linear least-squares routine (Figs. 19 and 20).

The parameters for the fits obtained for all 22 groups are given in Appendix B. The parameters for fits obtained for MAT 1261, MF=80, MT=802 (betas) are also given in Appendix B.

As a second example to illustrate the operation of the FITPULS code, consider an Oak Ridge National Laboratory (ORNL) irradiation experiment,[8] the data for which are shown in Table VII. These experiments, in which ^{239}Pu was irradiated with thermal neutrons, were performed for three irradiation times, namely, 1, 5, and 100 s, and beta and gamma fission-product decay energy were measured. Note that other experiments at other laboratories could also be included in generating a combined pulse function in this example. The input routine RUNTOTS, which was altered to conform to the format in which the experimental data were received, is given in the listing of the code in Appendix A. The formats in the routine are for reading the data as shown in Table VII. Generally, it is more convenient to change the data formats in RUNTOTS than to change the data.

For this particular example, the total decay energy (sum of beta plus gamma) is to be fit and the non-linear least-squares option was chosen on the first pass since a very good guess for the input parameters is available. This is the fit for total decay energy from ^{239}Pu irradiated with a pulse of thermal neutrons as calculated for the ANS standard. These parameters are given in Table VIII under the column labeled "ANS Standard".

The input for this second example is as follows:

Input to Program FITPULS:

Card 1	NPUN = 20
	IRAD = 1
	NCORS = 1
	NIAPL = 0

Input to OVLAY1

Card 2	Title Card

Card 3	IPROB = 1
	NTOTER = 1
	NPUN = 7
	NSTEP = 1
	NFINL = 1

Input to Subroutine RUNTOTS

Card 1A	(Data from Table VII)

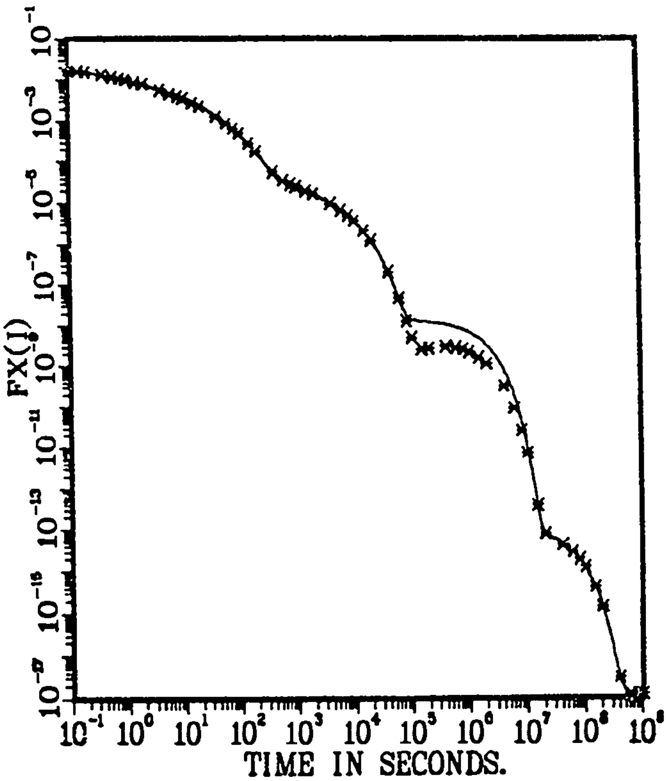
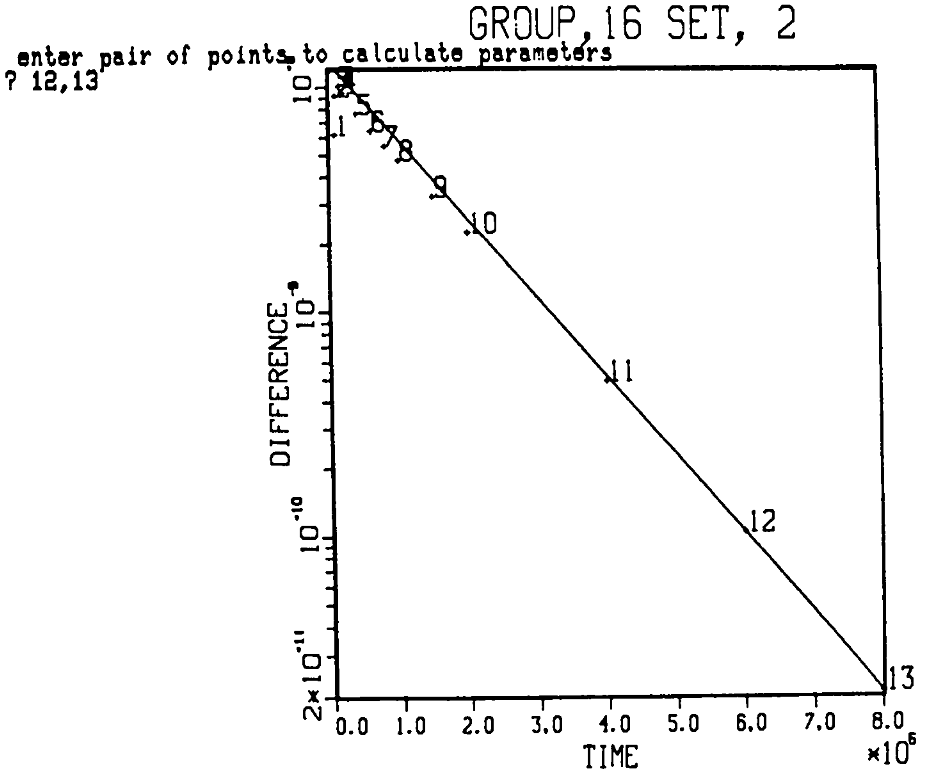
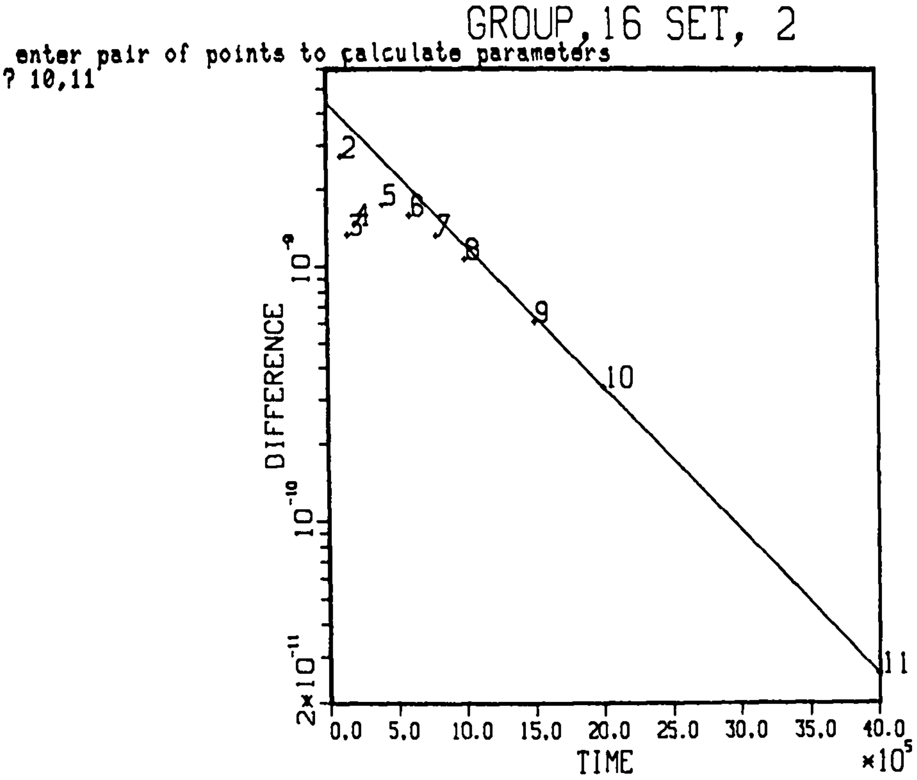


Fig. 8.
Fit without interactive generated parameters for group 16.



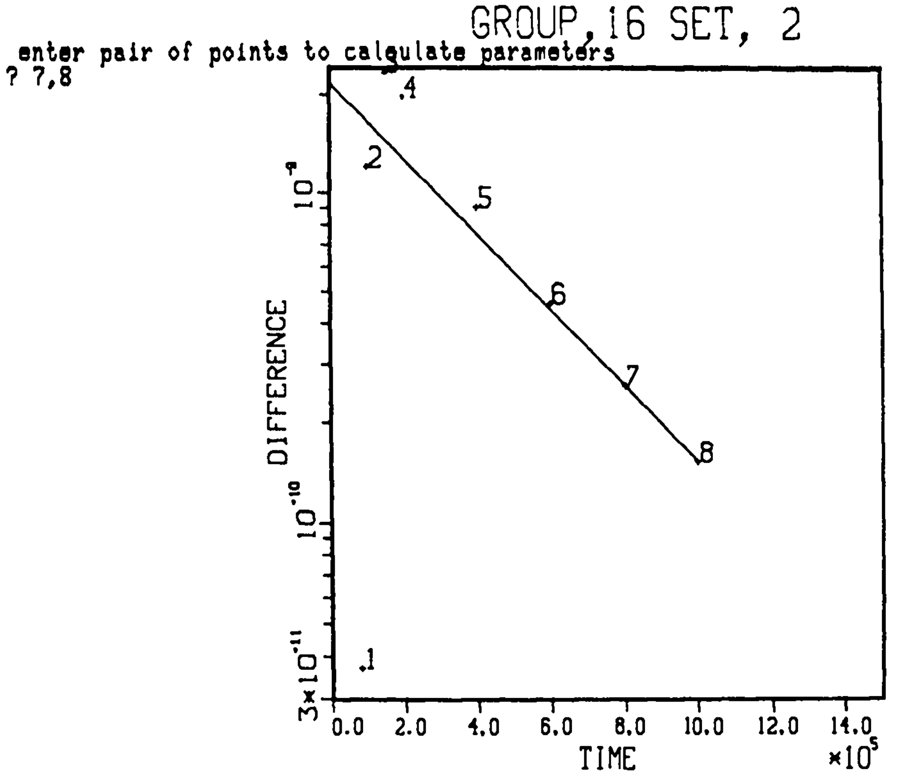
```
GROUP,16 SET, 2
enter pair of points to calculate parameters
? 12,13
```

Fig. 9.
First display, first attempt at group 16 fit.



