AUG 9 1967

TANDEM: A Program for the Two-Fold Least-Squares
Analysis of Capture and Fission Cross Sections with
the Multilevel Expansion.

D. B. ADLER

Department of Physics, University of Illinois

Technical Report No. 5

under contract AEC 1546

Principal Investigator

F. T. Adler

Department of Physics, University of Illinois

Urbana, Illinois

March 1967

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of 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. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

COO 1546-5

CPSTI PRICES

H.C. \$ 3.00, MN .65

TANDEM: A Program for the Two-Fold Least-Squares
Analysis of Capture and Fission Cross Sections with
the Multilevel Expansion.

D. B. ADLER

Department of Physics, University of Illinois

Technical Report No. 5

under contract AEC 1546

Principal Investigator

F. T. Adler

Department of Physics, University of Illinois

Urbana, Illinois

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

March 1967

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

INDEX

I	Introduction	2
II	Method of Analysis	5
	A) Definitions	5
	B) Discussion of the iterative procedures.....	10
III	Least squares equations for the twofold curve fitting of the cross sections.....	14
IV	Description of the program TANDEM and of the associated FORTRAN subroutines.....	19
IV-1	Main Program	19
IV-2	Subroutine WETA	20
IV-3	Subroutine SIGTWO	21
IV-4	Subroutine TEST	22
IV-5	Subroutine SIGPLT	23
IV-6	Subroutine PHIDO	24
IV-7	Function INVERT	24
IV-8	Subroutine SIGOUT	24
IV-9	Subroutine SIGIN	25
IV-10	Subroutine SIGAP	25
IV-11	Subroutine WIDTH	25
References		
Appendix A: Input and Output of TANDEM		27
Appendix B: Listing of TANDEM		32

ABSTRACT

Program TANDEM has been developed for the simultaneous fitting of capture and fission cross sections in the fissile elements, using the multilevel expansion. The version presented here is written in FORTRAN II and is operable on the IBM 7094 of the University of Illinois. This program consists of a combination of subroutines which at a later stage could be utilized to include the fitting of transmission data. This generalization of the program could be best accomplished using a larger computer such as an IBM 360 system.

Non-linear iterations on the values of the resonance energies and half-widths are executed by the program subject to flexible numerical tests and constraints which are introduced to enhance the convergence of the fitting process. The present formulation of the constraints attempts to utilize both the physical properties of the parameters which are involved in the fitting process, and the numerical experience gained with the preliminary use of the code.

I. Introduction

In the least-squares analysis of low-energy neutron cross sections, it is very useful to have access to a method for the automatized search of those parameters which appear as nonlinear variables in the trial function. For the analysis of reaction cross sections the nonlinear parameters are the level parameters defining the resonance energy and the half-width, while for the analysis of transmission measurements all the resonance parameters enter the trial function in a nonlinear form.

In all of these cases the search is based on a suitably linearized trial function which is assumed to approximate the original trial function. The method generally used is the classical Gauss method of linearization.^{1,2,3/} It is known that this method does not guarantee convergence, unless additional constraints, or tests, are incorporated in the searching routines.^{4/} More powerful methods exist, based on automatized minimization techniques^{5,6/}, which could be best utilized for this purpose on computers having large fast memory storages (larger than the one available with the IBM 7094).

Within the Gauss method, however, it is possible to attain more rapid convergence by combining the nonlinear iterations with constraints, and tests, which are based on the inherent physical properties of the quantities involved in the fitting. A natural constraint for the level parameters becomes available when rather than one cross section two sets of cross section data (for the same nuclide) are fitted jointly in the same energy range.

Simultaneous fitting of two or three sets of data using single level expansions has been obtained recently by Drawbaugh and Gibson ^{2/}.

With their code, however, the number of fitted resonances remains limited, in order to accommodate large storages for the data points. Also, extrapolated cross section values are introduced, in order to have a single set of energy points, for those cases in which sets of independently measured data are jointly fitted.

With a multilevel expansion, where the interferences between levels are taken into account, it is essential to fit together large numbers of resonances. Besides, in view of the high accuracy attainable with the multilevel analysis it is desirable to handle the data without previous extrapolating procedures, and therefore have available large memory storages.

For these reasons, the version of the program TANDEM presented in this report has been limited to the case of simultaneous analysis of capture and fission data, measured within the same experiment, using multilevel formulae.

TANDEM is a FORTRAN II code, using the same analytical method described in a previous report^{3/}. For the nonlinear fitting numerical tests have been included, which reduce the "hunting" for the nonlinear parameters, and therefore ensure a more rapid convergence than the one obtainable from the program CODILLI^{3/}. The "single level" option existing in CODILLI has not been included in TANDEM, since single level codes have appeared recently in the literature.^{2/} This option can, however, be inserted readily in the program.

The generalization to two sets of data obtained from different measurements (these could also be two independent measurements of the same reaction cross section, for the same nuclide) formally is straightforward. In this case the two sets of data would be defined on different

sets of data points, with differently defined statistical errors, different resolution functions, and possibly different temperatures.

For a threefold analysis, inclusive of transmission data, the required storage space would increase, and the convergency tests should be adjusted to the problem of properly handling the basically nonlinear cases of transmission measurements with large sample thickness. While the formal aspects of this generalization offer no difficulty, practically it appears possible to pursue this task only with computers having a larger fast memory than the IBM 7094. The imminent installation of an IBM 360 system at Illinois would make it feasible to obtain this goal.

In view of this generalization, the parts of the calculation in which are included the specific physical characteristics of the data, reflecting the experimental conditions, are coded separately, in the subroutines WETA (for the statistical variances) and WIDTH (for the resolution function). In the version presented here, these subroutines refer to the U-235 measurements of deSaussure and Weston at ORNL-RPI.^{7/} The switch to different prescriptions, for a different measurement, represents a very limited coding effort, consisting in replacing, or modifying, the subroutines WIDTH and WETA.

In Section II the method of analysis is outlined and in Section III the least-squares procedure is discussed. The program and associated subroutines are described in Section IV.

Appendix A contains the description of Input and Output for TANDEM, and Appendix B contains the listing of the program.

After completion of this work we have become aware of similar work being developed at ORNL, by Dr. G. deSaussure. We are indebted to Dr. deSaussure for this communication, and for stimulating discussions during the Summer 1966 at Oak Ridge. Informative discussions with Mr. W. Nestor, Jr., of Oak Ridge National Laboratory on nonlinear regression analysis are also gratefully acknowledged.

II. Method of Analysis

A) Definitions

The method of analysis used in the Program TANDEM is based on the same multilevel expansions and on the same definitions which have been employed in the Program CODILLI^{3/}. The basic differences between the two programs consist in the fact that with TANDEM it is possible to analyze two sets of cross section data simultaneously, although at present only reaction cross sections have been accommodated. The present version of TANDEM is set up to analyze data obtained within the same experimental conditions, and measured at the same set of energy points. The same set of level parameters, μ_k , ν_k is assumed to be shared by either cross section, and the same type of multilevel expansion is taken to describe either fission or capture reaction rate:

$$\sqrt{E} \sigma^{(x)}(E) = C \sum_{k=1}^{NG} X_k^{(x)} \phi_k(E; T) + \sqrt{E} \sigma_{Dis}^{(x)}(E) \quad (1)$$

where (x) characterizes the type of reaction cross section. Here σ_{Dis} represents the contribution of resonances having known parameters, which can be either external or internal to the interval of energy in which

the data are analyzed: denoting with $\bar{X}_k$ the known parameters, and with NOUT the number of the "known" resonances, one defines

$$\sqrt{E} \sigma_{\text{Dis}}^{(x)}(E) = C \sum_{j=1}^{2\text{NOUT}} \bar{X}_j^{(x)} \phi_j(E;T) \quad (2)$$

where the constant $C = 6.52 \times 10^5$ barns $\times$ eV.

In equation (1), $NG = 2\text{NRT} + \text{KNEG} + \text{KPOS}$, where $\text{NRT} = \text{NR} + \text{NEIGH}$ represents the total number of fitted resonances, being NR the number of those which lie inside and NEIGH the number of those which are "guessed" as being located in the vicinity of the energy range in question.^{3/} KNEG and KPOS define the number of additional background terms of the form E^i , with the integer $i \leq 0$ for KNEG, and $i > 0$ for KPOS.^{3/}

The functions $\phi_i(E;T)$ appearing in equations (1) and (2) are known functions of E and of the parameter T. These functions are generalized line shape functions, and in the expansions (1) and (2) can be interpreted either as convolutions with the Doppler broadening kernel, or as double convolutions with resolution and Doppler broadening kernels. It is assumed here that the resolution broadening be of Gaussian form, so that in both cases $\phi_k(E;T)$ is expressed by either a ψ or a X function, and the distinction between the two cases is made with the definition of the parameter T.^{3/}

In the following, T will indicate pure Doppler broadening, while T_R will represent the combined effect of temperature and resolution broadening; accordingly, $\sigma_c(E)$ is used to represent the temperature broadened cross section, while $\sigma_{\text{ap}}(E)$ represents the cross section evaluated with inclusion of temperature and resolution effects.^{3/} This is the "apparent" cross section used in the fitting of the observed cross section

data. The set of functions ϕ_k , and the corresponding parameters X_k , for one of the two cross sections, are given below:

$$\begin{array}{lll}
 \phi_k = \psi_k & X_k = GR_k = G_k/v_k & k = 1, NR \\
 \phi_k = \frac{1}{(\mu_j - E)^2 + v_j^2} & X_k = GR_j = G_j v_j & \begin{cases} j = 1, \text{ NEIGH and} \\ k = NR + j \end{cases} \\
 \phi_k = X_j & X_k = GI_j = H_j/v_j & \begin{cases} j = 1, NR \text{ and} \\ k = NRT + j \end{cases} \\
 \phi_k = \frac{\mu_j - E}{(\mu_j - E)^2 + v_j^2} & X_k = GI_j = H_j & \begin{cases} j = 1, \text{ NEIGH and} \\ k = NRT + NR + j \end{cases} \\
 \phi_k = E^{1-j} & X_k = A_j & \begin{cases} j = 1, \text{ KNEG and} \\ k = 2 \text{ NRT} + j \end{cases} \\
 \phi_k = E^j & X_k = B_j & \begin{cases} j = 1, \text{ KPOS and} \\ k = 2 \text{ NRT} + \text{KNEG} + j \end{cases}
 \end{array} \quad (3)$$

For all those cases in which the level parameters μ_k , v_k , are kept fixed, the trial function for each cross section, given by

$$\sqrt{E} \sigma_{ap}^{(x)}(E) = C \sum_{k=1}^{NG} X_k^{(x)} \phi_k(E; T_R) + \sqrt{E} \sigma_{Dis}^{(x)}(E) \quad (4)$$

contains the unknown parameters $X_k^{(x)}$ in a linear manner. Thus, the two cross section expansions are formally connected only through the vector $\vec{\Phi}$, whose components are shared by either expansion. There are two independent sets of unknown parameters $X_k^{(x)}$. In this case the total number of unknowns is $NX = 2NG$. The sequence chosen for the purpose of fitting simultaneously both cross sections is here $k = 1, NG$ for fission [$(x) = (1)$] and $k = NG+1, 2NG$ for capture [$(x) = (2)$].

For the cases in which also the level parameters μ_k, v_k are refitted, a formal connection is instead established between the two sets of parameters. In this case, referred to as nonlinear case, one assumes that a better set of level parameters μ, v can be obtained by evaluating the changes of the trial function when one allows μ and v to vary by a small amount, $\delta\mu, \delta v$ respectively, so that in the trial function the resulting change is assumed to be linear. Indicating now with the index (o) "known" parameters, as those derived from a previous fitting, and known functions $\phi_k^{(o)}$ as evaluated using previously known values of μ_k, v_k , in this case the expansion (4) is replaced by: ^{3/}

$$\begin{aligned} \sqrt{E} \sigma_{ap}^{(x)}(E) = & C \sum_{k=1}^{NG} X_k^{(x)} \phi_k^{(o)}(E; T_R) + C \sum_{k=1}^{NR} X_k^{(x)(o)} \left(\frac{\partial \phi_k}{\partial \mu_k} \delta\mu_k + \frac{\partial \phi_k}{\partial v_k} \delta v_k \right) \\ & + \sqrt{E} \sigma_{Dis}^{(x)}(E) . \end{aligned} \quad (5)$$

This assumption implies that the line function $\phi_k(E; T_R)$ slightly changes shape, in consequence of the changes $\delta\mu_k, \delta v_k$. Thus, the contribution of the k-th resonance to $\sigma_{ap}^{(x)}$, given by

$$\begin{aligned} X_k^{(x)} \phi_k^{(o)}(E; T_R) + X_{kk}^{(x)} \phi_{kk}^{(o)}(E; T_R) + \left(X_k^{(x)(o)} \frac{\partial \phi_k}{\partial \mu_k} \Big|_o + X_{kk}^{(x)(o)} \frac{\partial \phi_{kk}}{\partial \mu_k} \Big|_o \right) \delta\mu_k \\ + \left(X_k^{(x)(o)} \frac{\partial \phi_k}{\partial v_k} \Big|_o + X_{kk}^{(x)(o)} \frac{\partial \phi_{kk}}{\partial v_k} \Big|_o \right) \delta v_k \end{aligned} \quad (6)$$

where $k = 1, NR$, $kk = NRT + k$, and where $\Big|_o$ indicates functions evaluated at μ_k^o, v_k^o , has to approximate the contributions of a pair of modified line shape functions, ϕ_k, ϕ_{kk} , evaluated at $\mu_k = \mu_k^o + \delta\mu_k$,

$v_k = v_k^0 + \delta v_k$ with parameters $X_k^{(x)}$, $X_{kk}^{(x)}$.

In this manner to the array ϕ_k , defined in the equations (3), for each cross section (x) is added a new array (with 2NR components), defined by

$$\phi_j^{(x)} = X_k^{(x)}(o) \left. \frac{\partial \phi_k}{\partial \mu_k} \right|_o + X_{kk}^{(x)}(o) \left. \frac{\partial \phi_{kk}}{\partial \mu_k} \right|_o \quad (7)$$

$$\phi_{NR+j}^{(x)} = X_k^{(x)}(o) \left. \frac{\partial \phi_k}{\partial v_k} \right|_o + X_{kk}^{(x)}(o) \left. \frac{\partial \phi_{kk}}{\partial v_k} \right|_o \quad (8)$$

where $k = 1, NR$, $kk = NRT + k$, and $j = 2NG + 1, 2NG + NR$ for $(x) = 1$, while $j = 2NG + 2NR + 1, 2NG + 3NR$ for $(x) = 2$.

Thus, introducing the array of additional unknowns

$$X_\ell = \delta \mu_k \text{ with } \ell = 2NG + k \quad (9)$$

$$\text{and } X_{NR+\ell} = \delta v_k$$

the expansion of equation (6) is written as

$$X_k^{(x)} \phi_k^{(o)} + X_{kk}^{(x)} \phi_{kk}^{(o)} + X_\ell \phi_j^{(x)} + X_{NR+\ell} \phi_{NR+j}^{(x)} \quad (6')$$

The linearized form of the trial function will thus contain

$$NX = 2NG + 2NR$$

unknowns, while the total number of the ϕ components is now

$$NTOT = 2NG + 4NR.$$

Explicitely the trial function expansions, which are now interrelated, are given by:

$$\sqrt{E} \sigma_{ap}^{(1)}(E) = C \sum_{j=1}^{NG} X_j \phi_j^{(o)}(E; T_R) + C \sum_{j=2NG+1}^{NX} X_j \phi_j^{(o)}(E; T_R) + \sqrt{E} \sigma_{Dis}^{(1)}(E) \quad (10)$$

$$\sqrt{E} \sigma_{ap}^{(2)}(E) = C \sum_{j=NG+1}^{2NG} X_j \phi_j^{(o)}(E; T_R) + C \sum_{j=2NG+1}^{NX} X_j \phi_{2NR+j}^{(o)}(E; T_R) + \sqrt{E} \sigma_{Dis}^{(2)}(E) \quad (11)$$

where (o) indicates functions evaluated at μ_k^o, v_k^o . Similarly to the procedure utilized in CODILLI ^{3/} the non-linear analysis is applied only to the set of NR resonances, internal to the energy interval analysed. The same procedure, applied to the NEIGH terms, can produce numerical instabilities. Investigations on the negative energy levels have been previously done ^{8/} by parametric studies of the NEIGH levels. An alternate procedure, best suited for a small energy interval, consists in treating the contributions of the neighboring levels as an expansion in terms of powers of the energy, by choosing appropriate combinations of background terms ^{8,9/} as discussed elsewhere.