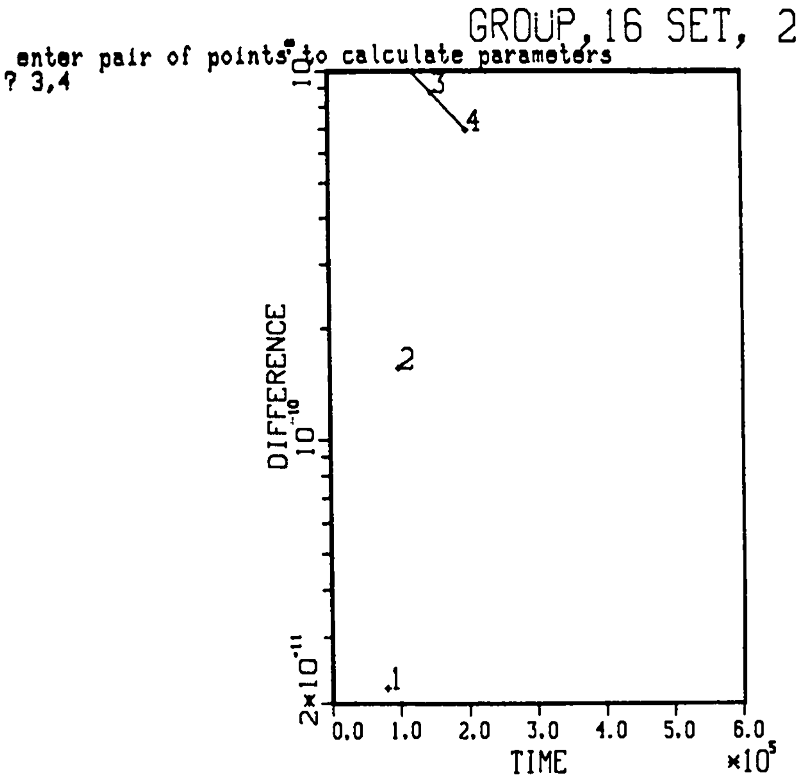
```
enter pair of points to calculate parameters
? 10,11
```

Fig. 10.
Second display, first attempt at group 16 fit.



```
enter pair of points to calculate parameters
? 7,8
```

Fig. 11.
Third display, first attempt at group 16 fit.



```
enter pair of points to calculate parameters
? 3,4
```

Fig. 12.
Fourth display, first attempt at group 16 fit.

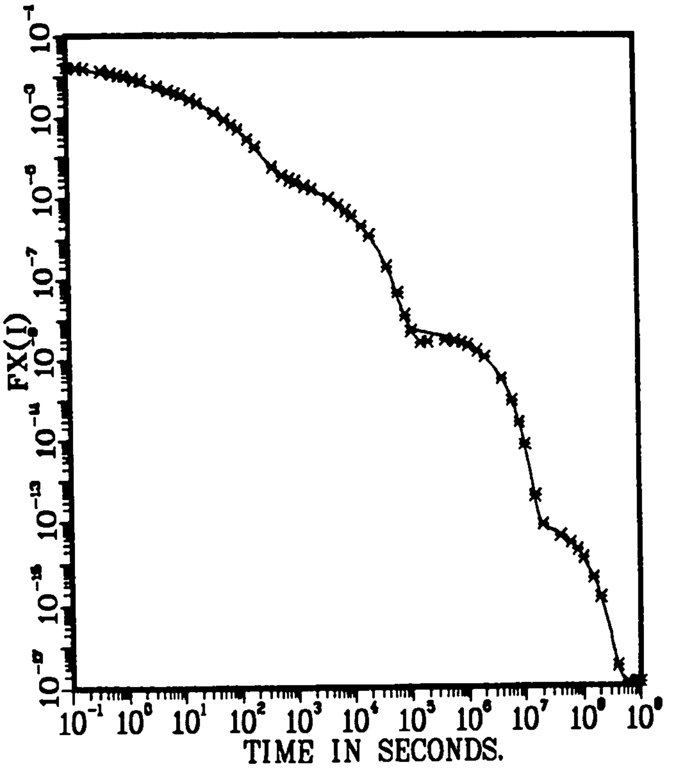
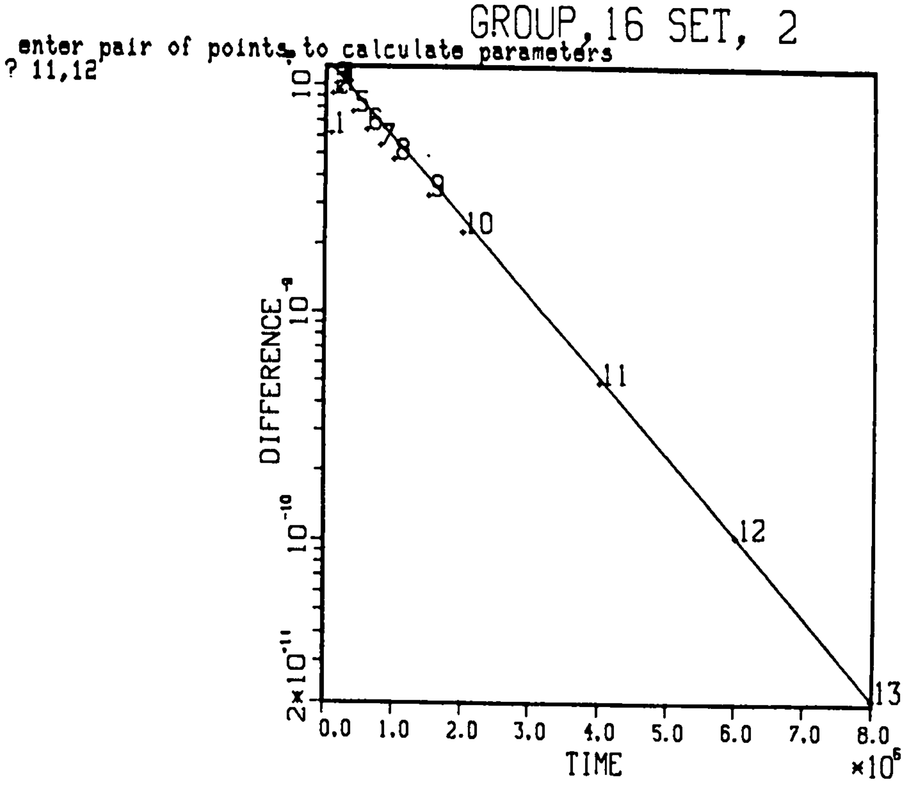
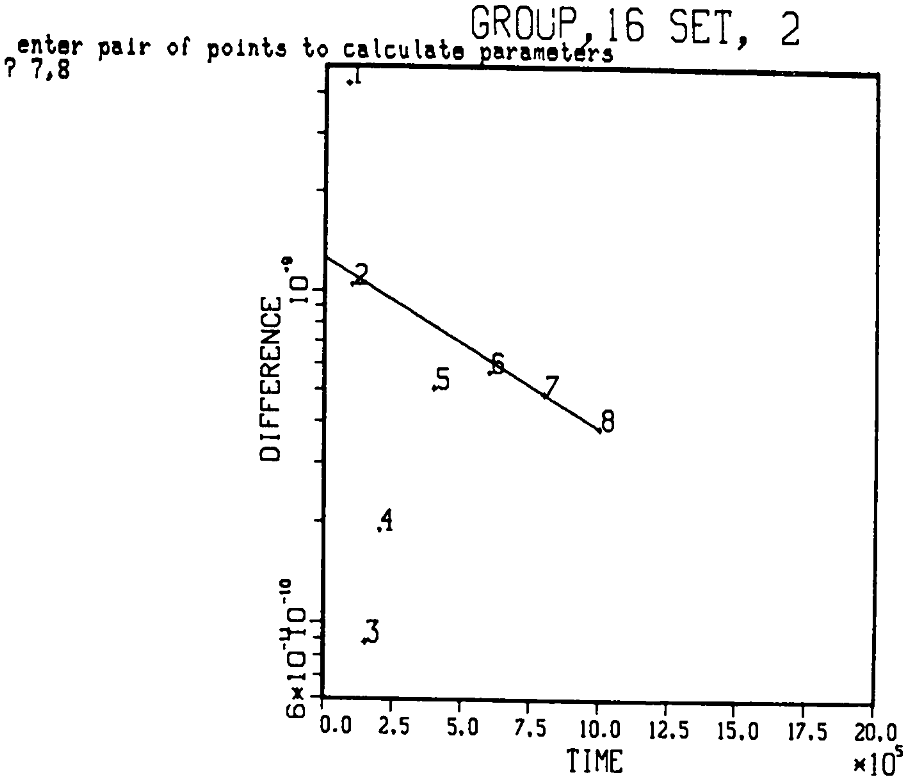


Fig. 13.
Unsuccessful first attempt at group 16 fit.



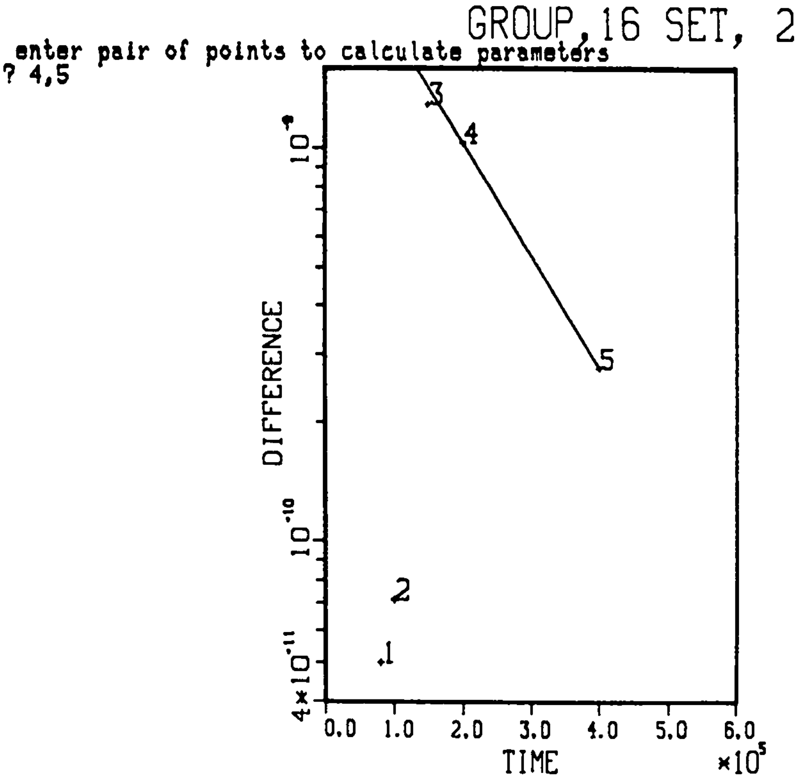
```
enter pair of points to calculate parameters
? 11,12
```

Fig. 14.
First display, second attempt at group 16 fit.



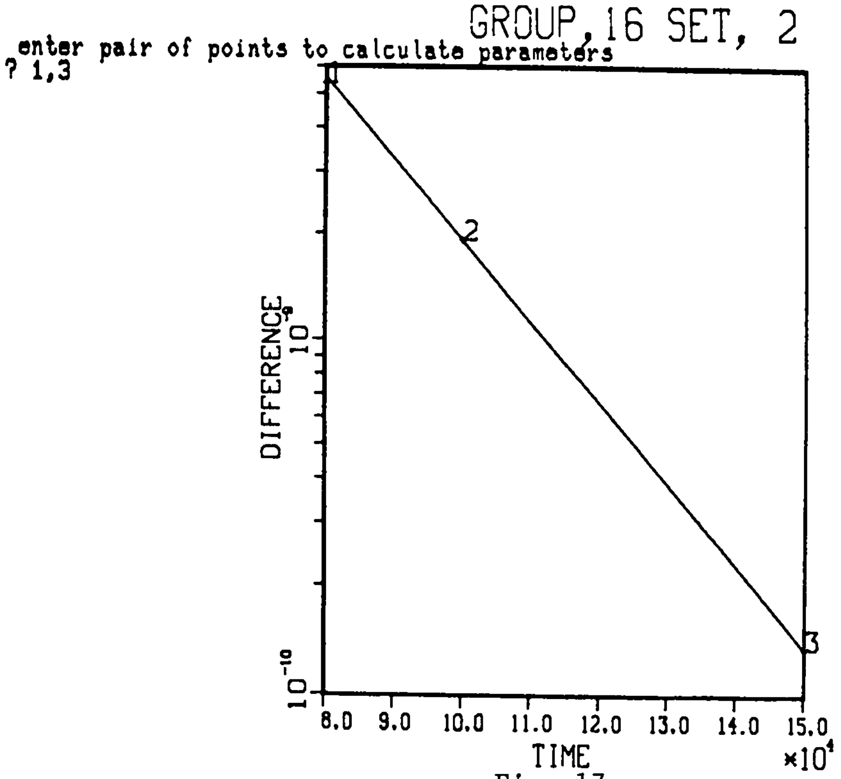
```
enter pair of points to calculate parameters
? 7,8
```

Fig. 15.
Second display, second attempt at group 16 ft.



```
enter pair of points to calculate parameters
? 4,5
```

Fig. 16.
Third display, second attempt at group 16 fit.



```
enter pair of points to calculate parameters
? 1,3
```

Fig. 17.
Fourth display, second attempt at group 16 fit.

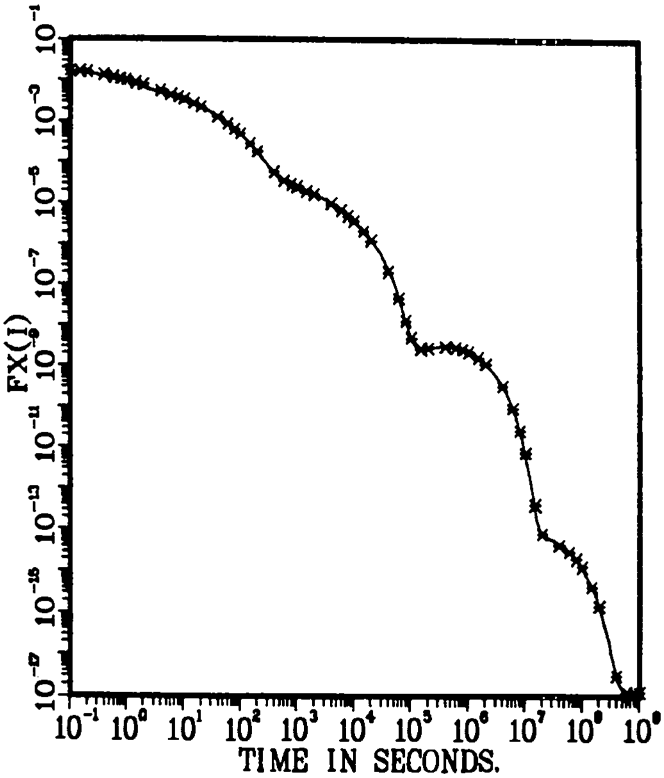


Fig. 18.
Successful first pass group 16 fit.

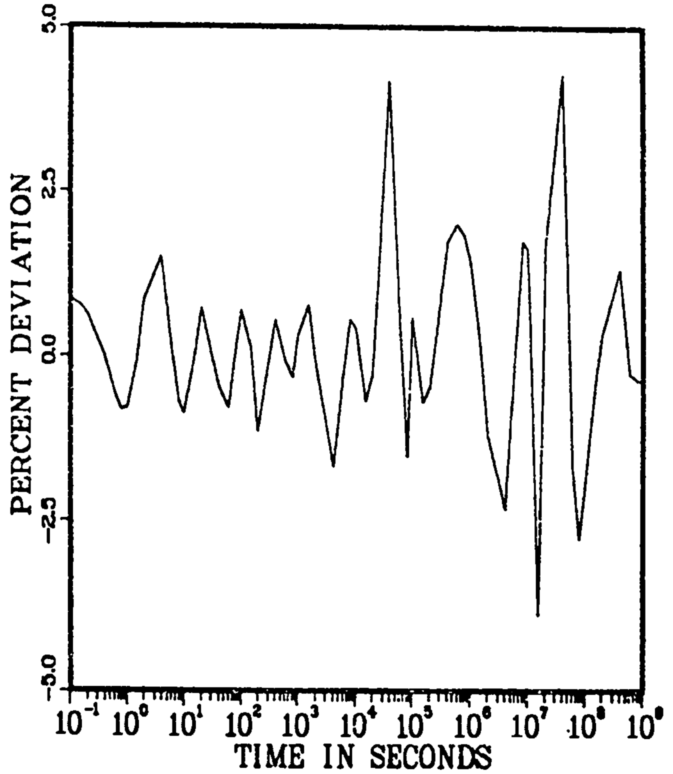


Fig. 19.
Deviation from input for group 16.

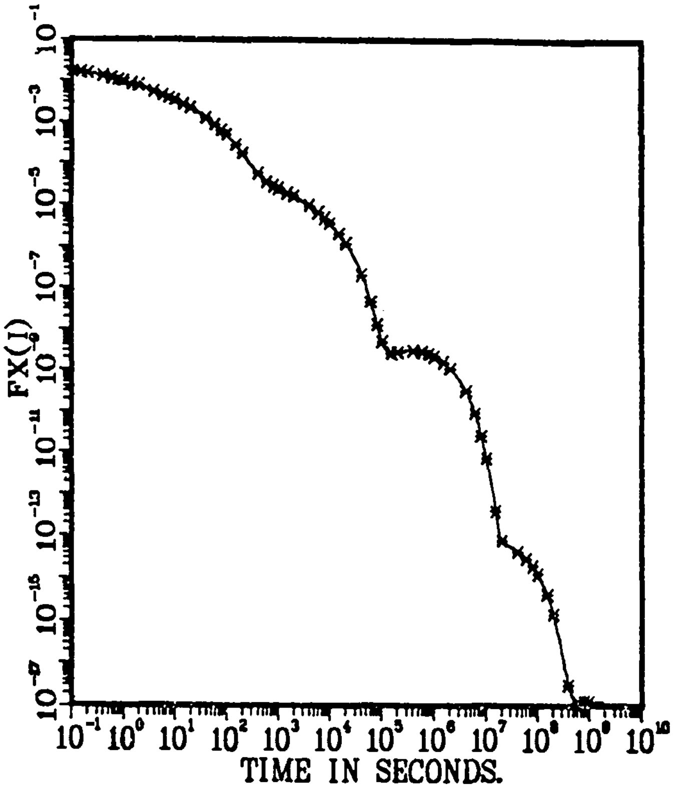


Fig. 20.
Five percent fit for group 16.

Input to OVLAY1 cont'd.

Card 4	DIFLIM = 1.0
	RUNTIM = 295.0
	TMIN = 2.2
	TMAX = 1.195+4
	GXMAX = 0.

Card 5	KKN(1) = 1

The parameters for the ANS standard shown in Table VIII were placed in an ENDF-like format and attached after card 5.

The problem ran to a minimum chi-square convergence of 0.00175 before the input accuracy of 1% could be achieved. The results are given in Table IX, and, as can be seen from this table, the largest per-cent difference is 1.37% occurring at a cooling time of 20.2 s, so the experiment seems to be self-consistent to within at least this amount. The resulting parameters are given in Table VIII under the column labeled "ORNL Exp."

It is interesting to compare these results with the 1978-79 ANS 5.1 standard. It should be noted that the ORNL experiment used as Example 2 was one of three experiments, along with calculated values, considered in formulating the ANS standard. The comparison is shown in Fig. 21. Note that the ORNL experiment lies well within the uncertainty band of the standard.

IV. SUMMARY

This report describes the FITPULS code, and examples are given in which the use of interactive graphics is illustrated. Useful fits for gamma and beta fission-product decay energy spectra in 22 neutron energy groups for ^{235}U fuel irradiated with a pulse of thermal neutrons are given in Appendix B. The fits are given for both units of MeV per fission-second and particles per fission-second.

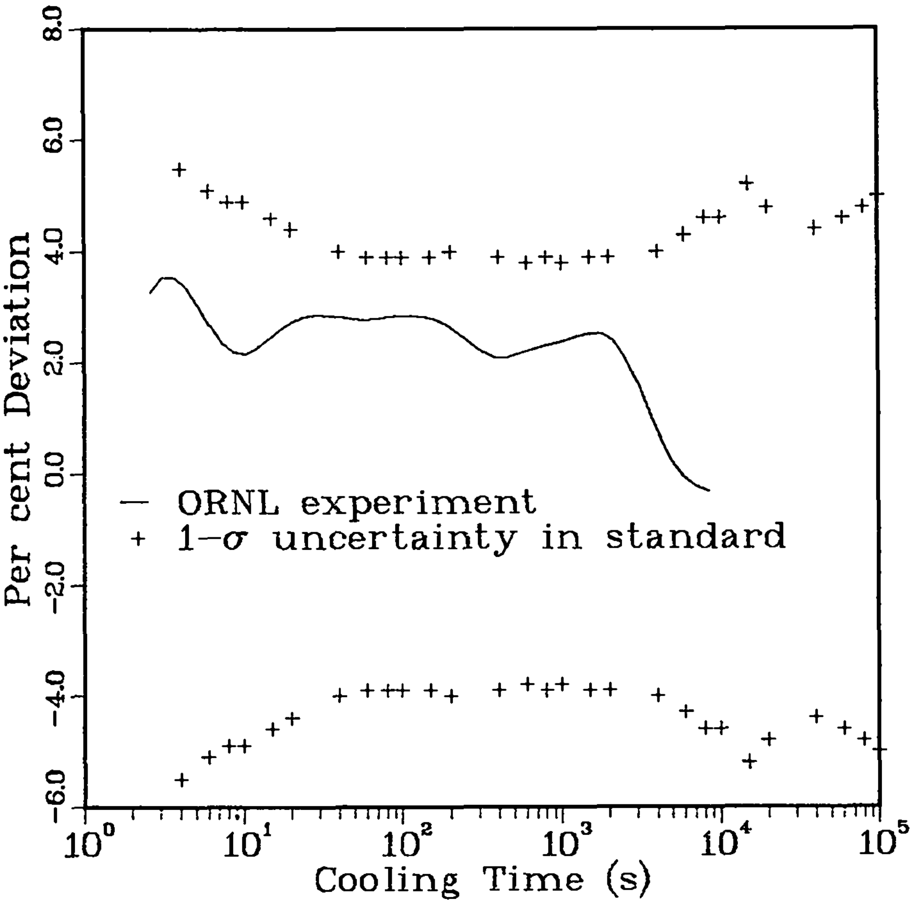


Fig. 21.
ORNL experiment compared to standard.