B) Discussion of the iterative procedures

The requirement that the first order terms of the truncated Taylor expansion, given in eq. (6), are already sufficiently small, so that the expansion (6) reproduces to a sufficiently good approximation the contribution

$$X_k^{(x)} \phi_k + X_{kk}^{(x)} \phi_{kk} \quad (12)$$

evaluated at μ_k, v_k , amounts to impose the conditions that the second order terms of the Taylor expansion are either very small or cancel.

Ideally, the introduction of constraints based on this concept into the least squares scheme, would enhance the convergence of the non-linear process.

Serious consideration has been given to this aspect of the program, since a limiting feature for the changes in parameters occurring in the non-linear iterations appears necessary. One of the occurrences is, in fact, that the changes $\delta\mu$ and δv are of a magnitude such that the convergence

of the expansion (6) to the expansion of the form (12) numerically is not verified. In these cases, the linearized fitting curve may represent the data to a high degree of accuracy, whereas the curve, calculated from the line shape functions evaluated at $\mu^0 + \delta\mu$, $\nu^0 + \delta\nu$, may diverge from the experimental and from the fitted curve. In this case the non-linear iteration is considered as diverging, and the derived parameters, GR, GI are not representative of the data. In these cases, judgement can be used in interpreting the results, and subsequent fitting processes can be devised, for which the non-linear results are taken not as objectively good, but as indicative of the direction in which the non-linear parameters would tend to move, as in a gradient method⁽⁴⁾.

In the program TANDEM this point has been studied more closely, with the intent of improving the convergence of the non-linear procedure.

The literature offers a variety of minimizing schemes ensuring convergence^(5,6,10,11); in program TANDEM, simple minded tests have been inserted, which favour the convergence of the process (although no proof exists that convergence is guaranteed), and which have been chosen with the objective of keeping as much as possible of the memory storage available for the multi-level equations.

The basic criterion consists in the comparison between terms of the form (12) and terms of the form (6) around each resonance. In particular, the criterion used consists of two separate steps: at the end of each non-linear iteration the new values of μ_k and ν_k are tested by a first gross test, checking whether ν_k has become negative and whether μ_k has become either too large or too small (i.e., if μ_k lies outside the energy interval in question). If μ_k and (or) ν_k fail this test, a closer test on the accuracy of the non-linear results is meaningless. The result is understood

as indicative only of the direction of the required changes undergone by μ_k, v_k . Thus, to dampen possible oscillations which can lead to numerical instabilities, the changes of μ^0 and v^0 are reduced to $\frac{1}{10}$ of the smaller among either statistical error, $\Delta\mu$ (or Δv) and increment $\delta\mu$ (or δv). For "acceptable" values of μ_k and v_k , passing this check, a comparison is made, around μ_k , between the contribution [as in equation (12)]

$$GR_k \psi_k + GI_k \chi_k \quad (12')$$

and the corresponding contribution of the trial function (as in eq(6))

$$GR_k \psi_k^{(0)} + GI_k \chi_k^{(0)} + GR_k^{(0)} \left(\frac{\partial \psi_k}{\partial \mu_k} \delta\mu_k + \frac{\partial \psi_k}{\partial v_k} \delta v_k \right) + GI_k^{(0)} \left(\frac{\partial \chi_k}{\partial \mu_k} \delta\mu_k + \frac{\partial \chi_k}{\partial v_k} \delta v_k \right) \quad (6')$$

If the average absolute difference between the terms (12') and (6') for the interval $\mu_k - 3v_k, \mu_k + 3v_k$ differs by less than 2% from the contribution given by (12'), the second order effects which have been eliminated with the Gauss method are considered negligible, and μ_k and v_k accepted as stable values. If this test is not satisfied the values μ_k and v_k previously accepted now undergo the same damping procedure outlined above. The whole set of level parameters thus scanned is subsequently used as improved set of guesses for a linear fit meant to provide a corresponding set of linear parameters, which can be used either as estimated guesses for a subsequent non-linear fit, ($NT > 0$), or as final values.

During the non-linear process, instability can affect resonances which appeared satisfactorily fitted from the previous linear fit. To avoid this type of danger, a further test has been applied to the set of linear

parameters to be used as estimated guesses ($x_k^{(0)}$) for the subsequent non-linear process. This is a detailed test on the goodness of fit in an interval of width $6v_k$ around μ_k , where the "apparent" σ is compared to the experimental values. If the average absolute error is less than a given percentage (2%) the set of parameters pertaining to the k-th resonance are considered stable, and are eliminated from the non-linear search. This device has the advantage of stabilizing the non-linear fit and of reducing the computing time.

The general procedure adopted in TANDEM can be summarized as follows.

An initial set of level parameters, μ , v , is input, to obtain a first linear fit. If non-linear iterations are wanted, the results of the linear fit are tested by detailed evaluations of the goodness of fit around each resonance energy. Those levels which are well fitting are eliminated from the array NR and instead fitted only linearly, as a part of the array NRT. In the extreme (hopeful, but unlikely) case that all the "inside" resonances NR are well fitting, the calculations does not proceed, since in this case the previous linear fit is satisfactory. In the general cases the levels which do not pass satisfactorily the test are retained to undergo the non-linear process. At the end of it, a second type of test is entered (on the acceptability of the new values of μ , v) followed by an estimate on the magnitude of the second order variations of the trial function.

Damping of the oscillations of μ and v take place for those levels which do not satisfy either the initial test of acceptability, or the test on the second order terms. If the last test is satisfactory, the values of μ and v are not modified. In all cases the parameters GR, GI are not retained, but as all resonances have been scanned, the program enters a subsequent linear fit, providing improved linear parameters, using as

new guesses the μ, ν values scanned with the previous tests. As the wanted number of NT non-linear iterations is saturated, the process terminates with a last linear fit.

III.9. Least Squares Equations for the Two-Fold Curve Fitting of the Cross Sections.

It is known that whenever a given complete experiment is a combination of several experiments which are representable by different curves, functions of a common set of parameters and thus jointly fittable, the least squares sum may be generalized to a sum of partial least squares sums ^{12/}. It is convenient to assume here that the totality of the parameters $X_k^{(x)}$, describing fission and capture data obtained with the same experimental conditions are shared by both sets of data; these parameters can be represented by a vector $\vec{X}$ with NX components.

The known functions ϕ_k can be arranged to form a $2 \times NX$ matrix M , of the form

$$\begin{array}{lll} M_{1k} = \phi_k & M_{2k} = 0 & \text{for } k = 1, NG \\ M_{1k} = 0 & M_{2k} = \phi_k & \text{for } k = NG + 1, 2NG \\ M_{1k} = \phi_k & M_{2k} = \phi_{2NR + k} & \text{for } k = 2NG + 1, NX \end{array} \quad (13)$$

So that the expansions given in (10) and (11) can be written in the vectorial form

$$Y_{ap}^{(n)} = \sum_{k=1}^{NX} M_{nk} X_k + Y_o^{(n)} \quad (14)$$

where

$$Y_{ap}^{(n)} = \sqrt{E} \sigma_{ap}^{(n)} / C \quad (15)$$

$$Y_o^{(n)} = \sqrt{E} \sigma_{Dis}^{(n)} / C \quad (16)$$

with $n = 1, 2$.

The generalization to more than two such experiments ($n > 2$) is obvious.

The least squares sums respectively are

$$S_1 = \sum_{i=1}^{NP} [Y_{\text{ex}}^{(1)}(E_i) - Y_{\text{ap}}^{(1)}(E_i)]^2 w^{(1)}(i) \quad (17)$$

$$S_2 = \sum_{i=1}^{NP} [Y_{\text{ex}}^{(2)}(E_i) - Y_{\text{ap}}^{(2)}(E_i)]^2 w^{(2)}(i) \quad (18)$$

where NP represents the common set of data points, $w^{(n)}(i)$ represent the inverse of the statistical variances at the i -th energy point, for $n = 1, 2$, and

$$Y_{\text{ex}}^{(n)}(E_i) = \sqrt{E_i} \sigma_{\text{ex}}(E_i) / C \quad (19)$$

is the scaled, measured cross section value at E_i , for $n = 1, 2$.

The combined least squares sum is written here in general as

$$S = S_1 + \lambda S_2 \quad (20)$$

where λ is a weighting factor of arbitrary choice,

$$0 < \lambda \leq 1 \quad (21)$$

The weighing factor λ is introduced here aiming to generalize the program to the analysis of data derived from different experiments (i.e., having different experimental conditions).

Although in these cases the curve fitting should rigorously be made independently, it is possible to consider the two sets of data, which are representative of the resonance structure of the same nuclide and whose trial functions share the same parameters as ideally obtained from the same general experiment. In this case one measurement may turn out to be in the

over-all less good than the other, so that by reducing the weighting of the one by a factor $\lambda < 1$ one may bring the two independent likelihood functions,

$$e^{-S_1}, e^{-\lambda S_2}$$

to have the same order of magnitude. For the actual case of data measured with the same experiment, as it is the case for the ORNL data ^{7/}, $\lambda = 1$.

The present version of TANDEM should be used regarding $\lambda > 0$.

The conditions of minimum applied to S give

$$\frac{\partial S}{\partial X_\ell} = \frac{\partial S_1}{\partial X_\ell} + \frac{\partial S_2}{\partial X_\ell} = 0 \quad \ell = 1, NX \quad (22)$$

where

$$\frac{\partial S_1}{\partial X_\ell} = -2 \sum_{i=1}^{NP} \left(Y_{ex}^{(1)}(i) - Y_o^{(1)}(i) - \sum_{j=1}^{NX} M_{1j} X_j \right) M_{1\ell} w^{(1)}(i) \quad (23)$$

$$\frac{\partial S_2}{\partial X_\ell} = -2 \sum_{i=1}^{NP} \left(Y_{ex}^{(2)}(i) - Y_o^{(2)}(i) - \sum_{j=1}^{NX} M_{2j} X_j \right) M_{2\ell} w^{(2)}(i) \quad (24)$$

where the index i has been omitted for brevity in the matrix elements M_{nj} , and where $\ell = 1, NX$, and $\lambda = 1$.

Defining now the vector

$$R_\ell = \sum_{n=1}^2 \sum_{i=1}^{NP} \left(Y_{ex}^{(n)}(i) - Y_o^{(n)}(i) \right) w^{(n)}(i) M_{n\ell} \quad (25)$$

and the matrix

$$Q_{\ell j} = \sum_{n=1}^2 \sum_{i=1}^{NP} M_{n\ell} M_{nj} w^{(n)}(i) \quad (26)$$

the minimum condition of equation (22) becomes

$$\sum_{j=1}^{NX} Q_{lj} X_j = R_l \quad l = 1, NX \quad (27)$$

This is the well-known set of normal equations* for the least squares fitting. For linear fitting, where $NX = 2 NG$, the matrix Q is block-diagonal, of the form

$$Q_{lj} = \sum_{i=1}^{NP} \Phi_l(i) \Phi_j(i) \cdot w^{(1)}(i) \quad l = 1, NG, j = 1, NG \quad (28)$$

$$Q_{lj} = 0 \quad l = 1, NG, j = NG + 1, NX$$

$$Q_{lj} = 0 \quad l = NG + 1, NX \quad j = 1, NG$$

$$Q_{lj} = \sum_{i=1}^{NP} \Phi_l(i) \Phi_j(i) \cdot w^{(2)}(i) \quad l = NG + 1, NX \quad j = NG + 1, NX$$

In this case the least square procedure is equivalent to the consecutive, independent curve fitting of the two sets of data, because the error matrix is also block diagonal. It should be remarked, however, that if the weights are not independent, but, as is the case for the ORNL data**, the variances of the capture data are functions of the fission data. The same functional

* If $\lambda < 1$, the same equations are valid provided M_{2l} is replaced by λM_{2l} .

** G. DeSaussure: private communication.

relation will affect the definitions of Q and $\vec{R}$ and indirectly the solutions of the normal equations. In the non-linear cases the matrix Q has a more interesting structure. The block-diagonal submatrix of rank $2NG$ is bordered by elements defined as follows:

$$Q_{\ell j} = \sum_{i=1}^{NP} \Phi_{\ell}^{(i)} \Phi_j^{(i)} w^{(1)}(i) \quad \text{for } \begin{cases} \ell = 2NG + 1, NX \\ j = 1, NG \end{cases} \quad (29)$$

and for $\begin{cases} \ell = 1, NG \\ j = 2NG + 1, NX \end{cases}$

$$Q_{\ell j} = \sum_{i=1}^{NP} \Phi_{\ell+2NR}^{(i)} \Phi_j^{(i)} w^{(2)}(i) \quad \text{for } \begin{cases} \ell = 2NG + 1, NX \\ j = NG + 1, 2NG \end{cases} \quad (30)$$

and

$$Q_{\ell j} = \sum_{i=1}^{NP} \Phi_{\ell}^{(i)} \Phi_{j+2NR}^{(i)} w^{(2)}(i) \quad \text{for } \begin{cases} \ell = NG + 1, 2NG \\ j = 2NG + 1, NX \end{cases}$$

$$Q_{\ell j} = \sum_{i=1}^{NP} \left[\Phi_{\ell}^{(i)} \Phi_j^{(i)} w^{(1)}(i) + \Phi_{\ell+2NR}^{(i)} \Phi_{j+2NR}^{(i)} w^{(2)}(i) \right] \quad (31)$$

for $\ell, j = 2NG + 1, NX$

While the submatrix defined in (31) contains explicitly the line shape functions related to the non-linear fitting, and, therefore, combines contributions from both trial functions, the rectangular matrices defined in (29) and (30), instead connect the unknown parameters X relative to one curve to those components of the vector $\vec{R}$ which refer to the other curve. Therefore, in the non-linear case the entire set of the parameters X is affected by the existence of this type of coupling. The unknown parameters are obtained by the solution of the normal equations

$$Q\vec{X} = \vec{R} \quad (32)$$

For each fitting, the minimum value of S , S^* , is calculated by introducing into equation (20) the trial functions calculated with the found solutions $\vec{X}$; hence the quantity

$$\chi = \sqrt{\frac{S^*}{ND}} \quad (33)$$

is obtained, as a measure of the goodness of fit, where $ND = 2NP - NX$.

Finally, for each solution X_k the corresponding standard deviation is obtained as

$$\delta X_k = \chi \sqrt{Q_{kk}^{-1}} \quad (34)$$

where Q^{-1} is the error matrix. The error matrix Q^{-1} is furthermore utilized to calculate for each value of E , the standard deviation $\delta y_{ap}^{(n)}$ of the fitted curve $y_{ap}^{(n)}$, given by:

$$\delta y_{ap}^{(n)} = \chi \left(\sum_{j=1}^{NX} \sum_{l=1}^{NX} M_{nj} M_{nl} Q_{jl}^{-1} \right)^{1/2} \quad (35)$$

IV Description of the program TANDEM and of the associated FORTRAN subroutines:

The detailed description of the input and output of TANDEM is given in Appendix A. The program listing forms Appendix B.

IV-1) Main Program. Control indices and data are read from cards, the data points are stored in the fast memory, after appropriate scaling, as given in equation (19).

The following FORTRAN subroutines are called from the Main program: SKIP, SHAPE, WETA, SIGTWO, TEST, and SIGPLT.