TABLE VII

ORNL DATA FOR THREE IRRADIATION TIMES OF ^{239}Pu WITH THERMAL NEUTRONS

Irrad	Waiting	Counting	MeV/fiss[a]	
Time(s)	Time(s)	Time(s)	β	γ
1.0	1.7	1.0	0.1811	0.1346
1.0	2.7	1.0	0.1470	0.1044
1.0	3.7	1.0	0.1211	0.0865
1.0	4.7	2.0	0.1945	0.1380
1.0	6.7	3.0	0.2152	0.1537
1.0	9.7	5.0	0.2438	0.1848
1.0	14.7	5.0	0.1680	0.1362
1.0	19.7	5.0	0.1288	0.1094
1.0	24.7	10.0	0.1905	0.1713
1.0	34.7	10.0	0.1406	0.1321
1.0	44.7	15.0	0.1580	0.1558
1.0	59.7	15.0	0.1208	0.1217
1.0	74.7	15.0	0.0980	0.0986
1.0	90.0	20.0	0.1062	0.1056
1.0	110.0	20.0	0.0849	0.0844
5.0	17.7	5.0	0.1258	0.1086
5.0	22.7	10.0	0.1852	0.1698
5.0	32.7	10.0	0.1363	0.1324
5.0	42.7	15.0	0.1545	0.1550
5.0	57.7	15.0	0.1185	0.1208
5.0	72.7	15.0	0.0949	0.0976
5.0	88.0	20.0	0.1028	0.1049
5.0	108.0	20.0	0.0827	0.0846
5.0	128.0	40.0	0.1279	0.1291
5.0	168.0	60.0	0.1366	0.1375
5.0	228.0	70.0	0.1157	0.1163
5.0	298.0	100.0	0.1210	0.1252
5.0	398.0	200.0	0.1703	0.1805
5.0	598.0	200.0	0.1225	0.1347
5.0	798.0	400.0	0.1741	0.1969
100.0	250.0	100.0	0.1218	0.1272
100.0	350.0	200.0	0.1706	0.1823
100.0	550.0	200.0	0.1232	0.1348
100.0	750.0	400.0	0.1766	0.1984
100.0	1150.0	400.0	0.1228	0.1436
100.0	1550.0	400.0	0.0922	0.1112
100.0	1950.0	500.0	0.0867	0.1089
100.0	2450.0	500.0	0.0653	0.0860
100.0	2950.0	1000.0	0.0917	0.1287
100.0	3950.0	2000.0	0.1071	0.1607
100.0	5950.0	2000.0	0.0631	0.0983
100.0	7950.0	2000.0	0.0439	0.0677
100.0	9950.0	4000.0	0.0617	0.0872

[a] See Ref. 8 for a clarification of this unit.

TABLE VIII

EXAMPLE 2: FIT TO EXPERIMENTAL DATA (^{239}Pu THERMAL FISSION)

Parameters from ANS Standard		Fit to ORNL Exp.	
Alpha	Lambda	Alpha	Lambda
2.0830E-01	1.0020E+01	3.5696E+00	3.5491E+00
3.8530E-01	6.4330E-01	4.1004E-01	7.2551E-01
2.2130E-01	2.1860E-01	2.2288E-01	2.2181E-01
9.4600E-02	1.0040E-01	9.5227E-02	1.0367E-01
3.5310E-02	3.7280E-02	3.4997E-02	3.88334E-02
2.2920E-02	1.4350E-02	2.2816E-02	1.4790E-02
3.9460E-03	4.5490E-03	4.0597E-03	4.7821E-03
1.3170E-03	1.3280E-03	1.3783E-03	1.4102E-03
7.0520E-04	5.3560E-04	7.0010E-04	5.3448E-04
1.4320E-04	1.7300E-04	1.4553E-04	1.7380E-04
1.7650E-05	4.8810E-05	1.7797E-05	4.7344E-05
7.3470E-06	2.0060E-05	7.1997E-06	1.8562E-05
1.7470E-06	8.3190E-06	1.6824E-06	1.9636E-06
5.4810E-07	2.3580E-06	4.3891E-07	2.9475E-07
1.6710E-07	6.4500E-07	1.4408E-07	8.0625E-08
2.1120E-08	1.2780E-07	1.2289E-08	1.5975E-08
2.9960E-09	2.4660E-08	2.8362E-09	3.1109E-09
5.1070E-11	9.3780E-09	3.4834E-11	1.3855E-09
5.7300E-11	7.4500E-10	5.4918E-11	1.5831E-09
4.1380E-14	2.4260E-10	1.1226E-14	3.5460E-10
1.0880E-15	2.2100E-13	9.6156E-13	1.1673E-13
2.4540E-15	2.6400E-14	3.1984E-17	4.0494E-14
7.5570E-17	1.3800E-14	1.7054E-16	9.6888E-15

TABLE IX

FINAL FIT OF ORNL EXPERIMENT

Time	Original Value	Computed Value	Per cent Difference
.2200E+01	.3157E+00	.3158E+00	.2457E-01
.3200E+01	.2514E+00	.2509E+00	-.1963E+00
.4200E+01	.2076E+00	.2083E+00	.3527E+00
.5700E+01	.1663E+00	.1660E+00	-.1213E+00
.8200E+01	.1230E+00	.1232E+00	.1942E+00
.1220E+02	.8572E-01	.8543E-01	-.3377E+00
.1720E+02	.6084E-01	.6089E-01	.7794E-01
.2220E+02	.4764E-01	.4733E-01	-.6593E+00
.2970E+02	.3618E-01	.3573E-01	-.1248E+01
.3970E+02	.2727E-01	.2710E-01	-.6221E+00
.5220E+02	.2092E-01	.2080E-01	-.5785E+00
.6720E+02	.1617E-01	.1612E-01	-.3180E+00
.8220E+02	.1311E-01	.1300E-01	-.7953E+00
.1000E+03	.1059E-01	.1045E-01	-.1362E+01
.1200E+03	.8465E-02	.8454E-02	-.1269E+00
.2020E+02	.2344E+00	.2376E+00	.1368E+01a
.2770E+02	.1775E+00	.1790E+00	.8632E+00
.3770E+02	.1344E+00	.1357E+00	.9736E+00
.5020E+02	.1032E+00	.1041E+00	.8689E+00
.6520E+02	.7977E-01	.8061E-01	.1057E+01
.8020E+02	.6417E-01	.6503E-01	.1348E+01
.9800E+02	.5193E-01	.5224E-01	.6079E+00
.1180E+03	.4183E-01	.4228E-01	.1084E+01
.1480E+03	.3213E-01	.3239E-01	.8367E+00
.1980E+03	.2284E-01	.2292E-01	.3217E+00
.2630E+03	.1657E-01	.1656E-01	-.7113E-01
.3480E+03	.1231E-01	.1236E-01	.3955E+00
.4980E+03	.8770E-02	.8801E-02	.3504E+00
.6980E+03	.6430E-02	.6462E-02	.4997E+00
.9980E+03	.4638E-02	.4640E-02	.5759E-01
.3000E+03	.2490E+00	.2495E+00	.2156E+00
.4500E+03	.1765E+00	.1767E+00	.1560E+00
.6500E+03	.1290E+00	.1295E+00	.4021E+00
.9500E+03	.9375E-01	.9291E-01	-.8932E+00
.1350E+04	.6660E-01	.6672E-01	.1783E+00
.1750E+04	.5085E-01	.5085E-01	.7817E-02
.2200E+04	.3912E-01	.3911E-01	-.1785E-01
.2700E+04	.3026E-01	.3039E-01	.4360E+00
.3450E+04	.2204E-01	.2205E-01	.5006E-01
.4950E+04	.1339E-01	.1330E-01	-.6353E+00
.6950E+04	.8070E-02	.8085E-02	.1868E+00
.8950E+04	.5580E-02	.5591E-02	.2058E+00
.1195E+05	.3723E-02	.3719E-02	-.9974E-01

a - maximum difference

REFERENCES

1. R. J. LaBauve, T. R. England, D. C. George, and M. G. Stamatelatos, "The Application of a Library of Processed ENDF/B-4 Fission-Product Aggregate Decay Data in the Calculation of Decay-Energy Spectra," Los Alamos Scientific Laboratory report LA-7483-MS (1978).

2. R. J. LaBauve and T. R. England, "Rapid Spectral and Decay-Heat Calculations Using Processed ENDF/B Fission Product Data," Trans. Am. Nucl. Soc. 28, 749 (1978).

3. R. J. LaBauve, T. R. England and W. B. Wilson, "Absorption Effects on Decay Heat and Spectral Pulse Kernels," Trans. Am. Nucl. Soc. 32, 739 (1979).

4. "American National Standard for Decay Heat Power in Light Water Reactors," Am. Nucl. Soc. report ANSI/ANS-5.1-1979 (1979).

5. R. J. LaBauve, T. R. England, M. G. Stametalatos, and D. C. George, "Approximations to Summation Calculations of Delayed Energy and Spectra from Fission Products," Los Alamos Scientific Laboratory report LA-6684-MS (1977).

6. J. H. Burrill, Jr., "360-STEPIT, A User's Manual," Ohio University Research Institute report TNP-1966-2 (1966).

7. M. E. Battat, R. J. LaBauve, and D. W. Muir, "The GAMMON Activation Library," Los Alamos Scientific Laboratory report LA-8040-MS (1979).

8. J. K. Dickens, J. F. Emery, T. A. Love, J. W. McConnell, K. J. Northcutt, R. W. Peelle, and H. Weaver, "Fission Product Energy Release for Times Following Thermal-Neutron Fission of ^{239}Pu Between 2 and 14 000 Seconds," Oak Ridge National Laboratory report ORNL/NUREG-34 (April 1978).

APPENDIX A

ORGANIZATION AND LISTING OF FITPULS CODE

The FITPULS code is organized into a main program (FITPULS) and three overlays (subroutine OVRLAY1, subroutine OVRLAY2, and subroutine OVRLAY3). The purpose of the main program is to read initial input and to call desired overlays.

The purpose of OVRLAY1 is to read additional input data and to rebin decay-energy input data if desired. This is done in subroutines CORSBIN and REBIN. OVRLAY1 also contains a subroutine, SELECT, for selecting a subset of the decay-energy input data to be fit and a subroutine, TRMSEE, used to aid the user in adjusting parameters to improve the fit. Finally, the subroutine RUNTOTS in OVRLAY1 is used for reading in experimental data.

OVRLAY2 contains subroutines for fitting options 1 and 2. These include the routines PULSFIT, PDIF, and SEEFIT. This overlay also contains subroutines for placing the final output (the parameters for the fit) in an ENDF-like format, routines PCHOUT and CXFP, and the subroutines used in the interactive graphics, routines SETUP, YMNMX, DECBND, PTLPT, and PICTUR.

OVRLAY3 contains the subroutines used for fitting option 3, the non-linear least-squares option. These include DHFIT, FUNK, and STEPIT. This overlay also contains the subroutine FINECHK for graphically examining the accuracy of the fit.

Users of FITPULS should be warned that the version of the code for which the listing below is given is dependent upon the Los Alamos Scientific Laboratory operating system and that some changes would be necessary if the code is used at another installation. In particular, a local linear equation solver and local plotting routines are used in this version.

LASL Identification
No. LP-847

```
      PROGRAM FITPULS (TAPE5,TAPE6,TAPE7,TAPE10,TAPE20,OUTPUT,TTY,TAPE11
     1 =TTY)
C
C     THIS PROGRAM ACCEPTS FISSION-PRODUCT DATA (BETA AND GAMMA) IN
C     UNITS OF ENERGY /FISSION WHICH HAVE BEEN ENERGY BINNED INTO FINE
C     GROUPS(150)FOR A NUMBER OF COOLING TIME STEPS DERIVED AS FOLLOWS-
C       YIELD DATA FROM ENDF WERE FIRST PROCESSED BY THE FPCYS CODE TO
C       SUPPLY FINE GROUP INPUT FOR THE FPSPEC CODE.  FPSPEC ALSO
C       REQUIRES OUTPUT FROM THE CINDER-10 CODE.  FINALLY THE OUTPUT OF
C       FPSPEC WAS PROCESSED BY THE FOTOELF CODE WHICH PUTS THE DATA IN
C       AN ENDF-LIKE FORMAT WHICH IS THE INPUT DATA LIBRARY FOR THIS
C       CODE(FITPULS).
C
C     FITPULS REBINS THIS DATA INTO A USER CHOSEN BROAD GROUP STRUCTURE
C     (NOTE THAT THIS REBINNING CHANGES UNITS TO ENERGY/FISSION-SEC.
C     OR,OPTIONALLY,TO UNITS OF NUMBER OF PARTICLES PER FISSION)
C     AND FITS THE DATA FOR EACH GROUP WITH A LINEAR COMBINATION OF
C     FUNCTIONS AS FOLLOWS --
C
C     GXC(I ,K)=SUM+ALF(L,K)*(EXP(-ALAM(L,K)*T(I )),SUM OVER L=1,KTRM,
C              WHERE KTRM IS THE NUMBER OF PAIRS OF PARAMETERS (ALF,
C              ALAM) FOR GROUP K.
C
C     GXC(I ,K)=ENERGY(MEV)/FISS-SEC FOR GROUP K AND COOLING TIME T(I ).
C     GXC(I ,K) CAN ALSO BE EXPRESSED AS PARTICLES/FISS BY SETTING THE
C         FLAG MEVU=1.
C     MAX.ALLOWED VALUE FOR KTRM(KTR(K)),NO. OF PARAM. PAIRS/GP=50.
C     MAX.ALLOWED NO. OF BROAD GROUPS,NERG=25.
C     MAX. ALLOWED NO. OF INPUT COOLING TIME STEPS,ITSP=70.
C
C     THE CODE ALSO HAS AN OPTION OF REDUCING EXPERIMENTAL DATA FOR
C     A FINITE IRRADIATION TIME TO A PULSE AND OBTAINING A FIT FOR THE
C     EQUIVALENT PULSE.  FOR THIS OPTION SET IRAD=1.
C
C      THERE ARE THREE FITTING OPTIONS AVAILABLE IN FITPULS. THE FIRST
C      TWO RUN VERY FAST WHEREAS THE THIRD, A NON-LINEAR LEAST
C      SQUARES FITTING OPTION IS TIME CONSUMING.  THESE OPTIONS ARE
C      SELECTED BY THE FLAG NIAPL AS FOLLOWS-
C         NIAPL=1  THE FIRST OPTION SELECTED. BEGINNING WITH LONGEST
C                  COOLING TIME PARAMETERS ARE COMPUTED FOR PAIRS OF
C                  USER SELECTED POINTS.  INTERACTIVE GRAPHICS AID IN
C                  POINT SELECTION. CANNOT BE USED FOR FINITE
C                  IRRADIATION FIT.  GOOD FOR PULSE FIT ONLY.
C         NIAPL=-1 THE SECOND OPTION IS SELECTED.  ALAM VALUES MUST BE
C                  INPUT AS THIS OPTION ONLY COMPUTES ALF VALUES GIVEN
C                  ALAM VALUES. METHOD DESCRIBED IN LA-6684-MS. USE
C                  THIS METHOD FOR FINITE IRRADIATION FITTING.
C         NIAPL=0  THE THIRD OPTION SELECTED. GOOD GUESSES FOR BOTH
C                  ALF AND ALAM VALUES MUST BE INPUT AS THIS IS THE
C                  NON-LINEAR LEAST-SQUARES FITTING OPTION. GUESSES
C                  ARE OBTAINED BY FIRST RUNNING ONE OF THE FIRST TWO
C                  OPTIONS.
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN , NOUT
      COMMON /PULSDAT/ NDT, EB(25), GX(25,400), NERG, ALF(25,50), ALAM(2
     1 5,50)
      LEVEL 2, A, B
      COMMON /PULSCAL/ A(50,50), B(50,1)
      COMMON /PULSOUT/ ALPHA(50), FXC(100), PCT(100)
      COMMON /MANI / WX(100), TITL(8), KTR(50), NS(10), KKN(25), DIFLIM
      COMMON /ENDF/MEVU, MAT, MF, MT, RUNTIM, NPUN
      COMMON /TRMOT/ TL(10), LTM(50), LT
      COMMON /FINRAD/ NIAPL, IRAD, NCORS, RADT(200), DELT(200)
      COMMON /COMO/ NP, TMIN, TMAX, NSTEP, GXMIN, TC(100), LXX(20),
     1 NFINL
      DATA NIN, NOUT, NDT /5,6,10/
C
      CALL FILEREP
      READ (NIN,20) NPUN,IRAD,NCORS,NIAPL
C
C     SET NPUN=7 HERE IF REBINNED DATA FILE WANTED,OTHERWISE NPUN=20.
C     IRAD=0,REGULAR PULSE FIT REQUESTED, =1,FIT FOR FINITE IRRADIATION
C           TIME DATA WANTED.
C     NCOR=0,CALL CORSBIN , =1,NO CALL.
C     NIAPL=1 FIT OPTION 1; NIAPL=-1 FIT OPTION 2; NIAPL=0 FIT OPTION 3
```

```
C     NIAPL=1  INTERACTIVE POINT SELECTION
      IF (NIAPL.EQ.1) CALL GTEKT (1HU,4H4014,960)
      IF (NIAPL.NE.1) CALL GPLOT (1HU,7HFITPULS,7)
      CALL OVERLAY (5HOVLY1,1,0,0)
      IF (NFINL.EQ.1) GO TO 10
      CALL OVERLAY (5HOVLY2,2,0,0)
   10 CONTINUE
      CALL OVERLAY (5HOVLY3,3,0,0)
      IF (NPUN.EQ.7) CALL PCHOUT (LXX)
      CALL DONEPL
      STOP
C
   20 FORMAT (12I6)
      END


      SUBROUTINE OVRLAY1
C
C     OVRLAY 1 READS INPUT AND REBINS DATA
C
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN, NOUT
      COMMON /PULSDAT/ NDT, EB(25), GX(25,400), NERG, ALF(25,50), ALAM(2
     1 5,50)
      LEVEL 2, A, B
      COMMON /PULSCAL/ A(50,50), B(50,1)
      COMMON /PULSOUT/ ALPHA(50), FXC(100), PCT(100)
      COMMON /MANI/ WX(100), TITL(8), KTR(50), NS(10), KKN(25), DIFLIM
      COMMON /ENDF/MEVU, MAT, MF, MT, RUNTIM, NPUN
      COMMON /TRMOT/ TL(10), LTM(50), LT
      COMMON /FINRAD/ NIAPL, IRAD, NCORS, RADT(200), DELT(200)
      COMMON /COMO/ NP, TMIN, TMAX, NSTEP, GXMIN, TC(100), LXX(20),
     1 NFINL
      REAL TMN(20), TMX(20)
C
      IF (NCORS.LE.0) CALL CORSBIN
C
C     SEE CORSBIN FOR MAT1,MF1,MT1 INPUT.
C
      READ (NIN,40) (TITL(I),I=1,8)
C
C     TITL=EIGHTY CHARACTER(HOLLERITH) TITLE.
C
C     TITL(1)=CHARACTER FOR CALLING SUBROUTINE SELECT TO CHOOSE DATA
C             TO BE USED IN FIT.  IF TITL(1)= SELECT    ,SEE SUBROUTINE
C             SELECT FOR INPUT.
C             IF TITL(1)= DO NOT GO,PROGRAM STOPS.  USED WHEN JUST
C             REBINNED DATA DESIRED.
C
      IF (TITL(1).EQ.10H SELECT    ) CALL SELECT
      WRITE (NOUT,50) (TITL(I),I=1,8)
      IF (TITL(1).EQ.10H DO NOT GO) STOP
      REWIND NPUN
      READ (NIN,60) IPROB,NTOTER,NPUN,NSTEP,NFINL
C
C     IPROB = PROBLEM NO.  MAKE NEGATIVE IF FIT IS MADE IN SEGMENTS.
C     NTOTER = FLAG TO DENOTE SPEC. OR TOTAL CALC.,=0,CODE READS SPECT.
C              DATA FROM TAPE FILE,=1,CODE READS DATA FROM INPUT FILE.
C     NPUN = FLAG FOR PARAMETERS OUTPUT IN ENDF-LIKE FORMAT,=7,OUTPUT,
C            =20 NO OUTPUT.
C     NSTEP = FLAG TO CALL NON-LINEAR LEAST SQUARES ROUTINE,=0,ROUTINE
C            NOT CALLED,=1, ROUTINE CALLED.
C     NFINL = FLAG FOR READING ALL PARAMATERS FROM PREVIOUS PROB,I.E.,
C             PULSFIT WILL NOT BE CALLED FOR ANY GROUP,=0,NO EFFECT,=1,
C             SEE READ,STATEMENTS BELOW.
C
      IF (NTOTER.GT.0) CALL RUNTOTS
      READ (NIN,70) DIFLIM,RUNTIM,TMIN,TMAX,GXMIN
C
C     DIFLIM = MAX. PERCENT POINTWISE DEVIATION ALLOWED IN STEPIT.
C              USUALLY SET HIGH ON INITIAL PASSES AND TIGHTENED UP  IN
C              SUBSEQUENT PASSES.
C     RUNTIM = RUNNING TIME.  MAKE FRACT. OF SECOND LESS THAN THAT USED
C              ON CONTROL CARD TO GET OUTPUT FILE FOR SUBSEQUENT RUN.
```

```
C     TMIN = LOWEST COOLING TIME DESIRED.
C     TMAX = HIGHEST COOLING TIME DESIRED.
C     GXMIN = MINIMUM ALLOWED VALUE OF GX(I,K).  THIS SHOULD BE SET
C            SO NO FIT IS ATTEMPTED OVER MORE THAN ABOUT 15 DECADES.
C
      NP=ITSP
      IF (T(1).LE.0.0) T(1)=1.E-4
      IF (T(NP).LE.0.0) NP=NP-1
      DO 10 N=1,NP
      TC(N)=T(N)
   10 CONTINUE
      NE=NERG
      NBDS=NE+1