Another subroutine, pertaining to the FORTRAN II system of the University of Illinois is FTRAP, which controls underflows, setting equal to zero numbers less than 10^{-36} .

The description of SKIP and SHAPE can be found in Reference 3.

IV-2) Subroutine WETA (IWE, KP). This subroutine calculates the weights of both data sets at each energy point. WETA is controlled by three indices: IWE, KP, JWE, this last being in common.

IWE can assume only the values 1 and 0. Initially IWE = 1, and the calculation branches to obtain two scaling factors for the two sets of weights, so that

$$\sum_{i=1}^{NP} w^{(n)}(i) = NP \quad n = 1, 2 \quad (36)$$

where NP is the number of the data points.

For the rest of the calculation IWE = 0, and in this branch the weights are calculated and scaled at each energy point, of index KP. The index JWE at present assumes two values, 1, and 2. For JWE = 1 equal weights are calculated. This option is useful in cases in which curve fitting of calculated functions is wanted. For JWE = 2 the routine calculates the energy dependent variances for the data of ORNL, as indicated by the authors*:

$$\text{var}(\sigma_F \sqrt{E}) = 2.6 \times 2^I (\sigma_F \sqrt{E} + .5) \quad (37)$$

* G. DeSaussure: private communication

$$\text{var}(\sigma_c \sqrt{E}) = 1.5 \times 2^I (\sigma_c \sqrt{E} + .41 \sigma_F \sqrt{E} + .35) \quad (37')$$

where I is the channel index provided by input. From (37) and (37'), the weights are derived as

$$w^{(1)}(KP) = \alpha_1 / \text{var}(\sigma_F \sqrt{E}) \quad (38)$$

$$w^{(2)}(KP) = \alpha_2 / \text{var}(\sigma_c \sqrt{E}) \quad (39)$$

where $E = E(KP)$ and where α_1, α_2 are the factors obtained in the initializing part of the calculation.

Other types of weights can be added readily including consecutive choices for the index JWE.

IV-3 Subroutine SIGTWO.

This subroutine controls the least squares procedure, first by forming the appropriate arrays, Φ_k , R_ℓ , and the matrix Q, as defined in eqs. (3), (7), (8), (25), and (26) to establish the normal equations. In this part of the calculation the subroutines used for the evaluation of these arrays are SIGOUT, and PHIDO.

As a second step, the inversion of the matrix Q, the vector solution $\vec{X}$, and the associated errors $\delta \vec{X}$, are obtained by calling the function INVERT.

In the third part of SIGTWO are calculated the quantities associated with the fitting, and defined in equations (33), (34), and (35). The last part of SIGTWO prepares output of parameters and errors and card punching of the same parameters, for use in CURVEPLOT^{3/} and LIST^{13/}

IV-4 Subroutine TEST.

This subroutine is controlled by the index ITEST, which can assume the values 1 and 2. TEST is entered from the Main program only when non-linear iterations are requested (input index NT $\neq$ 0).

For ITEST = 1, after a given linear fit preceeding a non-linear iteration, the goodness of fit is tested around each resonance peak. If the deviations are smaller than a given percentage, the parameters fitting the level in question are considered as "good", and are stored to be treated only linearly in the successive non-linear iteration. This is accomplished by storing temporarily these levels in the NOUT array, so as to continue the testing without loss of terms. At the end of the testing process, this group of levels is treated separately in entering the non-linear fitting. This process amounts to reduce the number of the levels to be fitted non-linearly, that is the dimensions of the vectors Φ_j of equations (7) and (8), keeping instead unchanged NRT, the total number of resonances fitted. If the test is not satisfied, the levels in question are instead retained for the subsequent non-linear iteration. In the extreme case in which the entire set of parameters is "good" the calculation does not proceed, since it would repeat exactly the previous linear fit.

The branch ITEST = 2 is used after a non-linear fit, before entering a subsequent linear fit. This test is to establish the acceptability of the values μ_k and ν_k as obtained from the non-linear procedure. For each resonance the values μ_k and ν_k undergo a preliminary test, to ascertain whether μ_k has moved out of range or ν_k has changed so uncontrollably as to become negative. Parameters not passing this test undergo changes meant

to dampen their oscillation, that is new values are obtained by producing changes for the original set μ_k^0, v_k^0 , which are a fraction (1/10) of the smaller of either previous increment or statistical error. These "adjusted" values are subsequently used as guesses for the following linear fit.

If the parameters pass the preliminary test, the calculation proceeds to evaluate the convergence of the linearized trial function [equation(6')] to the corresponding contribution of the form (12'). The value of this difference is taken as an indication of the magnitude of the second order terms in the truncated Taylor expansion. If this difference, around each level, is less than a given percentage, the parameters μ_k, v_k are kept as "stable" and used without modifications as input for the following linear fit. If the test is not satisfactory such values of μ_k and v_k undergo the same damping operation, described above, before entering the linear fit.

In this manner, the parameters GR, GI, derived from the non-linear process are entirely disregarded, and the non-linear process is used only to obtain "improved" sets of guesses for the linear fits.

IV-5 Subroutine SIGPLT

After completion of the wanted number of non-linear fits (which can as well be zero) the Main program calls SIGPLT to tabulate as functions of E the following quantities:

$$\sqrt{E} \sigma_{ex}^{(n)}, \sqrt{E} \sigma_{ap}^{(n)}, \sqrt{E} \sigma_c^{(n)}, \sqrt{E} [\sigma_{ex}^{(n)} - \sigma_{ap}^{(n)}], \% \text{ deviation, standard deviation.}$$

The evaluation of $\sqrt{E} \sigma_c^{(n)}$ is obtained from the subroutine SIGIN, while $\sqrt{E} \sigma_{ap}^{(n)}$ is calculated directly from the arrays $\phi, \vec{X}$ stored on tape, and transmitted by COMMON. SIGPLT also provides the values of $\sigma_{ap}^{(n)}$ at $E = .025$, to test in each case the fitted values against the tabulated thermal values.

IV-6 Subroutine PHIDO

The line shape functions ϕ_k are calculated in the subroutine PHIDO according to the definitions given in equations (3), (7), and (8). The arrays ϕ , forming the matrix M of equation (13) are thus stored on tape and transmitted to SIGTWO by COMMON.

The actual evaluation of ϕ is controlled by the index JS. This index can have the values 1 and 2.

For JS = 1 the line shape functions are obtained from SHAPE with Doppler broadening only.

For JS = 2 the resolution width, HR, is evaluated via subroutine WIDTH, and the line shape functions are obtained from SHAPE with a value of the "temperature" obtained by combining the broadening and the resolution widths $\frac{3}{2}$.

IV-7 Function INVERT

This "function" prepares the storage locations needed for the double precision inversion of Q. The inversion is obtained by calling the double precision inversion subroutine DNV1 (written in SCATRE). The values of the error matrix Q^{-1} are rounded, from double to single precision, with the function ROUND, and returned to SIGTWO in the original storage locations of Q. Similarly $\vec{X}$ and the errors $\delta\vec{X}$ are calculated in double precision arithmetic, rounded, and returned to SIGTWO in the arrays $\vec{R}$ and $\overline{VARX}$, respectively.

IV-8 Subroutine SIGOUT

In SIGOUT are evaluated the contributions of the known resonances, NOUT, as defined in equations (5) and (16). As in the subroutine PHIDO,

the broadening effects are obtained with the subroutines WIDTH, and SHAPE; depending on the value of the index JS. Among the NOUT can be included parameters previously obtained as NEIGH. In this case the parameters GR, GI pertaining to a resonance fitted as NEIGH need scaling (see eqs. (3)) when used as part of the NOUT set. For negative energy levels the temperature effect is not introduced, hence the calculation in this case skips the evaluation of the broadening. The NOUT set of parameters is transmitted by COMMON statements.

IV-9 Subroutine SIGIN

This subroutine evaluates purely Doppler broadened line shape functions and provides the corresponding values of $\sqrt{E} \sigma_c^{(n)}(E)$; for the contribution $\sigma_{Dis}^{(n)}$ to σ_c the subroutine SIGOUT is called, previous setting JS to the value 1.

IV-10 Subroutine SIGAP

This subroutine evaluates Doppler and resolution broadened values of the line shape functions and provides the corresponding values for $\sqrt{E} \sigma_{app}^{(n)}(E)$. The NOUT contributions are obtained from SIGOUT, as in SIGIN.

IV-11 Subroutine WIDTH (EN,HR)

For each value EN of the energy variable E the resolution width HR is calculated, being defined as the half-width at $\frac{1}{e}$ of the maximum of a Gaussian resolution function. This version of WIDTH contains the definition of HR, for the Linac time-of-flight experiment of the ORNL-RPI group*^{7/}.

$$HR^2 = A E^3 + B E^2 \quad (40)$$

with

$$\begin{aligned} A &= 3.15 \times 10^{-10} \times (39.10 + 4^I) \\ B &= 1.71 \times 10^{-6} \end{aligned} \quad (41)$$

* G. DeSaussure: private communication

where I is the channel index, and where HR is in eV. This subroutine can be enlarged to contain other such resolution functions or replaced if needed.

APPENDIX A

Input and Output of TANDEM

A-1 Constants Used in the Input

TEMP = Kinetic energy of the target nucleus in eV
ATM = Atomic mass number of the nucleus
CC = Conversion constant $C = 6.52 \times 10^5$ barns x eV
FACS = Weighting factor $\lambda (0 < \lambda \leq 1)$
EMAX = Largest energy point included in the data set

A-2 Indices Used in the Input Controlling
The Code Operations

NR = Number of resonance levels to be fitted inside the energy interval
pertaining to the data set $(0 < NR = NRT - NEIGH; 0 < NRT \leq 15)^*$
NEIGH = Number of resonance levels to be fitted, neighboring the energy
interval of interest $(0 \leq NEIGH = NRT - NR; 0 < NRT \leq 15)$
NOUT = Number of resonance levels, either inside or outside the energy
interval of interest, excluded from the fitting, and having known
parameters. $(0 \leq NOUT \leq 20)$.

*The index NRT is not input.

KNEG = Number of background terms of the form

$$A_i E^{1-i}, \quad i = 1, \text{KNEG.} \quad (\text{KNEG} \leq 3)$$

KPOS = Number of background terms of the form

$$B_i E^i \quad i = 1, \text{KPOS} \quad (\text{KPOS} \leq 5)$$

NT = Number of non-linear iterations requested after the first linear fit. $(\text{NT} \geq 0)$

JS = Index controlling the choice of the broadening for the line shape functions:

for JS = 1, Doppler broadening only

for JS = 2, Doppler and resolution broadening

JWE = Index controlling the choice of the statistical weights in WETA:

JWE = 1, equal weights

JWE = 2, weights calculated as energy dependent functions according to the experimental conditions.

Note: Further choices can be added (see e.g. ref. 3)

IPR = Index used to obtain test outputs for the matrix inversion, when set > 0 . For ordinary operations, IPR = 0.

A-3 Data Points

The data are given on cards, in the sequence, ICH(I), EV(I), YFS(I), YCA(I), four per card, where for the I-th energy point ICH is the channel index, EV the corresponding value of the energy E in eV, YFS the $\sqrt{E} \sigma_F(E)$ value at E, YCA the $\sqrt{E} \sigma_C(E)$ value at E, in barns * eV^{1/2}.

The data cards are counted while they are read; for this reason EMAX is set to be equal to the last E-value of the last data card.

This input format is the original format of the ORNL data, and can be adjusted for other data to their appropriate format.

A-4 Resonance Parameters

"Guesses" for NR + NEIGH resonances, containing the values of μ (resonance energy) and ν (half width) follow the data.

Known parameters for the NOUT sets must also be supplied, if NOUT $\neq$ 0, a set for fission followed by a set for capture parameters.

A-5 Preparation of the Input:

Sequence of Input Cards

Card 1	Comment card with a run number in Col 73-80	}	Format (3A6,A2)
Card 2	Comment card		
Card 3	Constants TEMP, ATM, CC, FACS, EMAX		Format (5E15.8)
Card 4	Indices NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, JWE, IPR		Format (9I5)

These cards are followed by the DATA CARDS, Format [4 (I1, F7.3, 2F6.1)]

The DATA are followed by:

NRT PARAMETER CARDS (Guesses) (where NRT = NR + NEIGH) for μ_k , v_k . Format (2E15.8).

Finally, the two sets of NOUT (when NOUT $\neq$ 0) follows, having

μ , v , GR, GI Format (4E15.8)

in each card. Usually these last cards are available as previous output from this program or from CODILLI 3/.

A-6 Output of TANDEM: Printed Output

The first page of the output reproduces and describes the input. In the second page, quantities relevant to the least squares procedure, as defined in equation (33) are output:

$$RS = S^*$$

$$GF = \chi$$

$$CHISQ = \chi^2$$

The resonance parameters are then tabulated in the following sequences:

NR resonances

k μ $\delta\mu$ GR^F δGR^F GR^C δGR^C (k = 1, NR)

k v δv GI^F δGI^F GI^C δGI^C (k = 1, NR)

These are followed by two similar sets for the neighboring levels when NEIGH $\neq$ 0.

The constants A_i , B_i , are then output.

At the end of the last iteration the subroutine SIGPLT tabulates the obtained value of the cross sections as follows:

KP E(KP), $\sqrt{E} \sigma_{ex}^{(1)}(KP)$, $\sqrt{E} \sigma_{ap}^{(1)}(KP)$, $\sqrt{E} \sigma_c^{(1)}(KP)$, Dev., % Dev., St. Dev.,

where KP = 1, NP is the data point index, σ_{ex} , σ_{ap} , σ_c are defined in sections 2 and 3, and Dev. = $(\sigma_{ex} - \sigma_{ap}) \sqrt{E}$

% Dev. = $(\sigma_{ex} - \sigma_{ap}) / \sigma_{ex}$

St. Dev. = Standard deviation (see eq. (35) in Section 3)

A similar tabulation for $\sigma^{(2)}$ follows.

A-7 Output of TANDEM: Punched Output

Parameter cards are punched, after each fit, linear or non-linear, to be used as input in the programs CURVEPLOT and SIGMA. These cards have the format, μ , ν , GR, GI, NC, K, LR, A(14), A(13).

Format (4E15.7, I3, I5, I3, A6, A2)

where NC = number of iteration ($0 \leq NC \leq NT$)

K = index of the resonance level

LR = index characterizing either cross section (LR = 1, 2)

A(14), A(13) are the alphanumeric symbols of the first comment card,

meant to identify the Run.

After each linear fit an additional set of cards is punched, containing the errors, δGR , δGI , as defined in eq. (34), and errors for $\delta\mu$ and $\delta\nu$ which are based on the value of the resolution width at μ , combined with the statistical variance derived from the non-linear fit. These cards are useful for the purpose of obtaining automated tabulations of the results, using the routine LIST (13).

APPENDIX B

PROGRAM TANDEM

JAN 1967

TNDM	1
TNDM	2
TNDM	3
TNDM	4
TNDM	5
TNDM	6
TNDM	7
TNDM	8
TNDM	9
TNDM	10
TNDM	11
TNDM	12
TNDM	13
TNDM	14
TNDM	15
TNDM	16
TNDM	17
TNDM	18
TNDM	19
TNDM	20
TNDM	21
TNDM	22
TNDM	23
TNDM	24
TNDM	25
TNDM	26
TNDM	27
TNDM	28
TNDM	29
TNDM	30
TNDM	31
TNDM	32
TNDM	33
TNDM	34
TNDM	35
TNDM	36
TNDM	37
TNDM	38
TNDM	39
TNDM	40
TNDM	41
TNDM	42
TNDM	43
TNDM	44
TNDM	45
TNDM	46
TNDM	47
TNDM	48
TNDM	49
TNDM	50
TNDM	51
TNDM	52
TNDM	53
TNDM	54
TNDM	55
TNDM	56
TNDM	57
TNDM	58

TWOFOLD LEAST SQUARES ANALYSIS OF CAPTURE AND FISSION CROSS
SECTION DATA WITH THE MULTILEVEL EXPANSION -- THIS VERSION
APPLIES TO THE ORNL-RPI DATA

FOR THE INPUT OPERATION SEE WRITE-UP..... NOTE HERE NP
IS NOT INPUT, BUT EMAX IS PICKED EQUAL TO THE LAST E-VALUE OF THE
DATA CARDS-- DATA CARDS FORMAT IS THE ORIGINAL ORNL FORMAT.

DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),
2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(TNDM
3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)TNDM
4, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)TNDM
COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, TNDM
2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCA, EV,TNDM
3 A TNDM

COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,TNDM
2 NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS TNDM

CALL FTRAP
CALL SHAPE (1,X,X,X,X,2)
100 CALL SKIP(1)
CALL SKIP(2)

READ CONSTANTS
READ 500, TEMP, ATM, CC, FACS, EMAX
WOT 6, 501, TEMP, ATM, CC, FACS

READ INDICES
READ 502,NR,NEIGH, NOUT, KNEG, KPOS, NT,JS,JWE,IPR

KP = 0

READ DATA
READ 503, (ICHA(I), EV(I), YFS(I), YCA(I), I = 1,4)
DO 2 I = 1,4
KP = KP + 1
E(KP) = EV(I)
Y(KP) = YFS(I) / CC
NKP = 1000 + KP
Y(NKP) = YCA(I) / CC

IF (ABSF(EV(4) - EMAX) - .0001) 3,3,1

NP = KP
DO 4 KP = 1,NP
NKP = NP + KP
NNKP = 1000 + KP
Y(NKP) = Y(NNKP)
CALL WETA(1,KP)
WOT 6, 504, NP, NT, JS, IPR
WOT 6, 505, NR, NEIGH, NOUT, KNEG, KPOS, JWE

READ NR GUESSES

C		TNDM	59
	WOT 6, 506	TNDM	60
	DO 5 K = 1,NR	TNDM	61
	READ 507, FU(K), GU(K)	TNDM	62
C		TNDM	63
	WOT 6, 508	TNDM	64
	DO 6 K = 1,NR	TNDM	65
5	WOT 6, 509, K, FU(K), GU(K)	TNDM	66
	IF (NEIGH) 7,10,7	TNDM	67
		TNDM	68
	READ NEIGH GUESSES	TNDM	69
C		TNDM	70
7	WOT 6, 510	TNDM	71
	DO 8 K = 1,NEIGH	TNDM	72
	KN = NR + K	TNDM	73
8	READ 507, FU(KN), GU(KN)	TNDM	74
		TNDM	75
	WOT 6, 508	TNDM	76
	DO 9 K = 1,NEIGH	TNDM	77
	KN = NR + K	TNDM	78
9	WOT 6, 509, K, FU(KN), GU(KN)	TNDM	79
10	IF (NOUT) 11,16,11	TNDM	80
		TNDM	81
	READ NOUT PARAMETERS	TNDM	82
11	WOT 6, 511	TNDM	83
	WOT 6, 513	TNDM	84
	DO 12 K = 1,NOUT	TNDM	85
	READ 512, DL(K), DW(K), DGR(1,K), DGI(1,K)	TNDM	86
12	WOT 6, 512, DL(K), DW(K), DGR(1,K), DGI(1,K)	TNDM	87
	WOT 6, 514	TNDM	88
	DO 13 K = 1,NOUT	TNDM	89
	READ 512, DL(K), DW(K), DGR(2,K), DGI(2,K)	TNDM	90
13	WOT 6, 512, DL(K), DW(K), DGR(2,K), DGI(2,K)	TNDM	91
		TNDM	92
	PREPARE LINEAR FIT	TNDM	93
C		TNDM	94
16	NLIN=0	TNDM	95
116	NC=0	TNDM	96
	CALL SKIP(2)	TNDM	97
	WOT 6, 515	TNDM	98
	NG = 2 * (NR+NEIGH) + KNEG + KPOS	TNDM	99
	NX = 2 * NG	TNDM	100
	NTOT = NX	TNDM	101
	NRT=NR+NEIGH	TNDM	102
	NINS = 0	TNDM	103
	CALL SIGTWO	TNDM	104
	IF (NT) 17,17,14	TNDM	105
C		TNDM	106
C	PREPARE NON-LINEAR FIT	TNDM	107
14	NC = NC + 1	TNDM	108
	IF (NT-NC) 17,15,15	TNDM	109
15	CALL SKIP(2)	TNDM	110
	ITN = NC + NLIN	TNDM	111
	WOT6, 516, ITN	TNDM	112
	CALL TEST (1)	TNDM	113
	CALL SKIP(2)	TNDM	114
	WOT6,516,ITN	TNDM	115
	IF(NR) 100,100,117	TNDM	116
117	NRT = NR + NEIGH + NINS	TNDM	117
	NG = 2 * NRT + KPOS + KNEG	TNDM	118

	NX = 2 * NG + 2 * NR	TNDM 119
	NTOT = NX + 2 * NR	TNDM 120
	CALL SIGTWO	TNDM 121
	CALL TEST(2)	TNDM 122
	GO TO 116	TNDM 123
		TNDM 124
	TABULATE SIGMAS	TNDM 125
17	CALL SIGPLT	TNDM 126
	GO TO 100	TNDM 127
		TNDM 128
500	FORMAT (5E15.8)	TNDM 129
501	FORMAT (6HOTEMP=F8.4, 5H ATM=F5.0, 6H CC=E15.8, 6H FACS=E15.8)	TNDM 130
02	FORMAT (9I5)	TNDM 131
03	FORMAT (4(I1, F7.3, 2F6.1))	TNDM 132
504	FORMAT (65HOTWOFOLD FITTING OF FISSION AND CAPTURE DATA....THIS RUN	TNDM 133
	2N CONTAINS I4, 26H DATA POINTS, WILL PERFORM I2, 23H NON-LINEAR IT	TNDM 134
	3ERATIONS,/24H WITH TRIAL FUNCTION JS=I1, 10H, AND IPR=I1)	TNDM 135
505	FORMAT (10HOTHER ARE I3, 19H INSIDE RESONANCES, I3, 24H NEIGHBORIT	TNDM 136
	2NG RESONANCES, I3, 20H OUTSIDE RESONANCES, I2, 13H CONSTANTS A, I2	TNDM 137
	3, 18H CONSTANTS B, JWE=I1)	TNDM 138
506	FORMAT (30HOGUESSES FOR INSIDE RESONANCES)	TNDM 139
507	FORMAT (2E15.8)	TNDM 140
08	FORMAT (1H0, 4X, 1HK, 12X, 2HMU, 18X, 2HNU/)	TNDM 141
09	FORMAT (I5, 2E20.8)	TNDM 142
510	FORMAT (32HOGUESSES FOR NEIGHBOR RESONANCES)	TNDM 143
11	FORMAT (34HOPARAMETERS FOR OUTSIDE RESONANCES)	TNDM 144
12	FORMAT (4E15.8)	TNDM 145
513	FORMAT (34HOPARAMETERS FOR FISSION RESONANCES//10X,2HMU, 13X, 2HNUT	TNDM 146
	2, 12X, 3HDGR, 12X, 3HDGI/)	TNDM 147
14	FORMAT (34HOPARAMETERS FOR CAPTURE RESONANCES//10X,2HMU, 13X, 2HNUT	TNDM 148
	2, 12X, 3HDGR, 12X, 3HDGI/)	TNDM 149
515	FORMAT (11HOLINEAR FIT)	TNDM 150
16	FORMAT (22HONON-LINEAR FIT NUMBER I3)	TNDM 151
	END	TNDM 152

SUBROUTINE SIGTWO

SOLUTION OF NORMAL EQUATIONS AND PARAMETERS OUTPUT---

	SIG2	1
	SIG2	2
	SIG2	3
	SIG2	4
DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),	SIG2	5
2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(SIG2	SIG2	6
3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)SIG2	SIG2	7
4, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)SIG2	SIG2	8
COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, SIG2	SIG2	9
2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV,SIG2	SIG2	10
3 A	SIG2	11
COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,SIG2	SIG2	12
2 NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS	SIG2	13
NXX = NX + 1	SIG2	14
DO 1 K = 1,NX	SIG2	15
R(K) = 0.0	SIG2	16
PAM(K) = 0.0	SIG2	17
DO 1 J = 1,NX	SIG2	18
1 Q(K,J) = 0.0	SIG2	19
C PREPARE ARRAYS FOR FIT	SIG2	20
REWIND 9	SIG2	21
VARX(NXX) = 0.0	SIG2	22
NRT = NR + NINS + NEIGH	SIG2	23
KP = 1	SIG2	24
9 EN = E(KP)	SIG2	25
CALL SIGOUT (EN)	SIG2	26
CALL PHIDO (EN)	SIG2	27
WRITE TAPE 9, DSIG(1), DSIG(2), (PHI(I), I = 1,NTOT)	SIG2	28
CALL WETA(0,KP)	SIG2	29
COMMON TO LINEAR AND TO NON-LINEAR CASES	SIG2	30
NKP = NP + KP	SIG2	31
YE = Y(KP) - DSIG(1)	SIG2	32
YC = Y(NKP) - DSIG(2)	SIG2	33
DO 2 J = 1,NG	SIG2	34
R(J) = R(J) + W(1) * PHI(J) * YE	SIG2	35
JNG = NG + J	SIG2	36
R(JNG) = R(JNG) + W(2) * PHI(JNG) * YC	SIG2	37
DO 2 JJ = 1,NG	SIG2	38
Q(J,JJ) = Q(J,JJ) + W(1) * PHI(J) * PHI(JJ)	SIG2	39
JJNG = NG + JJ	SIG2	40
2 Q(JNG,JJNG) = Q(JNG,JJNG) + W(2) * PHI(JNG) * PHI(JJNG)	SIG2	41
IF (NC) 8,8,3	SIG2	42
C NON-LINEAR CASES	SIG2	43
3 KMIN = 2 * NG + 1	SIG2	44
DO 4 K = KMIN,NX	SIG2	45
KK = 2 * NR + K	SIG2	46
4 R(K) = R(K) + W(1) * YE * PHI(K) + FACS * W(2) * YC * PHI(KK)	SIG2	47
JMIN = 2*NG + 1	SIG2	48
DO 5 K = 1,NG	SIG2	49
DO 5 J = JMIN,NX	SIG2	50
Q(K,J) = Q(K,J) + W(1) * PHI(K) * PHI(J)	SIG2	51
5 Q(J,K) = Q(K,J)	SIG2	52
KMIN = NG + 1	SIG2	53
KMAX = 2*NG	SIG2	54
JMIN = 2*NG + 1	SIG2	55
DO 6 K = KMIN,KMAX	SIG2	56
DO 6 J = JMIN,NX	SIG2	57
JJ = J + 2*NR	SIG2	58
Q(K,J) = Q(K,J) + W(2) * PHI(K) * PHI(JJ)	SIG2	59

6	Q(J,K) = Q(K,J)	SIG2	60
	KMIN = 2 * NG + 1	SIG2	61
	DO 7 K = KMIN,NX	SIG2	62
	DO 7 J = KMIN,NX	SIG2	63
	JJ = J + 2*NR	SIG2	64
	KK = K + 2 * NR	SIG2	65
7	Q(K,J) = Q(K,J) + W(1)*PHI(K)*PHI(J) + FACS*W(2)*PHI(KK)*PHI(JJ)	SIG2	66
8	KP = KP + 1	SIG2	67
	IF (KP - NP) 9,9,10	SIG2	68
	C SCALE Q	SIG2	69
	10 DO 100 K=1,NX	SIG2	70
	DO 100 J=1,NX	SIG2	71
	100 Q(K,J)=Q(K,J)*(1.0/100.0)	SIG2	72
	C TEST OUTPUT	SIG2	73
	IF(IPR) 13,13,11	SIG2	74
11	WOT 6, 900	SIG2	75
	DO 12 I = 1,NX	SIG2	76
	WOT 6, 901, I, (Q(I,J), J = 1,NX)	SIG2	77
12	WOT 6, 902, R(I)	SIG2	78
13	REWIND 9	SIG2	79
	I = INVERT (NX)	SIG2	80
	WOT 6, 903, DET(2), I	SIG2	81
	C RESCALE SOLUTIONS AND Q	SIG2	82
	DO 101 K=1,NX	SIG2	83
	R(K)=R(K)/100.0	SIG2	84
	VARX(K)=VARX(K)/10.0	SIG2	85
	DO 101 J=1,NX	SIG2	86
	101 Q(K,J)=Q(K,J)/100.0	SIG2	87
	IF (IPR) 16,16,14	SIG2	88
14	WOT 6, 904	SIG2	89
	DO 15 I = 1,NX	SIG2	90
15	WOT 6, 901, I, (Q(I,J), J = 1,NX)	SIG2	91
	CALL SKIP(2)	SIG2	92
	C EVALUATE FITNESS	SIG2	93
16	RS=0.0	SIG2	94
	DO 20 KP = 1,NP	SIG2	95
	READ TAPE 9, DSIG(1), DSIG(2), (PHI(I), I = 1,NTOT)	SIG2	96
	PART = 0.0	SIG2	97
	PORT = 0.0	SIG2	98
	NKP = NP + KP	SIG2	99
	CALL WETA(0,KP)	SIG2	100
	DO 17 I = 1,NG	SIG2	101
	PART = PART + R(I) * PHI(I)	SIG2	102
	IC = NG + I	SIG2	103
17	PORT = PORT + R(IC) * PHI(IC)	SIG2	104
	IF (NC) 20,20,18	SIG2	105
18	IMIN = 2 * NG + 1	SIG2	106
	DO 19 I = IMIN,NX	SIG2	107
	PART = PART + R(I) * PHI(I)	SIG2	108
	IC = 2*NR + I	SIG2	109
19	PORT = PORT + R(I) * PHI(IC)	SIG2	110
20	RS = RS + ((Y(KP) - DSIG(1) - PART)**2) * W(1) + FACS * ((Y(NKP) -	SIG2	111
	2 DSIG(2) - PORT)**2)* W(2)	SIG2	112
	REWIND 9	SIG2	113
	ND=2*NP-NX-1	SIG2	114
	D = FLOATF(ND)	SIG2	115
	XYZ = RS/D	SIG2	116
	GF = SQRT(XYZ)	SIG2	117
	WOT 6, 905, RS, GF, XYZ, NR	SIG2	118
	DO 21 K = 1,NX	SIG2	119