      EB(NBDS)=7.5
C
C     KKN = FLAG FOR RUNNING CODE GROUP BY GROUP, =0,GROUP SKIPPED,
C          =GROUP NUMBER, FIT COMPUTED FOR THAT GROUP, =NEGATIVE GROUP
C          NO., TERMSEE ROUTINE CALLED.
C
      READ (NIN,60) (KKN(K),K=1,NERG)
C
C     OUTPUT FILE FROM PREVIOUS PROBLEM ARE READ HERE IF NFINL=1.
C
      IF (NFINL.NE.1) GO TO 30
      READ (NIN,40) (TITL(I),I=1,8)
      READ (NIN,80) C1,C2,NUL,NLU,NUL,NERG
      DO 20 K=1,NERG
      READ (NIN,80) EB(K),EB(K+1),NUL,NUL,NUL,KNG
      READ (NIN,80) TMN(K),TMX(K),NUL,NUL,NUL,KTR(K)
      KTRM=KTR(K)
      READ (NIN,90) (ALF(K,L),ALAM(K,L),L=1,KTRM)
      KKX=KKN(K)
      IF (KKX.LT.0) CALL TRMSEE (K,1)
      IF (KKX.LT.0) CALL TRMSEE (K,0)
      KTR(K)=KTRM
   20 CONTINUE
   30 CONTINUE
      RETURN
C
   40 FORMAT (8A10)
   50 FORMAT (1H1,10X,8A10)
   60 FORMAT (12I6)
   70 FORMAT (6E12.5)
   80 FORMAT (2E11.4,4I11,I4,I2,I3,I5)
   90 FORMAT (6E11.4)
      END


      SUBROUTINE SELECT
C
C     THIS ROUTINE USED FOR SELECTING A SUBSET OF THE DATA TO BE FIT.
C
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN, NOUT
      COMMON /PULSDAT/ NDT, EB(25), GX(25,400), NERG, ALF(25,50), ALAM(2
     1 5,50)
      LEVEL 2, A, B
      COMMON /PULSCAL/ A(50,50), B(50,1)
      COMMON /MANI/ WX(100), TITL(8), IPT(50), NS(10), KKN(25), DIFLIM
C
      READ (NIN,50) ITS,(IPT(I),I=1,ITS)
C
C     ITS=NO OF TIME STEPS DESIRED.
C     IPT=INDEXES OF DESIRED TIME STEPS.
C
      DO 20 K=1,NERG
      DO 10 I=1,ITS
      II=IPT(I)
      ALAMDA(I)=T(II)
      A(K,I)=GX(K,II)
   10 CONTINUE
   20 CONTINUE
      DO 40 K=1,NERG
      DO 30 I=1,ITS
      GX(K,I)=A(K,I)
      T(I)=ALAMDA(I)
   30 CONTINUE
      WRITE (NOUT,60) (I,T(I),GX(K,I),I=1,ITS)
   40 CONTINUE
      ITSP=ITS
      RETURN
C
   50 FORMAT (12I6)
   60 FORMAT (4H  I=,I3,3H T=,1E12.5,4H GX=,1E12.5)
      END


      SUBROUTINE TRMSEE (LK,KKX)
C
C     THIS ROUTINE IS CALLED IF KKN IS SET NEGATIVE FOR A GROUP.
C     ROUTINE PRINTS OUT TERM BY TERM CALCULATION OF FX FOR USE IN
C     ADJUSTING FITTING PARAMETERS.  THIS DONE BY CHANGING AND/OR
C     REMOVING PARAMETERS.
C
      COMMON /MANI/ WX(100), TITL(8), KTR(50), NS(10), KKN(25), DIFLIM
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN, NOUT
      COMMON /PULSDAT/ NDT, EB(25), GX(25,400), NERG, ALF(25,50), ALAM(2
     1 5,50)
      LEVEL 2, TRM, TPRT
      COMMON /PULSCAL/ TRM(50,50), TPRT(50)
      COMMON /FINRAD/ NIAPL, IRAD, NCORS, RADT(200), DELT(200)
      COMMON /TRMOT/ TL(10), LTM(50), LT
C
      K=LK
      KTRM=KTR(K)
      TL(1)=10HORIGINAL P
      TL(2)=10HARAMATRS F
      TL(3)=10HOR GROUP
      IF (KKX.EQ.0) GO TO 10
      WRITE (NOUT,150) (TL(I),I=1,3),K
      WRITE (NOUT,160) (L,ALF(K,L),ALAM(K,L),L=1,KTRM)
   10 IF (KKX.GT.0) GO TO 90
      DO 40 I=1,ITSP
      FX(I)=0.
      DO 30 L=1,KTRM
      TRM(I,L)=ALF(K,L)*EXP(-ALAM(K,L)*T(I))
      IF (IRAD.LE.0) GO TO 20
      IF (TRM(I,L).LT.0.) TRM(I,L)=0.
      COFF=ALF(K,L)/DELT(I)/ALAM(K,L)**2
      XPO1=1.-EXP(-ALAM(K,L)*RADT(I))
      XPO2=1.-EXP(-ALAM(K,L)*DELT(I))
      XPO3=EXP(-ALAM(K,L)*(T(I)-DELT(I)/2.))
   20 CONTINUE
      IF (IRAD.EQ.1) TRM(I,L)=COFF*XPO1*XPO2*XPO3
      LTM(L)=L
      FX(I)=FX(I)+TRM(I,L)
   30 CONTINUE
   40 CONTINUE
      K1=1
   50 K2=K1+4
      IF (K2.GT.KTRM) K2=KTRM
      WRITE (NOUT,170) (LTM(L),L=K1,K2)
      DO 60 I=1,ITSP
      WRITE (NOUT,180) T(I),GX(K,I),(TRM(I,L),L=K1,K2)
   60 CONTINUE
      K1=K2+1
      IF (K1.LE.KTRM) GO TO 50
      DO 80 I=1,ITSP
      PCTDIF=0.
      IF (GX(K,I).LE.0.) GO TO 70
      PCTDIF=(GX(K,I)-FX(I))/GX(K,I)*100.
   70 CONTINUE
      WRITE (NOUT,190) I,T(I),GX(K,I),FX(I),PCTDIF
   80 CONTINUE
      IF (KKX.LE.0) RETURN
   90 CONTINUE
      READ (NIN,210) MLT
      IF (MLT.EQ.0) GO TO 110
      DO 100 MM=1,MLT
      READ (NIN,200) L,ALF(K,L),ALAM(K,L)
  100 CONTINUE
  110 CONTINUE
      READ (NIN,210) LT,(LTM(JJ),JJ=1,LT)
C
C     MLT=NO OF PARAMETERS TO BE CHANGED
C     LT = NUMBER OF TERMS TO BE REMOVED.
C     LTM(L) = TERM NOS. OF TERMS TO BE REMOVED.
C
      IF (LT.EQ.0) RETURN
      LTX=0
      LL=1
      KTRM=KTR(K)
      DO 130 L=1,KTRM
      IF (LTM(LL).EQ.L) GO TO 120
      LTX=LTX+1
      TRM(1,LTX)=ALF(K,L)
      TRM(2,LTX)=ALAM(K,L)
      GO TO 130
  120 CONTINUE
      LL=LL+1
  130 CONTINUE
      KTRM=LTX
      DO 140 L=1,KTRM
      ALF(K,L)=TRM(1,L)
      ALAM(K,L)=TRM(2,L)
  140 CONTINUE
      TL(1)=10HREVISED PA
      TL(2)=10HRAMETERS F
      KTR(K)=KTRM
      TL(3)=10HOR GROUP
      WRITE (NOUT,150) (TL(I),I=1,3),K
      WRITE (NOUT,160) (L,ALF(K,L),ALAM(K,L),L=1,KTRM)
      RETURN
C
  150 FORMAT (1H1,5X,3A10,I3)
  160 FORMAT (1H0,3H L=,I3,10H ALF(K,L)=,1PE11.4,11H ALAM(K,L)=,1PE11.4)
  170 FORMAT (1H0,25H     COOL TIME           GX,5I15)
  180 FORMAT (1P7E15.5)
  190 FORMAT (1H0,3H I=,I3,3H T=,1E12.5,4H GX=,1E12.5,4H FX=,1E12.5,8H P
     1CTDIF=,1E12.5)
  200 FORMAT (I6,2E12.5)
  210 FORMAT (12I6)
      END


      SUBROUTINE CORSBIN
C
C     ROUTINE FORMS COARSE GROUPS FROM FINE GROUP DATA IN ENDF-LIKE
C     FORMAT.
C
      COMMON /PULSIN/ AO(50), FX(200), T(401), KE, ITSP, IPROB, NIN,
     1 NOUT
      COMMON /PULSDAT/ NDF, EB(25), GB(25,400), NERG, EO(200), DUM(70),
     1 TOT(50)
      COMMON /ENDF/MEVU, MAT1, MF1, MT1, RUNTIM, NPUN
C     READ INPUT
      READ (NIN,90) MAT1,MF1,MT1,MEVU
C
C     MAT1=MAT NO. OF FISSIONING NUCLIDE DESIRED.
C     MF1=DATA TYPE DESIRED,MF1=80=F.P.DATA FOR THERMAL PULSE,MF1=81=
C         F.P. DATA FOR FAST PULSE,ETC.
C     MT1=TYPE OF F.P. WANTED,MT1=801=DATA FOR BETA- PLUS GAMMA,MT1=802=
C         DATA FOR GAMMA ONLY,MT1=803=DATA FOR BETA- ONLY,ETC.
C     MEVU=0,UNITS ARE MEV/FISS,  =1,UNITS ARE PARTICLE/FISS.
C     SEARCH ENDF TAPE FOR DESIRED DATA.
C
   10 CONTINUE
      READ (NDF,100) (AO(I),I=1,7),MAT,MF,MT,NSEQ
      IF (MF.EQ.-0) MF=80
      IF (MAT.EQ.-1) WRITE (NOUT,110) MAT,NDF
      IF (MAT.EQ.-1) STOP
      IF (MAT.LT.MAT1) GO TO 10
      IF (MAT.EQ.MAT1) GO TO 20
      WRITE (NOUT,110) MAT1,NDF
      STOP
   20 CONTINUE
      IF (MF.LT.MF1) GO TO 10
      IF (MF.EQ.MF1) GO TO 30
      WRITE (NOUT,120) MF1,NDF
      STOP
   30 CONTINUE
      IF (MT.LT.MT1) GO TO 10
      IF (MT.EQ.MT1) GO TO 40
      WRITE (NOUT,130) MT1,NDF
      STOP
   40 CONTINUE
      READ (NDF,140) NT
      READ (NDF,90) NICHT
      ITSP=NT
C
C     READ BROAD GROUP STRUCTURE
C
      READ (NIN,90) NE
      READ (NIN,150) (EB(N),N=1,NE)
C
C     NE=NO OF BROAD GROUPS PLUS ONE.
C     EB=ENERGY BOUNDS INCLUDING UPPER AND LOWER BOUNDS IN MEV
      NE=NE-1
      NERG=NE
      DO 70 IT=1,NT
      READ (NDF,160) T(IT),KE
      READ (NDF,90) NICHT
      READ (NDF,150) (E0(K),FX(K),K=1,KE)
      KE1=KE+1
      E0(KE1)=(E0(KE)-E0(KE-1))+E0(KE)
C
C     THE FOLLOWING LOOP SELECTS THE CORRECT UNITS.
      DO 50 K=1,KE
      IF (MEVU.EQ.1) FX(K)=FX(K)/((E0(K)+E0(K+1))/2.0)*1.E+4
      IF (MEVU.EQ.0) FX(K)=FX(K)*1.E+4
   50 CONTINUE
      CALL REBIN
      DO 60 IE=1,NE
      GB(IE,IT)=DUM(IE)
   60 CONTINUE
   70 CONTINUE