21	VARX(K) * VARX(K) * GF	SIG2 120
C DEFINITION OF PARAMETERS		SIG2 121
IF (NC) 22,22,40		SIG2 122
LINEAR CASE		SIG2 123
22	DO 23 K = 1,NRT	SIG2 124
	KK = NRT + K	SIG2 125
	GR(1,K) = R(K)	SIG2 126
	GI(1,K) = R(KK)	SIG2 127
	KG = NG + K	SIG2 128
	KKG = NG + KK	SIG2 129
	GR(2,K) = R(KG)	SIG2 130
23	GI(2,K) = R(KKG)	SIG2 131
	KMIN = 2 * NRT + 1	SIG2 132
	JMIN = NG + KMIN	SIG2 133
	KMAX = 2 * NRT + KNEG	SIG2 134
	JMAX = NG + KMAX	SIG2 135
	IF (KNEG) 30,30,24	SIG2 136
24	DO 25 K = KMIN,KMAX	SIG2 137
	KC = K - 2 * NRT	SIG2 138
25	AC(1,KC) = R(K)	SIG2 139
	DO 26 J = JMIN,JMAX	SIG2 140
	JC = J - 2*NRT - NG	SIG2 141
26	AC(2,JC) = R(J)	SIG2 142
30	IF (KPOS) 33,33,31	SIG2 143
31	DO 32 K = 1,KPOS	SIG2 144
	KC = KMAX + K	SIG2 145
	JC = JMAX + K	SIG2 146
	BC(1,K) = R(KC)	SIG2 147
32	BC(2,K) = R(JC)	SIG2 148
	GO TO 33	SIG2 149
C NON LINEAR CASE		SIG2 150
40	DO 41 K = 1,NRT	SIG2 151
	KK = NRT + K	SIG2 152
	KJ = NG + K	SIG2 153
	KKJ = NG + KK	SIG2 154
	PAM(K) = GR(1,K)	SIG2 155
	PAM(KK) = GI(1,K)	SIG2 156
	GR(1,K) = R(K)	SIG2 157
	GI(1,K) = R(KK)	SIG2 158
	PAM(KJ) = GR(2,K)	SIG2 159
	PAM(KKJ) = GI(2,K)	SIG2 160
	GR(2,K) = R(KJ)	SIG2 161
41	GI(2,K) = R(KKJ)	SIG2 162
27	IF (KNEG) 44,44,42	SIG2 163
42	DO 43 K = 1,KNEG	SIG2 164
	KK = 2 * NRT + K	SIG2 165
	KJ = NG + KK	SIG2 166
	AC(1,K) = R(KK)	SIG2 167
43	AC(2,K) = R(KJ)	SIG2 168
44	IF (KPOS) 47,47,45	SIG2 169
45	DO 46 K = 1,KPOS	SIG2 170
	KK = 2 * NRT + KNEG + K	SIG2 171
	KJ = NG + KK	SIG2 172
	BC(1,K) = R(KK)	SIG2 173
	BC(2,K) = R(KJ)	SIG2 174
47	DO 48 K = 1,NR	SIG2 175
	KK = 2 * NG + K	SIG2 176
	KJ = NR + KK	SIG2 177
	PAM(KK) = FU(K)	SIG2 178
	PAM(KJ) = GU(K)	SIG2 179

	FU(K) = FU(K) + R(KK)	SIG2 180
	GU(K) = GU(K) + R(KJ)	SIG2 181
	WRITE OUTPUT	SIG2 182
33	WOT 6, 906	SIG2 183
	WOT6,907	SIG2 184
	DO36 K=1,NR	SIG2 185
	KK=NG+K	SIG2 186
	IF(NC) 34,34,35	SIG2 187
34	II=NX+1	SIG2 188
	GO TO 36	SIG2 189
35	II=2*NG +K	SIG2 190
36	WOT6,908,K,FU(K),VARX(II),GR(1,K),VARX(K),GR(2,K),VARX(KK)	SIG2 191
	IF (NINS) 120,120,121	SIG2 192
121	II = NX + 1	SIG2 193
	DO 122 KK = 1,NINS	SIG2 194
	K = NR + KK	SIG2 195
	KS = NG +K	SIG2 196
122	WOT6,908, K,FU(K),VARX(II),GR(1,K),VARX(K),GR(2,K),VARX(KS)	SIG2 197
120	WOT 6, 909	SIG2 198
	DO 81 K =1,NR	SIG2 199
	KK = NRT + K	SIG2 200
	KKG = NG + KK	SIG2 201
	VARI = VARX(KK)	SIG2 202
	VARO = VARX(KKG)	SIG2 203
	IF (NC) 38,38,39	SIG2 204
38	II = NX + 1	SIG2 205
	GO TO 81	SIG2 206
39	II= 2*NG+ NR + K	SIG2 207
81	WOT6, 908,K,GU(K),VARX(II),GI(1,K),VARI,GI(2,K),VARO	SIG2 208
	IF (NINS) 123,123,124	SIG2 209
124	II = NX + 1	SIG2 210
	DO 125 KN = 1,NINS	SIG2 211
	K = NR + KN	SIG2 212
	KK = NRT + K	SIG2 213
	KKG = NG + KK	SIG2 214
	VARI = VARX(KK)	SIG2 215
	VARO = VARX(KKG)	SIG2 216
125	WOT6, 908,K,GU(K),VARX(II),GI(1,K),VARI,GI(2,K),VARO	SIG2 217
123	IF (NEIGH) 87,87,37	SIG2 218
37	WOT6, 910	SIG2 219
	WOT6,907	SIG2 220
	DO 82 K=1,NEIGH	SIG2 221
	KK = NR + NINS + K	SIG2 222
	KKK=NG+KK	SIG2 223
	VARNE=0.0	SIG2 224
82	WOT6,908,K,FU(KK),VARNE,GR(1,KK),VARX(KK),GR(2,KK),VARX(KKK)	SIG2 225
	WOT6,909	SIG2 226
	DO 86 K=1,NEIGH	SIG2 227
	KK = NR + NINS + K	SIG2 228
	LK=NG+KK	SIG2 229
	VARNE=0.0	SIG2 230
	KV= NRT+KK	SIG2 231
	KVG =KV+ NG	SIG2 232
86	WOT6,908, K,GU(KK),VARNE,GI(1,KK),VARX(KV),GI(2,KK),VARX(KVG)	SIG2 233
87	IF(NC) 130,131,130	SIG2 234
131	MT=NC	SIG2 235
	NC = MT+ NLIN	SIG2 236
	DO 88 LR =1,2	SIG2 237
	DO 89 K=1,NRT	SIG2 238
		SIG2 239

	FU(K) = FU(K) + R(KK)	SIG2 180
	GU(K) = GU(K) + R(KJ)	SIG2 181
	LTE OUTPUT	SIG2 182
33	WOT 6, 906	SIG2 183
	WOT6,907	SIG2 184
	DO 36 K=1,NR	SIG2 185
	KK=NG+K	SIG2 186
	IF(NC) 34,34,35	SIG2 187
34	II=NX+1	SIG2 188
	GO TO 36	SIG2 189
35	II=2*NG +K	SIG2 190
36	WOT6,908,K,FU(K),VARX(II),GR(1,K),VARX(K),GR(2,K),VARX(KK)	SIG2 191
	IF (NINS) 120,120,121	SIG2 192
121	II = NX + 1	SIG2 193
	DO 122 KK = 1,NINS	SIG2 194
	K = NR + KK	SIG2 195
	KS = NG +K	SIG2 196
122	WOT6,908, K,FU(K),VARX(II),GR(1,K),VARX(K),GR(2,K),VARX(KS)	SIG2 197
20	WOT 6, 909	SIG2 198
	DO 81 K =1,NR	SIG2 199
	KK = NRT + K	SIG2 200
	KKG = NG + KK	SIG2 201
	VARI = VARX(KK)	SIG2 202
	VARO = VARX(KKG)	SIG2 203
	IF (NC) 38,38,39	SIG2 204
38	II = NX + 1	SIG2 205
	GO TO 81	SIG2 206
39	II= 2*NG+ NR + K	SIG2 207
81	WOT6, 908,K,GU(K),VARX(II),GI(1,K),VARI,GI(2,K),VARO	SIG2 208
	IF (NINS) 123,123,124	SIG2 209
124	II = NX + 1	SIG2 210
	DO 125 KN = 1,NINS	SIG2 211
	K = NR + KN	SIG2 212
	KK = NRT + K	SIG2 213
	KKG = NG + KK	SIG2 214
	VARI = VARX(KK)	SIG2 215
	VARO = VARX(KKG)	SIG2 216
125	WOT6, 908,K,GU(K),VARX(II),GI(1,K),VARI,GI(2,K),VARO	SIG2 217
123	IF (NEIGH) 87,87,37	SIG2 218
37	WOT6, 910	SIG2 219
	WOT6,907	SIG2 220
	DO 82 K=1,NEIGH	SIG2 221
	KK = NR + NINS + K	SIG2 222
	KKK=NG+KK	SIG2 223
	VARNE=0.0	SIG2 224
82	WOT6,908,K,FU(KK),VARNE,GR(1,KK),VARX(KK),GR(2,KK),VARX(KKK)	SIG2 225
	WOT6,909	SIG2 226
	DO 86 K=1,NEIGH	SIG2 227
	KK = NR + NINS + K	SIG2 228
	LK=NG+KK	SIG2 229
	VARNE=0.0	SIG2 230
	KV= NRT+KK	SIG2 231
	KVG =KV+ NG	SIG2 232
86	WOT6,908, K,GU(KK),VARNE,GI(1,KK),VARX(KV),GI(2,KK),VARX(KVG)	SIG2 233
87	IF(NC) 130,131,130	SIG2 234
131	MT=NC	SIG2 235
	NC = MT+ NLIN	SIG2 236
	DO 88 LR =1,2	SIG2 237
	DO 89 K=1,NRT	SIG2 238
		SIG2 239

PUNCH PARAMETERS AFTER A LINEAR ITERATION		SIG2 240
89	PUNCH 1000,FU(K),GU(K),GR(LR,K),GI(LR,K),NC,K,LR,A(13),A(14)	SIG2 241
C		SIG2 242
88	CONTINUE	SIG2 243
130	IF(KNEG) 55, 55, 50	SIG2 244
50	WOT6,911	SIG2 245
	ICOS=2*NRT	SIG2 246
	DO 54 K=1,KNEG	SIG2 247
	KK=ICOS + K	SIG2 248
	KKK= KK + NG	SIG2 249
54	WOT6, 913, K,AC(1,K),VARX(KK),AC(2,K),VARX(KKK)	SIG2 250
55	IF(KPOS) 61,61,56	SIG2 251
56	WOT6,912	SIG2 252
	ICOS= 2*NRT +KNEG	SIG2 253
	DO 60 K=1,KPOS	SIG2 254
	KK=ICOS +K	SIG2 255
	KKK= KK + NG	SIG2 256
60	WOT6,913,K,BC(1,K),VARX(KK),BC(2,K),VARX(KKK)	SIG2 257
C		SIG2 258
C	PUNCH CALCULATED ERRORS FOR INPUT TO THE ROUTINE LIST	SIG2 259
C		SIG2 260
61	IF(MT) 110,110,79	SIG2 261
110	LR=1	SIG2 262
70	DO 69 K=1,NRT	SIG2 263
	IF(K-NR) 62,62,68	SIG2 264
62	EN= FU(K)	SIG2 265
	CALL WIDTH(EN,HR)	SIG2 266
	IF(NLIN) 63,63,64	SIG2 267
63	ERMU=HR	SIG2 268
	ERNU=ERMU	SIG2 269
	GO TO 65	SIG2 270
64	KV=2*NG+K	SIG2 271
	KKV=KV+NR	SIG2 272
	ERMU=HR**2 + VARX(KV)**2	SIG2 273
	ERNU=HR**2 + VARX(KKV)**2	SIG2 274
	ERMU = SQRT(ERMU)	SIG2 275
	ERNU=SQRT(ERNU)	SIG2 276
65	IF(LR-1) 66,66,67	SIG2 277
66	LK=K	SIG2 278
	LKK = NRT + LK	SIG2 279
	GO TO 69	SIG2 280
67	LK=NG+K	SIG2 281
	LKK = NRT + LK	SIG2 282
	GO TO 69	SIG2 283
68	ERMU=0.0	SIG2 284
	ERNU=0.0	SIG2 285
	GO TO 65	SIG2 286
69	PUNCH 1000, VARX(LK), VARX(LKK), ERMU, ERNU, NC, K, LR, A(13), A(14)	SIG2 287
	2 14)	SIG2 288
	LR=LR+1	SIG2 289
	IF(LR-2) 70,70,71	SIG2 290
71	LR=1	SIG2 291
78	IF(KNEG) 74,74,72	SIG2 292
72	DO 73 K=1,KNEG	SIG2 293
73	PUNCH 1001, AC(LR,K),NC,K,LR,A(13),A(14)	SIG2 294
74	IF(KPOS) 77,77,75	SIG2 295
75	DO 76 K=1,KPOS	SIG2 296
76	PUNCH 1001, BC(LR,K),NC,K,LR,A(13),A(14)	SIG2 297
77	LR=LR+1	SIG2 298
	IF(LR-2) 78,78,79	SIG2 299

79	CONTINUE	SIG2 300
	NC = MT	SIG2 301
	RETURN	SIG2 302
900	FORMAT (3H0 I 60X, 6HQ(I,J) 56X, 4HR(I))	SIG2 303
901	FORMAT (1H0 12, 7E16.8/(3X,7E16.8))	SIG2 304
902	FORMAT (118X, E15.8)	SIG2 305
903	FORMAT (13H0DETERMINANT=E14.8, 5X, 4HINV=12)	SIG2 306
904	FORMAT (1H057X, 7HINVERSE)	SIG2 307
905	FORMAT (5H- RS=E15.8, 4H GF=E15.8, 8H CHI SQ=E15.8, 4H NR=15)	SIG2 308
906	FORMAT(///22X,20HRESONANCE PARAMETERS21X,7HFISSION21X,7HCAPTURE//2	SIG2 309
	22X,20HINSIDE RESONANCES //)	SIG2 310
907	FORMAT(1H012X,2HMU17X,3HDMU18X,3HGRF16X,4HDGRF15X,3HGRC16X,4HDGRC/	SIG2 311
	2)	SIG2 312
909	FORMAT(1H012X,2HNU17X,3HDNU18X,3HGIF16X,4HDGIF15X,3HGIC16X,4HDGIC/	SIG2 313
	2)	SIG2 314
908	FORMAT(I3,6E18.6//)	SIG2 315
910	FORMAT(///22X,22HNEIGHBORING RESONANCES//)	SIG2 316
911	FORMAT(///20H-CONSTANTS A FISSIION20X,20H-CONSTANTS A CAPTURE/10X,1	SIG2 317
	2HA9X,2HDA18X,1HA9X,2HDA/)	SIG2 318
912	FORMAT(///20H-CONSTANTS B FISSIION20X,20H-CONSTANTS B CAPTURE/10X,1	SIG2 319
	2HA9X,2HDB18X,1HA9X,2HDB/)	SIG2 320
913	FORMAT(I3,4E15.7)	SIG2 321
000	FORMAT (4E15.7, I3, I5, I3, 1X, A6, A2)	SIG2 322
001	FORMAT (E15.7, 45X, I3, I5, I3, 1X, A6, A2)	SIG2 323
	END	SIG2 324

SUBROUTINE TEST (ITEST)

	TEST	1
	TEST	2
TEST COMPATIBILITY OF LINEAR FIT AND CONVERGENCE OF NON-LINEAR --	TEST	3
	TEST	4
DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),	TEST	5
2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(	TEST	6
3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)	TEST	7
4, DGR(2,20), DGI(2,20), ICHA(4), YFS(4), YCA(4), EV(4), A(28), SIGPA(2)	TEST	8
COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, TEST	TEST	9
2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV, TEST	TEST	10
3 A	TEST	11
COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC, TEST	TEST	12
2 NP, TEMP, ATM, CC, FACS, RS, GF, NRT, JWE, SIGPA, NLIN, NINS	TEST	13
GO TO (1,2), ITEST	TEST	14
C	TEST	15
C TEST COMPATIBILITY AFTER LINEAR FIT	TEST	16
C	TEST	17
1 K = 1	TEST	18
NINS = 0	TEST	19
3 ELO = FU(K) - 3.0*GU(K)	TEST	20
EHI = FU(K) + 3.0*GU(K)	TEST	21
SUM1 = 0.0	TEST	22
SUM2 = 0.0	TEST	23
SUMP = 0.0	TEST	24
I = 1	TEST	25
10 IF (E(I) - ELO) 11,14,12	TEST	26
11 I = I + 1	TEST	27
IF(I - NP) 10,10,13	TEST	28
12 IF (E(I) - EHI) 14,14,13	TEST	29
14 EN = E(I)	TEST	30
CALL SIGAP(EN)	TEST	31
SUM1 = SUM1 + ABSF(Y(I) - SIGPA(1)) / Y(I)	TEST	32
IP = NP + I	TEST	33
SUM2 = SUM2 + ABSF(Y(IP) - SIGPA(2)) / Y(IP)	TEST	34
SUMP = SUMP + 1.0	TEST	35
GO TO 11	TEST	36
13 FOL = .02*SUMP	TEST	37
IF (SUM1 - FOL) 15,15,16	TEST	38
15 IF (SUM2 - FOL) 100,100,16	TEST	39
16 K = K + 1	TEST	40
160 IF(K-NR) 3,3,50	TEST	41
100 IF (NOUT - 20) 101,104,50	TEST	42
101 NOUT = NOUT + 1	TEST	43
DL(NOUT) = FU(K)	TEST	44
DW(NOUT) = GU(K)	TEST	45
DGR (1,NOUT) = GR(1,K)	TEST	46
DGR (2,NOUT) = GR(2,K)	TEST	47
DGI (1,NOUT) = GI(1,K)	TEST	48
DGI (2,NOUT) = GI(2,K)	TEST	49
NINS = NINS + 1	TEST	50
KMIN=K+1	TEST	51
IF(KMIN-NR) 170,170,172	TEST	52
0 DO 171 KLEV=KMIN,NR	TEST	53
KKL=KLEV-1	TEST	54
FU(KKL)=FU(KLEV)	TEST	55
GU(KKL)=GU(KLEV)	TEST	56
GR(1,KKL)=GR(1,KLEV)	TEST	57
GR(2,KKL)=GR(2,KLEV)	TEST	58
GI(2,KKL)=GI(2,KLEV)	TEST	59

171	GI(1,KKL)=GI(1,KLEV)	TEST	60
	NR=NR-1	TEST	61
	NRT = NR+NEIGH	TEST	62
	GO TO 160	TEST	63
172	NR=NR-1	TEST	64
	NRT = NR+NEIGH	TEST	65
	IF(NR) 40,40,50	TEST	66
40	WOT 6, 519, NINS, NINS	TEST	67
	RETURN	TEST	68
104	WOT 6, 518, NINS	TEST	69
50	IF (NINS) 300,300,301	TEST	70
301	WOT 6, 517, NINS	TEST	71
	WOT6,516	TEST	72
	DO 302 L = 1,NINS	TEST	73
	KL = NOUT - NINS + L	TEST	74
	KR = NR + L	TEST	75
	FU(KR) = DL(KL)	TEST	76
	GU(KR) = DW(KL)	TEST	77
	DGR(1,KL) =0.0	TEST	78
	DGI(1,KL) =0.0	TEST	79
	DGR(2,KL) =0.0	TEST	80
	DGI(2,KL) =0.0	TEST	81
302	WOT 6, 515, L, FU(KR), GU(KR)	TEST	82
	NOUT = NOUT- NINS	TEST	83
300	RETURN	TEST	84
		TEST	85
		TEST	86
C	TEST(2)	TEST	87
C		TEST	88
C	TEST ON GRADIENT OF NON LINEAR PARAMETERS AFTER NON LINEAR FIT	TEST	89
C		TEST	90
2	K = 1	TEST	91
C	TEST ON GOODNESS OF MU AND NU	TEST	92
130	JTEST = 0	TEST	93
	IF (FU(K) - E(1)) 202,202,208	TEST	94
202	JTEST = JTEST + 1	TEST	95
	GO TO 210	TEST	96
208	IF (FU(K) - E(NP)) 210,202,202	TEST	97
210	IF (GU(K)) 206,206,211	TEST	98
206	JTEST = JTEST + 1	TEST	99
211	IF (JTEST) 116,220,116	TEST	100
220	ELO = FU(K) - 3.0*GU(K)	TEST	101
	EHI = FU(K) + 3.0*GU(K)	TEST	102
	SUM1 = 0.0	TEST	103
	SUM2 = 0.0	TEST	104
	SUMP = 0.0	TEST	105
	REWIND 9	TEST	106
	KP = 1	TEST	107
110	READ TAPE 9, DSIG(1), DSIG(2), (PHI(I), I = 1,NTOT)	TEST	108
	IF (E(KP) - ELO) 111,114,112	TEST	109
111	KP = KP + 1	TEST	110
	IF(KP-NP) 110,110,113	TEST	111
112	IF (E(KP) - EHI) 114,114,113	TEST	112
C	TEST OF SECOND ORDER CHANGES AROUND K-TH LEVEL	TEST	113
114	EN = E(KP)	TEST	114
	X = (FU(K) - EN) / GU(K)	TEST	115
	T = TEMP * EN / (ATM*GU(K)*GU(K))	TEST	116
	IF (JS-1) 120,120,121	TEST	117
121	CALL WIDTH (EN,HR)	TEST	118
	T = T + .25 * HR * HR / (GU(K)*GU(K))	TEST	119
120	CALL SHAPE (2,X,T,PSI,CHI,2)	TEST	119

PART1 = GR(1,K) * PSI + GI(1,K) * CHI	TEST 120
PART2 = GR(2,K) * PSI + GI(2,K) * CHI	TEST 121
SUMP = SUMP + PSI	TEST 122
K1 = NRT + K	TEST 123
K2 = 2*NG + K	TEST 124
K3 = NR + K2	TEST 125
K4 = NG + K	TEST 126
K5 = NG + K1	TEST 127
K6 = 2*NR + K2	TEST 128
K7 = 2*NR + K3	TEST 129
PART1 = PART1 - R(K) * PHI(K) - R(K1) * PHI(K1) - R(K2) * PHI(K2)	TEST 130
2 - R(K3) * PHI(K3)	TEST 131
SUM1 = SUM1 + ABSF(PART1)	TEST 132
PART2 = PART2 - R(K4) * PHI(K4) - R(K5) * PHI(K5) - R(K2) * PHI(K6) -	TEST 133
2 R(K3) * PHI(K7)	TEST 134
SUM2 = SUM2 + ABSF(PART2)	TEST 135
GO TO 111	TEST 136
113 FOL = .10 * SUMP	TEST 137
XG = ABSF(GR(1,K))	TEST 138
YG = ABSF(GR(2,K))	TEST 139
IF (SUM1 / XG - FOL) 115,115,116	TEST 140
115 IF (SUM2/YG - FOL) 200,200,116	TEST 141
C TEST INCREMENTS AGAINST VARIANCES	TEST 142
116 MI = 2*NG + K	TEST 143
NI = NR + MI	TEST 144
XMU = ABSF(FU(K) - PAM(MI))	TEST 145
XNU = ABSF(GU(K) - PAM(NI))	TEST 146
IF (XMU - VARX(MI)) 122,122,123	TEST 147
123 FU(K) = PAM(MI) + VARX(MI) * (FU(K) - PAM(MI)) / XMU	TEST 148
122 IF (XNU - VARX(NI)) 124,124,125	TEST 149
125 GU(K) = PAM(NI) + VARX(NI) * (GU(K) - PAM(NI)) / XNU	TEST 150
124 FU(K) = PAM(MI) + (FU(K) - PAM(MI)) / 10.0	TEST 151
126 GU(K) = PAM(NI) + (GU(K) - PAM(NI)) / 10.0	TEST 152
IF (GU(K)) 126,126,200	TEST 153
200 K=K+1	TEST 154
IF (K - NR) 130,130,150	TEST 155
C RESTORE NINS INTO NR BEFORE LINEAR CASE	TEST 156
150 IF (NINS) 400,400,401	TEST 157
401 KN = 1	TEST 158
KKN = NR + KN	TEST 159
KR = 1	TEST 160
402 IF (FU(KKN) - FU(KR)) 404,404,403	TEST 161
403 KR = KR + 1	TEST 162
IF (KR - NR) 402,402,410	TEST 163
404 PAM(KN) = FU(KKN)	TEST 164
PAM (KKN) = GU(KKN)	TEST 165
JR = KKN	TEST 166
JJR = NR	TEST 167
405 FU(JR) = FU(JJR)	TEST 168
GU(JR) = GU(JJR)	TEST 169
JR = JR - 1	TEST 170
JJR = JJR - 1	TEST 171
IF (JJR - KR) 406,405,405	TEST 172
406 FU(KR) = PAM(KN)	TEST 173
GU(KR) = PAM(KKN)	TEST 174
NR = NR + 1	TEST 175
NINS = NINS - 1	TEST 176
GO TO 150	TEST 177
410 NR = NR + NINS	TEST 178
NINS = 0	TEST 179

b

400	NT = NT - 1	TEST 180
	IF (NT) 151,152,152	TEST 181
151	NT = 0	TEST 182
152	NLIN = NLIN + 1	TEST 183
	WOT 6, 520, NLIN, NLIN	TEST 184
	RETURN	TEST 185
515	FORMAT (15, 2E20.8)	TEST 186
516	FORMAT (1H0, 4X, 1HK, 12X, 2HMU, 18X, 2HNU/)	TEST 187
517	FORMAT (14H-NOTE--- THE, I3, 49H LEVELS LISTED BELOW WILL BE FITTEST 188	
	2TED ONLY LINEARLY//)	TEST 189
518	FORMAT (11H-NOTE--- I3, 68H LEVELS SHOUL) BE FITTED LINEARLY, BUTEST 190	
	2T STORAGE SPACE IS NOT ADEQUATE//)	TEST 191
519	FORMAT (11H-NOTE--- I3, 48H LEVELS SHOULD BE FITTED LINEARLY, BUTEST 192	
	2T SINCE NR= I3, 30H, CALCULATION DOES NOT PROCEED//)	TEST 193
520	FORMAT (58HONOTE . . . CALCULATION PROCEEDS WITH LINEAR FIT NUMTEST 194	
	2BER I3, 6H AFTER I3, 17H NON-LINEAR FITS./)	TEST 195
	END	TEST 196

b

SUBROUTINE PHIDO (EN)

FORM VECTOR PHI NEEDED TO CONSTRUCT THE NORMAL EQUATIONS

DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),
 2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(PHDO
 3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)PHDO
 4, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)PHDO
 COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, PHDO
 2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV,PHDO
 3 A PHDO
 COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,PHDO
 2 NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS PHDO

C COMMON TO BOTH LINEAR AND NON LINEAR CASES
 C

IF (NC) 50,50,51
 51 NSTOR = NR
 NR = NR + NINS
 50 DO 4 K = 1,NR
 X = (FU(K) - EN) / GU(K)
 T = TEMP * EN / (GU(K)*GU(K)*ATM)
 GO TO (3,2), JS
 2 CALL WIDTH (EN,HR)
 T = T + .25 * HR * HR / (GU(K) * GU(K))
 3 CALL SHAPE (2,X,T,PSI,CHI,2)
 PHI(K) = PSI
 NK = NRT + K
 4 PHI(NK) = CHI
 IF (NEIGH) 7,7,5
 5 DO 6 K = 1,NEIGH
 NK = NR + K
 BW = (FU(NK) - EN)**2 + GU(K)**2
 NNK = NRT + NK
 PHI(NK) = 1.0 / BW
 6 PHI(NNK) = (FU(NK) - EN) / BW
 7 IF (KNEG) 10,10,8
 8 DO 9 K = 1,KNEG
 NK = 2 * NRT + K
 9 PHI(NK) = EN / (EN**K)
 10 IF (KPOS) 13,13,11
 11 DO 12 K = 1,KPOS
 NK = 2 * NRT + KNEG + K
 12 PHI(NK) = E**K
 13 DO 14 K = 1,NG
 KK = NG + K
 14 PHI(KK) = PHI(K)
 IF (NC) 1,1,15
 1 CONTINUE
 RETURN

C
 C NON LINEAR CASES ONLY

15 NR = NSTOR
 DO 33 K = 1,NR
 X = (FU(K) - EN) / GU(K)
 T = TEMP * EN / (GU(K) * GU(K) * ATM)
 GO TO (29,28), JS
 28 CALL WIDTH (EN,HR)

T = T + .25 * HR * HR / (GU(K) * GU(K))

PHDO 60

CALL SHAPE (2,X,T,PSI,CHI,2)

PHDO 61

IF (TEMP) 30,30,31

PHDO 62

30 DEPSI = -2.0 * PSI * CHI

PHDO 63

DECHI = DEPSI * X + PSI

PHDO 64

GO TO 32

PHDO 65

31 DEPSI = (CHI - X * PSI) / (2.0 * T)

PHDO 66

DECHI = (1.0 - PSI - X * CHI) / (2.0 * T)

PHDO 67

32 TEPSI = DECHI - PSI

PHDO 68

TECHI = -DEPSI-CHI

PHDO 69

KM = 2 * NG + K

PHDO 70

KN = 2 * NG + NR + K

PHDO 71

PHI(KM) = (GR(1,K) * DEPSI + GI(1,K) * DECHI) / GU(K)

PHDO 72

PHI(KN) = (-GR(1,K) * TEPSI + GI(1,K) * TECHI) / GU(K)

PHDO 73

KKM = 2 * NR + KM

PHDO 74

KKN = 2 * NR + KN

PHDO 75

PHI(KKM) = (GR(2,K) * DEPSI + GI(2,K) * DECHI) / GU(K)

PHDO 76

33 PHI(KKN) = (-GR(2,K) * TEPSI + GI(2,K) * TECHI) / GU(K)

PHDO 77

RETURN

PHDO 78

END

PHDO 79

SUBROUTINE SIGAP (EN)

		SGAP	1
		SGAP	2
C	CALCULATE DOPPLER AND RESOLUTION BROADENED CROSS SECTIONS VALUES	SGAP	3
C	FROM GIVEN PARAMETERS-----	SGAP	4
		SGAP	5
	DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),	SGAP	6
2	W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(SGAP		7
3	62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)SGAP		8
4	, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)SGAP		9
	COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, SGAP		10
2	CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV,SGAP		11
3	A	SGAP	12
	COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,SGAP		13
2	NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS	SGAP	14
	CONS(1) = 0.0	SGAP	15
	CONS(2) = 0.0	SGAP	16
	IF (KNEG) 3,3,1	SGAP	17
1	DO 2 LR = 1,2	SGAP	18
	DO 2 K = 1,KNEG	SGAP	19
2	CONS(LR) = CONS(LR) + AC(LR,K) * EN / (EN**K)	SGAP	20
3	IF (KPOS) 6,6,4	SGAP	21
4	DO 5 LR = 1,2	SGAP	22
	DO 5 K = 1,KPOS	SGAP	23
5	CONS(LR) = CONS(LR) + BC(LR,K) * (EN**K)	SGAP	24
6	DO 7 K = 1,NRT	SGAP	25
	IF (K-NR) 8,8,9	SGAP	26
8	X = (FU(K) - EN) / GU(K)	SGAP	27
	T = TEMP * EN / ((GU(K)**2) * ATM)	SGAP	28
	GO TO (13,14), JS	SGAP	29
14	CALL WIDTH (EN,HR)	SGAP	30
	T = T + .25 * (HR**2) / (GU(K)**2)	SGAP	31
13	CALL SHAPE (2,X,T,PSI,CHI,2)	SGAP	32
	PSIN = PSI	SGAP	33
	CHIN = CHI	SGAP	34
	GO TO 10	SGAP	35
9	BW = (FU(K) - EN)**2 + GU(K)**2	SGAP	36
	PSIN = 1.0/BW	SGAP	37
	CHIN = (FU(K) - EN) / BW	SGAP	38
10	DO 11 LR = 1,2	SGAP	39
	CONS(LR) = CONS(LR) + GR(LR,K)*PSIN + GI(LR,K)*CHIN	SGAP	40
11	SIGPA(LR) = CONS(LR)	SGAP	41
7	CONTINUE	SGAP	42
	CALL SIGOUT(EN)	SGAP	43
	DO 12 LR = 1,2	SGAP	44
12	SIGPA(LR) = SIGPA(LR) + DSIG(LR)	SGAP	45
	RETURN	SGAP	46
	END	SGAP	47