      DO 80 IE=1,NE
      WRITE (NOUT,170) IE,EB(IE),EB(IE+1)
      WRITE (NOUT,180) (IT,T(IT),GB(IE,IT),IT=1,NT)
      WRITE (NPUN,90) MAT1,MF1,MT1
      WRITE (NPUN,90) NE,NT
      WRITE (NPUN,190) IE,EB(IE),EB(IE+1)
      WRITE (NPUN,190) (IT,T(IT),GB(IE,IT),IT=1,NT)
   80 CONTINUE
      REWIND 10
      RETURN
C
   90 FORMAT (6I11)
  100 FORMAT (6A10,A6,I4,I2,I3,I5)
  110 FORMAT (1H1,15H  SORRY, MAT = ,I4,13H NOT ON TAPE ,I3)
  120 FORMAT (1H1,14H  SORRY, MF = ,I4,13H NOT ON TAPE ,I3)
  130 FORMAT (1H1,14H  SORRY, MT = ,I4,13H NOT ON TAPE ,I3)
  140 FORMAT (55X,I11)
  150 FORMAT (6E11.4)
  160 FORMAT (11X,1E11.4,33X,I11)
  170 FORMAT (1H0,17H  ENERGY BIN NO. ,I3,6H FROM ,1PE12.5,8H MEV TO ,1P
     1 E12.5,5H MEV.)
  180 FORMAT (1H ,13H TIME STEP = ,I3,16H COOLING TIME = ,1PE12.5,6H FX
     1= ,1PE12.5)
  190 FORMAT (2(I5,2E15.8))
      END


      SUBROUTINE REBIN
C
```

```
C     ROUTINE FOR BROAD GROUP BINNING.  BOUNDARIES DO NOT HAVE TO
C     COINCIDE WITH FINE GROUP BOUNDARIES.  WRITTEN BY GRAHAM
C     FOSTER,LASL,1975.
C     CHANGED OCT, 1977, LASL, D. GEORGE
C
      COMMON /PULSIN/ A0(50), Y(200), T(401), NX, ITSP, IPROB, NIN, NOUT
      COMMON /PULSDAT/ NDT, U(25), GB(25,400), NU, X(200), V(70), TOT(50
     1 )
      SUM1=SUM2=0.
      NU1=NU+1
      NX1=NX+1
      DO 10 IU=1,NU1
   10 V(IU)=0.
      DO 20 IX=1,NX
      IF (X(IX).LT.U(1)) GO TO 20
      IF (X(IX).GT.U(NU1)) GO TO 20
      SUM1=SUM1+Y(IX)
   20 CONTINUE
C  FIND THE FIRST BIN
      DO 30 IX=1,NX
      IF (X(IX).EQ.U(1)) GO TO 50
   30 IF (X(IX).GT.U(1)) GO TO 40
      WRITE (NOUT,110)
      RETURN
   40 V(1)=Y(IX-1)*(X(IX)-U(1))/(X(IX)-X(IX-1))
   50 IX=IX+1
      DO 80 IU=1,NU
   60 IF (X(IX).GT.(U(IU+1)*1.0001)) GO TO 70
      V(IU)=V(IU)+Y(IX-1)
      IF (IX.GT.NX) GO TO 90
      IX=IX+1
      GO TO 60
   70 CRIN=Y(IX-1)*(U(IU+1)-X(IX-1))/(X(IX)-X(IX-1))
      V(IU)=V(IU)+CRIN
      Y(IX-1)=Y(IX-1)-CRIN
   80 CONTINUE
   90 IF (X(NX1).GT.U(NU1)) WRITE (NOUT,120) X(NX1),U(NU1)
      DO 100 IU=1,NU
  100 SUM2=SUM2+V(IU)
      ERRD=0.001
      IF (ABS(SUM2-SUM1).GT.ERRD*SUM1) WRITE (NOUT,130) SUM1,SUM2
C     IF (SUM2.GT.SUM1) WRITE (NOUT,101) (I,X(I),Y(I),I=1,NX)
C     IF (SUM2.GT.SUM1) WRITE (NOUT,102) (N,U(N),V(N),N=1,NU)
      RETURN
C
  110 FORMAT (29H CANT FIND FIRST ENERGY BOUND)
  120 FORMAT (17H0***** LAST DATUME10.3,27H EXTENDS BEYOND END OF GRIDE1
     1 0.3,6H *****/)
  130 FORMAT (32H0***** INTEGRAL BEFORE REBINNINGE10.3,41H DOES NOT EQUA
     1L INTEGRAL AFTER REBINNING E10.3,6H *****/)
      END


      SUBROUTINE RUNTOTS
C
C     ROUTINE TO READ TOTALS,I.E.,GX(K,I) SUMMED OVER ENERGY.
C
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN, NOUT
      COMMON /PULSDAT/ NDT, EB(25), GX(25,400), NERG, ALF(25,50), ALAM(2
     1 5,50)
      COMMON /PULSOUT/ ALPHA(50), FXC(100), PCT(100)
      COMMON /FINRAD/ NIAPL, IRAD, NCORS, RADT(200), DELT(200)
C
C     THIS SUBROUTINE IS USED FOR READING EXPERIMENTAL DATA.
C     UNFORTUNATELY, THIS CAN BE RECEIVED IN A VARIETY OF FORMATS,SO
C     THIS ROUTINE MUST BE CONTINUALLY CHANGED TO ACCOMADATE WHATEVER
C     FORMAT THS DATA IS IN.  THE FUNNY LOOKING STATEMENTS BELOW ARE
C     FOR READING SOME ORNL DATA IN THE WHITTEMORE JUNK FORMAT.
C
      DIMENSION HDR(8)
      READ (NIN,30) NP,NE,NHD
      DO 10 L=1,NHD
      READ (NIN,50) (HDR(I),I=1,8)
      WRITE (NOUT,50) (HDR(I),I=1,8)
   10 CONTINUE
      DO 20 I=1,NP
      IF (I.LE.NE) EB(I)=0.0
      READ (NIN,40) (GX(K,I),K=10,15)
      WRITE (NOUT,40) (GX(K,I),K=10,15)
      RADT(I)=GX(10,I)
      T(I)=GX(11,I)+GX(12,I)/2.0
      DELT(I)=1.0
      GX(1,I)=(GX(13,I)+GX(14,I))*GX(10,I)/GX(12,I)
   20 CONTINUE
      ITSP=NP
      NE=1
      NERG=1
      RETURN
C
   30 FORMAT (12I6)
   40 FORMAT (6E12.5)
   50 FORMAT (8A10)
      END


      SUBROUTINE OVRLAY2
C
C     OVRLAY2 IS ROUTINE FOR FITTING OPTIONS 1 AND 2
C
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN, NOUT
      COMMON /PULSDAT/ NDT, EB(25), GX(25,400), NERG, ALF(25,50), ALAM(2
     1 5,50)
      LEVEL 2, A, B
      COMMON /PULSCAL/ A(50,50), B(50,1)
      COMMON /PULSOUT/ ALPHA(50), FXC(100), PCT(100)
      COMMON /MANI/ WX(100), TITL(8), KTR(50), NS(10), KKN(25), DIFLIM
      COMMON /FINRAD/ NIAPL, IRAD, NCORS, RADT(200), DELT(200)
      COMMON /COMO/ NP, TMIN, TMAX, NSTEP, GXMIN, TC(100), LXX(20),
     1 NFINL
      COMMON /CLWT/ LWTR
      LWTR=0
      DO 70 K=1,NERG
      IF (KKN(K).NE.K) GO TO 70
C
C     THIS PORTION OF ROUTINE ALLOWS FIT IN SEVERAL SEGMENTS.  TO
C     ACTIVATE, SET IPROB NEGATIVE.  NOTE - INPUT NEEDED FOR EA. GROUP.
C
C     NSEG = NUMBER OF SEGMENTS + 1
C     NS = BREAKPOINTS OF SEGMENTS.
C
      NSEG=2
      NS(1)=1
      IF (IPROB.LT.0) READ (NIN,80) NSEG,(NS(LX),LX=2,NSEG)
C
      IF (NSEG.LE.2) NS(2)=NP
      NSEG1=NSEG-1
      LTRM=0
      KTR(K)=0
      DO 40 N=1,NSEG1
      N1=NS(N)
      N2=NS(N+1)
      ITP=N2-N1+1
      ITSP=0
      IX=0
      IF (TMIN.LT.TC(1)) TMIN=TC(1)
      IF (TMAX.LE.0.0) TMAX=TC(NP)
      IF (TMAX.GT.TC(NP)) TMAX=TC(NP)
      DO 10 I=1,ITP
      NN=N1+I-1
      IF (GX(K,NN).LT.GXMIN) GO TO 10
      IF (TC(NN).LT.TMIN) GO TO 10
      IF (TC(NN).GT.TMAX) GO TO 10
      IX=IX+1
      ITSP=IX
      LXX(K)=IX
      FX(IX)=GX(K,NN)
      T(IX)=TC(NN)
   10 CONTINUE
      CALL PULSFIT (K)
      IF (N.NE.NSEG1) KTRM=KTRM-1
      IX=0
      DO 20 I=1,ITP
      NN=N1+I-1
      IF (GX(K,NN).LT.GXMIN) GO TO 20
      IF (TC(NN).LT.TMIN) GO TO 20
      IF (TC(NN).GT.TMAX) GO TO 20
      IX=IX+1
      ITSP=IX
      LXX(K)=IX
      FX(IX)=GX(K,NN)
      T(IX)=TC(NN)
   20 CONTINUE
      IF (NIAPL.EQ.1) CALL SEEFIT (K)
      DO 30 J=1,KTRM
      LTRM=LTRM+1
      ALF(K,LTRM)=B(J,1)
      ALAM(K,LTRM)=ALAMDA(J)
   30 CONTINUE
      KTR(K)=KTR(K)+KTRM
   40 CONTINUE
      KTRM=KTR(K)
      DO 50 J=1,KTRM
      ALAMDA(J)=ALAM(K,J)
      B(J,1)=ALF(K,J)
   50 CONTINUE
      WRITE (NOUT,90) K,EB(K),EB(K+1)
      DO 60 J=1,KTRM
      WRITE (NOUT,100) J,ALAMDA(J),B(J,1)
   60 CONTINUE
   70 CONTINUE
      RETURN
C
   80 FORMAT (12I6)
   90 FORMAT (1H1,24H  RESULTS FOR GROUP NO. ,I3,11H E-LOWER = ,1PE12.5,
     1 17H MEV.  E-UPPER = ,1PE12.5,5H MEV.)
  100 FORMAT (1H0,4H J =,I3,9H ALAMDA =,1PE12.5,4H B =,1PE12.5)
      END


      SUBROUTINE PULSFIT (KKF)
C
C     THIS ROUTINE PROVIDES 1 AND 2 PARAMETER FITS FOR A COMBINATION
C     OF LINEAR FUNCTIONS OF THE FORM --
C
C         FX(I)=SUM+ALPHA(K)*EXP(-ALAMDA(K)*T(I)),SUM OVER K=1,KTRM,
C                   THE NUMBER OF TERMS USED TO REPRESENT FX.
C
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN, NOUT
      LEVEL 2, A, B
      COMMON /PULSCAL/ A(50,50), B(50,1)
      COMMON /PULSOUT/ ALPHA(50), FXC(100), PCT(100)
      COMMON /MANI / W(100), TITL(8), KTR(50), NS(10), KKN(25), DIFLIM
      COMMON /FINRAD/ NIAPL, IRAD, NCORS, RADT(200), DELT(200)
      COMMON /CLWT/ LWTR
      DIMENSION KCAL(71), IPVT(50)
      ITS=1
      IPRO=KKF
      IF (NIAPL.EQ.1) GO TO 10
      IF (NIAPL.EQ.0) RETURN