		SGIN	1
	CALCULATE DOPPLER BROADENED CROSS SECTION VALUES FROM GIVEN PARAM	SGIN	2
C	ETERS	SGIN	3
	SUBROUTINE SIGIN (EN)	SGIN	4
	DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),	SGIN	5
	2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(SGIN	SGIN	6
	3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)SGIN	SGIN	7
	4, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)SGIN	SGIN	8
	COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, SGIN	SGIN	9
	2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV,SGIN	SGIN	10
	3 A	SGIN	11
	COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,SGIN	SGIN	12
	2 NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS	SGIN	13
	CONS(1) = 0.0	SGIN	14
	CONS(2) = 0.0	SGIN	15
	IF (KNEG) 3,3,1	SGIN	16
1	DO 2 LR = 1,2	SGIN	17
	DO 2 K = 1,KNEG	SGIN	18
2	CONS(LR) = CONS(LR) + AC(LR,K) * EN / (EN**K)	SGIN	19
3	IF (KPOS) 6,6,4	SGIN	20
4	DO 5 LR = 1,2	SGIN	21
	DO 5 K = 1,KPOS	SGIN	22
5	CONS(LR) = CONS(LR) + BC(LR,K) * (EN**K)	SGIN	23
6	DO 7 K = 1,NRT	SGIN	24
	IF (K-NR) 8,8,9	SGIN	25
8	X = (FU(K) - EN) / GU(K)	SGIN	26
	T = TEMP * EN / ((GU(K)**2) * ATM)	SGIN	27
	CALL SHAPE (2,X,T,PSI,CHI,2)	SGIN	28
	PSIN = PSI	SGIN	29
	CHIN = CHI	SGIN	30
	GO TO 10	SGIN	31
9	BW = (FU(K) - EN)**2 + GU(K)**2	SGIN	32
	PSIN = 1.0/BW	SGIN	33
	CHIN = (FU(K) - EN) / BW	SGIN	34
10	DO 11 LR = 1,2	SGIN	35
	CONS(LR) = CONS(LR) + GR(LR,K)*PSIN + GI(LR,K)*CHIN	SGIN	36
11	SIGCA(LR) = CONS(LR)	SGIN	37
7	CONTINUE	SGIN	38
	NJS = JS	SGIN	39
	JS = 1	SGIN	40
	CALL SIGOUT(EN)	SGIN	41
	JS = NJS	SGIN	42
	DO 12 LR = 1,2	SGIN	43
12	SIGCA(LR) = SIGCA(LR) + DSIG(LR)	SGIN	44
	RETURN	SGIN	45
	END	SGIN	46

SUBROUTINE SIGPLT

C TABULATE VALUES OF SIGMAS AND DEVIATIONS AFTER COMPLETION OF ALL
C WANTED ITERATIONS---

DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),
2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(SGPT
3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)SGPT
4, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)SGPT
COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, SGPT
2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV,SGPT
3 A SGPT
COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,SGPT
2 NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS SGPT

C PLOT FISSION CROSS SECTION

2 LINES = 0 SGPT
CALL SKIP(2) SGPT
WOT 6, 921 SGPT
INDE = 1 SGPT
DO 1 KP = 1,NP SGPT
EN = E(KP) SGPT
READ TAPE 9, DSIG(1), DSIG(2), (PHI(I), I = 1,NTOT) SGPT
SD = 0.0 SGPT
YAPP = DSIG(1) SGPT
IMIN = 2 * NG + 1 SGPT
DO 3 I = 1,NG SGPT
YAPP = YAPP + R(I) * PHI(I) SGPT
DO 3 J = 1,NG SGPT
3 SD = SD + Q(I,J) * PHI(I) * PHI(J) SGPT
IF (NC) 2,2,4 SGPT
4 DO 5 I = IMIN,NX SGPT
YAPP=YAPP+ R(I)*PHI(I) SGPT
DO 5 J=IMIN,NX SGPT
5 SD=SD+Q(I,J)*PHI(I)*PHI(J) SGPT
DO 20 I=IMIN,NX SGPT
DO 20 J=1,NG SGPT
20 SD=SD+(Q(I,J)+Q(J,I))*PHI(I)*PHI(J) SGPT

2 CALL SIGIN(EN) SGPT
SIGMA = SIGCA(1) * CC SGPT
YAPP = YAPP * CC SGPT
YE = Y(KP) * CC SGPT
8 RESID = YE - YAPP SGPT
PCER = RESID * 100. / YE SGPT
QW =ABSF(SD) SGPT

SD = SQRT(QW) * GF * CC SGPT
LINES = LINES + 1 SGPT
IF (LINES - 50) 7,7,6 SGPT

6 CALL SKIP(2) SGPT
WOT 6, 921 SGPT
LINES = 1 SGPT
7 WOT 6, 920, KP, EN, YE, YAPP, SIGMA, RESID, PCER, SD SGPT
1 Y(KP) = SIGCA(2) SGPT

C PLOT CAPTURE CROSS SECTION

REWIND 9 SGPT
LINES = 0 SGPT
CALL SKIP(2) SGPT
WOT 6, 921 SGPT
INDE = 2 SGPT
DO 10 KP = 1,NP SGPT

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59

	NK = NP + KP	SGPT 60
	EN = E(KP)	SGPT 61
	SD = 0.0	SGPT 62
	SIGMA = Y(KP) * CC	SGPT 63
	IMIN = 2 * NG + 1	SGPT 64
	READ TAPE 9, DSIG(1), DSIG(2), (PHI(I), I = 1,NTOT)	SGPT 65
	YAPP = DSIG(2)	SGPT 66
	DO 11 K = 1,NG	SGPT 67
	KK = NG + K	SGPT 68
	YAPP = YAPP + R(KK) * PHI(KK)	SGPT 69
	DO 11 J = 1,NG	SGPT 70
	JJ = NG + J	SGPT 71
11	SU = SD + Q(KK,JJ) * PHI(KK) * PHI(JJ)	SGPT 72
	IF (NC) 12,12,14	SGPT 73
14	DO 15 I = IMIN,NX	SGPT 74
	II = 2 * NR + I	SGPT 75
	YAPP = YAPP + R(I) * PHI(II)	SGPT 76
	DO 15 J = IMIN,NX	SGPT 77
	JJ = 2 * NR + J	SGPT 78
15	SD = SD + Q(I,J) * PHI(II) * PHI(JJ)	SGPT 79
	JMIN = NG + 1	SGPT 80
	JMAX = 2 * NG	SGPT 81
	DO 21 J=JMIN,JMAX	SGPT 82
	DO 21 K=IMIN,NX	SGPT 83
	KK=K+2*NR	SGPT 84
21	SD=SD+(Q(J,K)+Q(K,J))*PHI(J)*PHI(KK)	SGPT 85
12	YE = Y(NK) * CC	SGPT 86
	YAPP = YAPP * CC	SGPT 87
	RESID = YE - YAPP	SGPT 88
	PCER = RESID * 100.0 / YE	SGPT 89
	QW =ABSF(SD)	SGPT 90
	SD = SQRT(QW) * GF * CC	SGPT 91
	LINES = LINES + 1	SGPT 92
	IF (LINES - 50) 17,17,16	SGPT 93
16	CALL SKIP(2)	SGPT 94
	WOT 6, 921	SGPT 95
	LINES = 1	SGPT 96
17	WOT 6, 920, KP, EN, YE, YAPP, SIGMA, RESID, PCER, SD	SGPT 97
10	CONTINUE	SGPT 98
	CALCULATE THERMAL VALUES	SGPT 99
	EN = .0253	SGPT 100
	SEN = SQRT(EN)	SGPT 101
	CALL SIGIN(EN)	SGPT 102
	FISTH = CC * SIGCA(1) / SEN	SGPT 103
	CAPTH = CC * SIGCA(2) / SEN	SGPT 104
	CONTFS = AC(1,1) * CC / SEN	SGPT 105
	CONTCA = AC(2,1) * CC / SEN	SGPT 106
	WOT 6, 922, FISTH, CONTFS	SGPT 107
	WOT 6, 923, CAPTH, CONTCA	SGPT 108
920	FORMAT (I5, 7(1X, F15.4))	SGPT 109
921	FORMAT(1H014X,6HENERGY6X,10HEXP. SIGMA6X,10HAPP. SIGMA6X,10HCAL. SIGMA7X,9HDEVIATION3X,13HPERCENT ERROR2X,14HSTD. DEVIATION)	SGPT 110
922	FORMAT (26HOAT E=.0253,SIGMA FISSION=E10.4, 24H AND CONTRIBUTION	SGPT 111
	2F A1=E10.4//)	SGPT 112
923	FORMAT (26HOAT E=.0253,SIGMA CAPTURE=E10.4, 24H AND CONTRIBUTION	SGPT 113
	2F A1=E10.4//)	SGPT 114
	RETURN	SGPT 115
	END	SGPT 116
		SGPT 117

SUBROUTINE SIGOUT (EN)

CALCULATE CONTRIBUTION OF THE NOUT LEVELS, DSIG-----

		SIGU	1
		SIGU	2
		SIGU	3
		SIGU	4
	DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),	SIGU	5
2	W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(SIGU	SIGU	6
3	62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)	SIGU	7
4	DGR(2,20), DGI(2,20), ICHA(4), YFS(4), YCA(4), EV(4), A(28), SIGPA(2)	SIGU	8
	COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, SIGU	SIGU	9
2	CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV, SIGU	SIGU	10
3	A	SIGU	11
	COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC, SIGU	SIGU	12
2	NP, TEMP, ATM, CC, FACS, RS, GF, NRT, JWE, SIGPA, NLIN, NINS	SIGU	13
	DSIG(1) = 0.0	SIGU	14
	DSIG(2) = 0.0	SIGU	15
	IF (NOUT) 8,8,1	SIGU	16
1	DO 8 K = 1, NOUT	SIGU	17
	IF (DL(K)) 2,2,3	SIGU	18
2	BW = (DL(K) - EN) ** 2 + DW(K) ** 2	SIGU	19
	PSID = DW(K)**2 / BW	SIGU	20
	CHID = PSID * (DL(K) - EN) / DW(K)	SIGU	21
	GO TO 4	SIGU	22
3	X = (DL(K) - EN) / DW(K)	SIGU	23
	T = TEMP * EN / ((DW(K)**2) * ATM)	SIGU	24
	GO TO (5,6), JS	SIGU	25
6	CALL WIDTH (EN, HR)	SIGU	26
	T = T + .25 * (HR**2) / (DW(K)**2)	SIGU	27
5	CALL SHAPE (2, X, T, PSI, CHI, 2)	SIGU	28
	PSID = PSI	SIGU	29
	CHID = CHI	SIGU	30
4	DO 7 LR = 1, 2	SIGU	31
7	DSIG(LR) = DSIG(LR) + DGR(LR, K) * PSID + DGI(LR, K) * CHID	SIGU	32
8	CONTINUE	SIGU	33
	RETURN	SIGU	34
	END	SIGU	35

SUBROUTINE WETA(IWE,KP)

EVALUATE STATISTICAL WEIGHTS NEEDED FOR THE NORMAL EQUATIONS--

DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000), WETA

2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(WETA

3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)WETA

4, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)WETA

COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, WETA

2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV,WETA

3 A WETA

COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,WETA

2 NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS WETA

C IWE = 1 ONLY INITIALLY, OTHERWISE IWE = 0 WETA

IF (IWE) 21,100,21 WETA

21 SUMA=0.0 WETA

SUMC = 0.0 WETA

KP = 1 WETA

GO TO 102 WETA

13 KP = KP + 1 WETA

SUMA=SUMA+1.0/WX WETA

SUMC=SUMC+1.0/WY WETA

IF (NP-KP) 16,102,102 WETA

16 FAWA = FLOATF(NP) / SUMA WETA

FAWC = FLOATF(NP) / SUMC WETA

RETURN WETA

100 GO TO (101,102), JWE WETA

101 W(1) = 1.0 WETA

W(2) = 1.0 WETA

RETURN WETA

102 EN = E(KP) WETA

S = SQRT(EN) WETA

IF (EN - .7989) 1,1,2 WETA

1 NCDS = 8 WETA

GO TO 20 WETA

2 IF (EN - 1.4115) 3,3,4 WETA

3 NCDS = 7 WETA

GO TO 20 WETA

4 IF (EN - 3.1448) 5,5,6 WETA

5 NCDS = 6 WETA

GO TO 20 WETA

6 IF (EN - 6.5330) 7,7,8 WETA

7 NCDS = 5 WETA

GO TO 20 WETA

8 IF (EN - 12.1865) 9,9,10 WETA

9 NCDS = 4 WETA

GO TO 20 WETA

10 IF (EN - 27.6164) 11,11,12 WETA

11 NCDS = 3 WETA

GO TO 20 WETA

12 NCDS = 2 WETA

20 NKP = NP + KP WETA

WX = 2.6 * (2.0**NCDS) * (Y(KP) + .5/CC)/CC WETA

WY = 1.5 * (2.0**NCDS) * (35.0/CC + Y(NKP) + Y(KP)*.41) /CC WETA

IF (IWE) 13,22,13 WETA

22 W(1)=FAWA/WX WETA

W(2)=FAWC/WY WETA

RETURN WETA

END WETA

C	DESAUSSURE WIDTH	WDTH	1
	SUBROUTINE WIDTH (EN,HR)	WDTH	2
	IF (EN - .7989) 1,1,2	WDTH	3
1	NCDS = 8	WDTH	4
	GO TO 13	WDTH	5
2	IF (EN - 1.4115) 3,3,4	WDTH	6
3	NCDS = 7	WDTH	7
	GO TO 13	WDTH	8
4	IF (EN - 3.1448) 5,5,6	WDTH	9
5	NCDS = 6	WDTH	10
	GO TO 13	WDTH	11
6	IF (EN - 6.5330) 7,7,8	WDTH	12
7	NCDS = 5	WDTH	13
	GO TO 13	WDTH	14
8	IF (EN - 12.1865) 9,9,10	WDTH	15
9	NCDS = 4	WDTH	16
	GO TO 13	WDTH	17
10	IF (EN - 27.6164) 11,11,12	WDTH	18
11	NCDS = 3	WDTH	19
	GO TO 13	WDTH	20
12	NCDS = 2	WDTH	21
13	CDS3 = 39.10 + (4.0)**NCDS	WDTH	22
	DESA = CDS3 * (3.15E-10) * (EN**3) + (1.71E-06) * (EN**2)	WDTH	23
	HR = SQRT (DESA)	WDTH	24
	RETURN	WDTH	25
	END	WDTH	26

C	FORTRAN II SUBROUTINE SHAPE FOR THE EVALUATION OF THE	SHPE	1
	DOPPLER LINE SHAPE FUNCTIONS PSI(X,T) AND CHI(X,T).	SHPE	2
C	F.T.ADLER, Y.D.NALIBOFF, J.L.DAVIDOV	SHPE	3
	SUBROUTINE SHAPE (J,X,T,PSI,CHI,IIN)	SHPE	4
	FREQUENCY 1(0,50),3(0,1,50),6(1,0,50),7(1,0,50),11(50,0),16(4,1,0)	SHPE	5
	FREQUENCY 210(1,0,1),23(1,0,1),250(1,0,1),260(1,0,1),26(1,0,1),	SHPE	6
1	27(1,0,1),28(1,0,1),29(1,0,1),30(1,0,1),320(1,0,1),330(1,0,1),	SHPE	7
2	33(1,0,1),34(1,0,1),35(1,0,1),36(1,0,1),440(1,0,1),45(0,1,1)	SHPE	8
	DIMENSION A(6),CR(5),CI(5),PR(5),PI(5),PS(5),CH(5),RPS(5),RCH(5)	SHPE	9
	DIMENSION C1(2),C2(2),C3(2),C4(2),C5(2),C6(2)	SHPE	10
1	GO TO (2,3),J	SHPE	11
2	A(1)=2.5052367	SHPE	12
	A(2)=1.2831204	SHPE	13
	A(3)=0.2264718	SHPE	14
	A(4)=0.1306469	SHPE	15
	A(5)=-0.0202490	SHPE	16
	A(6)=0.0039132	SHPE	17
	CR(1)=0.	SHPE	18
	CR(2)=1.09147894	SHPE	19
	CR(3)=-2.30555691	SHPE	20
	CR(4)=-CR(2)	SHPE	21
	CR(5)=-CR(3)	SHPE	22
	CI(1)=1.32271498	SHPE	23
	CI(2)=1.29081178	SHPE	24
	CI(3)=1.17416680	SHPE	25
	CI(4)=CI(2)	SHPE	26
	CI(5)=CI(3)	SHPE	27
	PR(1)=0.	SHPE	28
	PR(2)=-10.9663038	SHPE	29
	PR(3)=24.240703	SHPE	30
	PR(4)=-PR(2)	SHPE	31
	PR(5)=-PR(3)	SHPE	32
	PI(1)=6.1043246	SHPE	33
	PI(2)=1.1317230	SHPE	34
	PI(3)=-78.663636	SHPE	35
	PI(4)=PI(2)	SHPE	36
	PI(5)=PI(3)	SHPE	37
	RPS(1)=0.21117643	SHPE	38
	RPS(2)=0.081635073	SHPE	39
	RPS(3)=-0.0050439914	SHPE	40
	RPS(4)=RPS(2)	SHPE	41
	RPS(5)=RPS(3)	SHPE	42
	RCH(1)=0.	SHPE	43
	RCH(2)=0.16088151	SHPE	44
	RCH(3)=-0.0098666871	SHPE	45
	RCH(4)=-RCH(2)	SHPE	46
	RCH(5)=-RCH(3)	SHPE	47
	C1(1)=1.	SHPE	48
	C2(1)=3.	SHPE	49
	C3(1)=15.	SHPE	50
	C4(1)=105.	SHPE	51
	C5(1)=945.	SHPE	52
	C6(1)=10395.	SHPE	53
	C1(2)=.33333333	SHPE	54
	C2(2)=.1	SHPE	55
	C3(2)=.23809524E-01	SHPE	56
	C4(2)=.46296296E-02	SHPE	57
	C5(2)=.75757575E-03	SHPE	58
	C6(2)=.10683761E-03	SHPE	59