C     INPUT TO THIS SUBROUTINE NEEDED ONLY IF NIAPL=-1.
      READ (NIN,220) LWT,KTRM,IPRT
      READ (NIN,230) (ALAMDA(K),K=1,KTRM)
C
C     LWT=WT FCN DESIRED,=0,W=1,=1,W=1/FX,=2,W=1/FX**2,=3,W=1/FX**1.5.
C     KTRM=NO. OF ALAMDAS USED IN FIT.
C     IPRT=1,PRINT A-MATRIX,=0,NO PRINT.
C     ALAMDA(K) = LAMBDAS PROVIDED BY USER, AS THIS OPTION ONLY FITS
C               COEFFICIENTS.
      GO TO 50
   10 CONTINUE
      I1=ITSP
      KK=1
   20 I2=I1-1
C     IF ((FX(I2)/FX(I1)).LT.5.0) I2=I1-2
      ALAMDA(KK)=ALOG(FX(I2)/FX(I1))/(T(I1)-T(I2))
      B(KK,1)=FX(I1)/EXP(-ALAMDA(KK)*T(I1))
      IT1=I2-1
      FT1=FX(IT1)-(B(KK,1)*EXP(-ALAMDA(KK)*T(IT1)))
      IF (-FT1.LT.FX(IT1)) GO TO 30
      IF (KGFL.NE.IPRO) GO TO 30
      I1=I2
      GO TO 20
   30 CONTINUE
      I3=I2-1
      IFLG=0
      DO 40 I=1,I3
      FT=FX(I)
      FX(I)=FX(I)-(B(KK,1)*EXP(-ALAMDA(KK)*T(I)))
      ATST=FX(I)/FT
      IF (ATST.GT.0.05) GO TO 40
      IFLG=IFLG+1
      WRITE (NOUT,180) I,FT,FX(I)
   40 CONTINUE
      I1=I3-IFLG
      KK=KK+1
      IF (I1.GE.2) GO TO 20
      KTRM=KK-1
      RETURN
   50 CONTINUE
C
C     WRITE OUT INPUT FOR DEBUG
      WRITE (NOUT,240) (K,ALAMDA(K),K=1,KTRM)
      WRITE (NOUT,250) (I,T(I),FX(I),I=1,ITSP)
      DO 60 I=1,ITSP
      W(I)=1./FX(I)**2
      IF (LWT.EQ.3) W(I)=1./FX(I)**1.5
      IF (LWT.EQ.1) W(I)=1./FX(I)
      IF (LWT.LE.0) W(I)=1.
   60 CONTINUE
      WRITE (NOUT,260) LWT
C
C    REORDER ALAMDAS HI TO LO
C
      CALL QQSORT (KTRM,ALAMDA,PCT(1),PCT(50),PCT(50))
      NTRM=KTRM/2
      DO 70 K=1,NTRM
      TEMP=ALAMDA(K)
      ALAMDA(K)=ALAMDA(KTRM-K+1)
   70 ALAMDA(KTRM-K+1)=TEMP
C
C     CALCULATE B MATRIX
C
      DO 100 K=1,KTRM
      B(K,1)=0.
      DO 90 I=1,ITSP
      IF (IRAD.LE.0) GO TO 80
      XP1=EXP(-ALAMDA(K)*RADT(I))
      XP2=EXP(-ALAMDA(K)*DELT(I))
      XP3=EXP(-ALAMDA(K)*(T(I)-DELT(I)/2.))
      B(K,1)=B(K,1)+FX(I)*1./DELT(I)/ALAMDA(K)**2*(1.-XP1)*(1.-XP2)*XP3
     1 *W(I)
      GO TO 90
   80 CONTINUE
      B(K,1)=B(K,1)+FX(I)*EXP(-ALAMDA(K)*T(I))*W(I)
   90 CONTINUE
  100 CONTINUE
C
C     CALCULATE A MATRIX
C
      DO 110 I=1,KTRM
      DO 110 J=1,KTRM
      A(I,J)=0.
  110 CONTINUE
      DO 150 K=1,KTRM
      ALAM=ALAMDA(K)
      DO 140 L=1,KTRM
      DO 130 I=1,ITSP
      IF (IRAD.LE.0) GO TO 120
      XP01=1.-EXP(-ALAMDA(K)*RADT(I ))
      XP02=1.-EXP(-ALAMDA(L)*RADT(I ))
      XP03=1.-EXP(-ALAMDA(K)*DELT(I ))
      XP04=i.-EXP(-ALAMDA(L)*DELT(I ))
      XP05=EXP(-ALAMDA(K)*(T(I )-DELT(I )/2.))
      XP06=EXP(-ALAMDA(L)*(T(I )-DELT(I )/2.))
      COFF=1./DELT(I )/ALAMDA(K)**2
      COFT=1./DELT(I )/ALAMDA(L)**2
      A(K,L)=A(K,L)+COFF*COFT*XP01*XP02*XP03*XP04*XP05*XP06*W(I )
      GO TO 130
  120 CONTINUE
      A(K,L)=EXP(-(ALAM+ALAMDA(L))*T(I ))*W(I )+A(K,L)
  130 CONTINUE
  140 CONTINUE
  150 CONTINUE
C
C     PRINT A MATRIX
C
      IF (IPRT.LT.1) GO TO 170
      WRITE (NOUT,270) IPRO
      DO 160 K=1,KTRM
      WRITE (NOUT,280) (A(K,L),L=1,KTRM)
  160 CONTINUE
  170 CONTINUE
      WRITE (NOUT,290) IPRO
      WRITE (NOUT,280) (B(K,1),K=1,KTRM)
C  SUBROUTINE LCMLSS SOLVES THE SET OF LINEAR EQUATIONS  AX=B
      CALL LCMLSS (KTRM,1,A,50,B,50,IPVT,DET,PCT(1),PCT(50))
      WRITE (NOUT,290) IPRO
      WRITE (NOUT,280) (B(K,1),K=1,KTRM)
      CALL PDIF
      WRITE (NOUT,190) IPRO
      WRITE (NOUT,200) (I ,T(I ),FX(I ),FXC(I ),PCT(I ),I=1,ITSP)
      RETURN
      WRITE (NOUT,210)
      STOP
C
  180 FORMAT (1H0,14H  POINT AT I  =,I3,14H ON SLOPE FT =,E12.6,5H FX =
     1 ,E12.6)
  190 FORMAT (109H1 STEP NO.      TIME                 ORIGINAL VALUE     COMPUT
     1ED  VALUE   PERCENT DIFFERENCE                    IPROB= ,I3)
  200 FORMAT (I6,1P4E18.5)
  210 FORMAT (16H SINGULAR SYSTEM)
  220 FORMAT (12I6)
  230 FORMAT (3(11X,E11.4))
  240 FORMAT (3H K=,I3,11H ALAMDA(K)=,1PE12.5)
  250 FORMAT (4H  I=,I3,6H T(I )=,1PE12.5,7H FX(I )=,1PE12.5)
  260 FORMAT (1H0,6H  LWT=,I3)
  270 FORMAT (16H0  A-MATRIX FOR ,I6)
  280 FORMAT (2X,10E12.3)
  290 FORMAT (15H0  B-MATRIX FOR,I6)
      END


      SUBROUTINE PDIF
C
C     ROUTINE CALCULATES PER CENT DIFFERENCES BETWEEN CALCULATED AND INP
C     DATA.
C
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN, NOOT
      LEVEL 2, A, B
      COMMON /PULSCAL/ A(50,50), B(50,1)
      COMMON /PULSOUT/ ALPHA(50), FXC(100), PCTDIF(100)
      COMMON /FINRAD/ NIAPL, IRAD, NCORS, RADT(200), DELT(200)
      DO 30 I=1,ITSP
      PCTDIF(I )=0.
      FXC(I )=0.
      DO 20 K=1,KTRM
      IF ((-ALAMDA(K)*T(I )).GT.300.) ALAMDA(K)=-300.0/T(I )
      IF (IRAD.LE.0) GO TO 10
      COFF=B(K)/DELT(I)/ALAMDA(K)**2
      XP01=1.-EXP(-ALAMDA(K)*RADT(I))
      XP02=1.-EXP(-ALAMDA(K)*DELT(I))
      XP03=EXP(-ALAMDA(K)*(T(I)-DELT(I)/2.))
      FXC(I)=FXC(I)+COFF*XP01*XP02*XP03
      GO TO 20
   10 CONTINUE
      FXC(I)=FXC(I)+B(K)*EXP(-ALAMDA(K)*T(I))
      ALPHA(K)=B(K)*EXP(-ALAMDA(K)*T(I))
   20 CONTINUE
   30 PCTDIF(I)=(FX(I)-FXC(I))/FX(I)*100.
      RETURN
      END


      SUBROUTINE SEEFIT (IP)
C
C     ROUTINE DOES INTERACTIVE GRAPHICS.
C
      COMMON /PULSIN/ ALAMDA(50), FX(200), T(401), KTRM, ITSP, IPROB,
     1 NIN, NOOT
      LEVEL 2, A, B
      COMMON /PULSCAL/ A(50,50), B(50,1)
      COMMON /PULSOUT/ ALPHA(50), FXC(100), PCTDIF(100)
      COMMON /FINRAD/ NIAPL, IRAD, NCORS, RADT(200), DELT(200)
      DIMENSION TI(10), XL(10), YL(10)
      DIMENSION FNG(7,20), TNG(7,20), NS(20), YP(20), XP(20)
      NOUT=20
      M=1
      N=0
      DO 10 JJX=1,5
      NS(JJX)=1
   10 CONTINUE
      CALL PDIF
      DO 40 I=1,ITSP
      IF (ABS(PCTDIF(I)).LT.10.) GO TO 40
      DIFF=FX(I)-FXC(I)
      TXT=ABS(PCTDIF(I))
C     IF (TXT.LT.20) GO TO 220
      IF (N.LT.20) N=N+1
      NS(M)=N
      IF (N.LE.1) GO TO 30
      IF (II.EQ.(I-1)) GO TO 30
      IF (N.LE.3) GO TO 20
      M=M+1
   20 N=1
   30 CONTINUE
      WRITE (NOUT,190) IP,I,PCTDIF(I),DIFF
      II=I
      TNG(M,N)=T(I)
      FNG(M,N)=DIFF
      MM=M
   40 CONTINUE
      IF (MM.EQ.0) GO TO 140
      MJM=KTRM
      DO 130 M=1,MM
      NN=NS(M)-1
      WRITE (NOUT,200) (M,N,TNG(M,N),FNG(M,N),N=1,NN)
      IF (NN.LE.2) GO TO 130
C FIND MINS AND MAXS
   50 FNGMX=FNG(M,1)
      TMX=TNG(M,1)
      FNGMN=FNG(M,1)
      TMN=TNG(M,1)
      DO 60 N=1,NN
      IF (TNG(M,N).GT.TMX) TMX=TNG(M,N)
      IF (TNG(M,N).LT.TMN) TMN=TNG(M,N)
      IF (FNG(M,N).GT.FNGMX) FNGMX=FNG(M,N)
      IF (FNG(M,N).LT.FNGMN) FNGMN=FNG(M,N)
      XP(N)=TNG(M,N)
      YP(N)=FNG(M,N)
   60 CONTINUE
C DRAW PICTURE
```

Printed in the United States of America. Available from
National Technical Information Service
US Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

Microfiche $3.00

001-025	4.00	126-150	7.25	251-275	10.75	376-400	13.00	501-525	15.25
026-050	4.50	151-175	8.00	276-300	11.00	401-425	13.25	526-550	15.50
051-075	5.25	176-200	9.00	301-325	11.75	426-450	14.00	551-575	16.25
076-100	6.00	201-225	9.25	326-350	12.00	451-475	14.50	576-600	16.50
101-125	6.50	226-250	9.50	351-375	12.50	476-500	15.00	601-up	

Note: Add $2.50 for each additional 100-page increment from 601 pages up.