TWOP=6.283185307	SHPE	60
RETURN	SHPE	61
3 IF (T) 5,4,5	SHPE	62
4 PSI=1./(1.+X**2)	SHPE	63
CHI=X*PSI	SHPE	64
RETURN	SHPE	65
5 X2=X*X	SHPE	66
B1=1.+X2	SHPE	67
B3=2.*T	SHPE	68
B4=4.*T	SHPE	69
RT=SQRT(B3)	SHPE	70
C16=X/RT	SHPE	71
C26=1./RT	SHPE	72
210 IF (ABSF(X)-1.) 21,21,22	SHPE	73
21 R=ABSF(C26)	SHPE	74
W=X2	SHPE	75
U=1.	SHPE	76
V=1.	SHPE	77
GO TO 23	SHPE	78
22 R=ABSF(C16)	SHPE	79
W=1./X2	SHPE	80
U=-1.	SHPE	81
V=X2	SHPE	82
23 IF (R-12.53) 30,24,24	SHPE	83
24 N=1	SHPE	84
B6=B3/B1**2*V	SHPE	85
250 IF (R-14.52) 37,25,25	SHPE	86
25 F1=0.	SHPE	87
F2=0.	SHPE	88
260 IF (R-18.24) 38,26,26	SHPE	89
26 IF (R-26.32) 39,27,27	SHPE	90
27 IF (R-54.40) 40,28,28	SHPE	91
28 IF (R-200.) 41,29,29	SHPE	92
29 IF (R-16384.) 42,43,43	SHPE	93
30 IF (R-.589) 31,31,48	SHPE	94
31 N=2	SHPE	95
B6=V/B4	SHPE	96
G=X/B3	SHPE	97
H=SQRT(T*3.1415927)	SHPE	98
E=EXP((1.-X2)/B4)/B3	SHPE	99
SI=SIN(G)	SHPE	100
CO=COS(G)	SHPE	101
320 IF (R-.456) 32,32,37	SHPE	102
32 F1=0.	SHPE	103
F2=0.	SHPE	104
330 IF (R-.318) 33,33,38	SHPE	105
33 IF (R-.191) 34,34,39	SHPE	106
34 IF (R-.084) 35,35,40	SHPE	107
35 IF (R-.0172) 36,36,41	SHPE	108
36 IF (R-.00017) 43,43,42	SHPE	109
37 F1=(((13.*W-286.)*W+1287.)*W-1716.)*W+715.)*W-78.)*W+1.)*C6(N)*	SHPE	110
1 B6	SHPE	111
F2=(((17.*W-78.)*W+715.)*W-1716.)*W+1287.)*W-286.)*W+13.)*C6(N)*B6	SHPE	112
38 F1=(((11.*W-165.)*W+462.)*W-330.)*W+55.)*W-1.)*U*C5(N)+F1)*B6	SHPE	113
F2=(((11.*W-55.)*W+330.)*W-462.)*W+165.)*W-11.)*U*C5(N)+F2)*B6	SHPE	114
39 F1=(((9.*W-84.)*W+126.)*W-36.)*W+1.)*C4(N)+F1)*B6	SHPE	115
F2=(((9.*W-36.)*W+126.)*W-84.)*W+9.)*C4(N)+F2)*B6	SHPE	116
40 F1=(((7.*W-35.)*W+21.)*W-1.)*U*C3(N)+F1)*B6	SHPE	117
F2=(((7.*W-21.)*W+35.)*W-7.)*U*C3(N)+F2)*B6	SHPE	118
41 F1=(((5.*W-10.)*W+1.)*C2(N)+F1)*B6	SHPE	119

	F2=((W-10.)*W+5.)*C2(N)+F2)*B6	SHPE 120
42	F1=((3.*W-1.)*U*C1(N)+F1)*B6	SHPE 121
	F2=((W-3.)*U*C1(N)+F2)*B6	SHPE 122
43	F1=(1.+F1)	SHPE 123
	F2=(1.+F2)	SHPE 124
440	IF (U) 45,44,44	SHPE 125
44	FF=F2	SHPE 126
	F2=F1	SHPE 127
	F1=FF	SHPE 128
45	IF (N-1) 46,46,47	SHPE 129
46	PSI=F2/B1	SHPE 130
	CHI=F1/B1*X	SHPE 131
	RETURN	SHPE 132
47	PSI=E*(CO*(H-F2)+X*SI*F1)	SHPE 133
	CHI=E*(-SI*(H-F2)+X*CO*F1)	SHPE 134
	RETURN	SHPE 135
48	IZ=1	SHPE 136
	B2=-2.*X*RT	SHPE 137
	DO 10 I=1,5	SHPE 138
6	IF (ABSF(X-CR(I)/CI(I))-0.005) 7,7,9	SHPE 139
7	IF (ABSF(CI(I)*RT-1.)-0.005) 8,8,9	SHPE 140
8	IZ=2	SHPE 141
	PS(I)=RPS(I)	SHPE 142
	CH(I)=RCH(I)	SHPE 143
	S1=0.	SHPE 144
	S2=0.	SHPE 145
	GO TO 10	SHPE 146
9	Q1=B1+B2*CR(I)+B3*(CR(I)**2-CI(I)**2)	SHPE 147
	Q2=(B2+B4*CR(I))*CI(I)	SHPE 148
	D1=PR(I)*Q1-PI(I)*Q2	SHPE 149
	D2=PI(I)*Q1+PR(I)*Q2	SHPE 150
	D=D1**2+D2**2	SHPE 151
	PS(I)=D2/D	SHPE 152
	CH(I)=(CR(I)*D2-CI(I)*D1)/D	SHPE 153
10	CONTINUE	SHPE 154
11	GO TO (12,19),IZ	SHPE 155
12	P16=A(1)	SHPE 156
	P26=0.	SHPE 157
	C262=C16*C26*2.	SHPE 158
	C162=C16**2-C26**2	SHPE 159
	C16N=C162	SHPE 160
	C26N=C262	SHPE 161
	I=2	SHPE 162
15	P16=P16+A(I)*C16N	SHPE 163
	P26=P26+A(I)*C26N	SHPE 164
16	IF (I-6) 17,18,18	SHPE 165
17	C1N=C162*C16N-C262*C26N	SHPE 166
	C2N=C262*C16N+C162*C26N	SHPE 167
	C16N=C1N	SHPE 168
	C26N=C2N	SHPE 169
	I=I+1	SHPE 170
	GO TO 15	SHPE 171
18	D3=P16**2+P26**2	SHPE 172
	S1=P16/(2.*RT*D3)	SHPE 173
	S2=(P26+X*P16)/(B4*D3)	SHPE 174
19	DO 20 I=1,5	SHPE 175
	S1=S1+PS(I)	SHPE 176
20	S2=S2+CH(I)	SHPE 177
	PSI=TWOPI*S1	SHPE 178
	CHI=X*PSI-RT*TWOPI*S2	SHPE 179

RETURN
END

SHPE 180
SHPE 181

SUBROUTINE SKIP--READ AND PRINT PAGE HEADINGS.

		SKIP	1
		SKIP	2
C	N.B.--DAT1 IS A SUBROUTINE WICH OBTAINS THE DATE FROM THE MONITOR	SKIP	3
C	AND STORES IT IN THE LOCATION OF THE ARGUMENT AND THE TWO FOLLOWING	SKIP	4
C	LOCATIONS--I.E. FORWARD IN CORE.	SKIP	5
C		SKIP	6
	SUBROUTINE SKIP(INIT)	SKIP	7
	DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),	SKIP	8
	2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(SKIP	9	
	3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)SKIP	10	
	4, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)SKIP	11	
	COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, SKIP	12	
	2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV,SKIP	13	
	3 A	SKIP	14
	COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,SKIP	15	
	2 NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS	SKIP	16
	DIMENSION DATE(3)	SKIP	17
	GO TO (10,30),INIT	SKIP	18
10	READ 20,(A(I), I=1,28)	SKIP	19
20	FORMAT (13A6,A2)	SKIP	20
	CALL DAT1(DATE(3))	SKIP	21
	IPAGE=0	SKIP	22
	RETURN	SKIP	23
30	IPAGE=IPAGE + 1	SKIP	24
	WOT6,40,(A(I),I=1,14),DATE(3),DATE(2),DATE(1),IPAGE,(A(I),I=15,28)SKIP	25	
40	FORMAT (1H13A6,A2,10X,3A6,7X,4HPAGE13/1H 13A6,A2,8X,26HPROGRAM TASKIP	26	
	2NDEM 14 JAN 1967)	SKIP	27
	RETURN	SKIP	28
	END	SKIP	29

	FUNCTION INVERT(DUMMY)	INVT	1
C	FUNCTION INVERT--PREPARE LEAST SQUARES MATRIX FOR DOUBLE PRECISION	INVT	2
C	INVERSION AND PARTIALLY COMPUTE VARIANCE OF PARAMETERS OBTAINED.	INVT	3
C		INVT	4
	DIMENSION Q(120,60), R(120), PIVOT(120), DET(2), Y(2000), E(1000),	INVT	5
	2 W(2), FU(30), GU(30), GR(2,30), GI(2,30), PAM(60), PHI(90), VARX(INVT	INVT	6
	3 62), CONS(2), DSIG(2), SIGCA(2), AC(2,3), BC(2,5), DL(20), DW(20)INVT	INVT	7
	4, DGR(2,20), DGI(2,20), ICHA(4),YFS(4),YCA(4),EV(4),A(28),SIGPA(2)INVT	INVT	8
	COMMON Q, R, PIVOT, DET, Y, E, W, FU, GU, GR, GI, PAM, PHI, VARX, INVT	INVT	9
	2 CONS, DSIG, SIGCA, AC, BC, DL, DW, DGR, DGI, ICHA, YFS, YCHA, EV,INVT	INVT	10
	3 A	INVT	11
	COMMON NR, NEIGH, NOUT, KNEG, KPOS, NT, JS, IPR, NG, NX, NTOT, NC,INVT	INVT	12
	2 NP, TEMP, ATM, CC, FACS, RS, GF,NRT,JWE, SIGPA, NLIN, NINS	INVT	13
	DIMENSION IPIVOT(60),INDEX(60,2)	INVT	14
	N = NX	INVT	15
	1 DO 2 I=1,NX	INVT	16
	Q(2*N,I)=Q(N,I)	INVT	17
	2 Q(2*N-1,I)=0.0	INVT	18
	R(2*N)=R(N)	INVT	19
	R(2*N-1)=0.0	INVT	20
	N=N-1	INVT	21
	IF(N) 3,3,1	INVT	22
	3 INVERT=DNV1(NX,120,1,Q,R,DET(2),PIVOT,IPIVOT,INDEX,60)	INVT	23
	DO 5 I=1,NX	INVT	24
	VARX(I)=ROUNDF(DSQRT(Q(2*I,I),PIVOT(2)))	INVT	25
	DO 4 N=1,NX	INVT	26
	4 Q(I,N)=ROUNDF(Q(2*I,N),Q(2*I-1,N))	INVT	27
	5 R(I)=ROUNDF(R(2*I),R(2*I-1))	INVT	28
	RETURN	INVT	29
	END	INVT	30
\$	SCATRE	INVT	31
\$	PUNCH OBJECT	ROUND000INVT	32
*	FUNCTION ROUND--ROUND DOUBLE PRECISION NUMBER TO SINGLE PRECISION	INVT	33
	ENTRY ROUND	INVT	34
ROUND	FRN	INVT	35
	TRA 1,4	INVT	36
	END	INVT	37

* UOI	DOUBLE	PRECISION SQRT		ROUN	1
	ENTRY	DSQRT		ROUN	2
	EVEN			ROUN	3
U	BSS	2	TEMP STORAGE	ROUN	4
DSQRT	CLA*	1,4		ROUN	5
	TMI	ERR	ERROR RETURN, X NEGATIVE	ROUN	6
	SXA	AA,1		ROUN	7
	ACL	=0177000000000	(P+128+127)*2**-8	ROUN	8
	XCL			ROUN	9
	LGL	8		ROUN	10
	ALS	27	((P/2)-1)+128 = NEW CHARACTERISTIC	ROUN	11
	STO	COMMON+1		ROUN	12
	TQP	**+2	MQ IS + IF P IS ODD, USE Y/2 FOR Y	ROUN	13
	LLS	1	MQ IS - IF P IS EVEN, USE Y	ROUN	14
	LLS	5	6 BITS OF Y USED FOR TABLE LOOKUP	ROUN	15
	PAX	0,1	OF INITIAL APPROXIMATION	ROUN	16
	TXL	A,1,15	TRA IF ARG NOT NORMAL	ROUN	17
	LLS	25	AC=Y/16 OR -Y/16	ROUN	18
	SLW	COMMON		ROUN	19
	VDH	TABLE+16,1,19	DIVIDE BY INITIAL APPROX. (X0/4)	ROUN	20
	LLS	16	MQ=Y/4X0 OR -Y/4X0	ROUN	21
	XCL		AC=Y/4X0, MAYBE P BIT IS 1	ROUN	22
	ADD	TABLE+16,1	AC=((Y/2X0)+(X0/2))/2=X1/2	ROUN	23
	STO	COMMON+2		ROUN	24
	CLA	COMMON	AC=Y/16	ROUN	25
	DVH	COMMON+2		ROUN	26
	XCA		AC=Y/8X1	ROUN	27
	ALS	2		ROUN	28
	ADD	COMMON+2	AC=(Y/2X1)+(X1/2)=X2, FIXED POINT	ROUN	29
	LRS	8		ROUN	30
	RQL	28		ROUN	31
	ADD	COMMON+1	X2/2, FLOATING POINT	ROUN	32
*				ROUN	33
	DST	U		ROUN	34
	DLD*	1,4	ACMQ=A	ROUN	35
	DFDH	U	ACMQ=2A/X2	ROUN	36
	SUB	=0002000000000	ACMQ=A/2X2	ROUN	37
	DFAD	U	ACMQ=X3=SQRT(A)	ROUN	38
	DST*	2,4	STORE RESULT	ROUN	39
*				ROUN	40
AA	AXT	**,1	RESTORE XR1	ROUN	41
	TOV	3,4	RETURN	ROUN	42
	TRA	3,4	RETURN	ROUN	43
*				ROUN	44
A	DLD*	1,4		ROUN	45
	TZE	AA-1	ZERO RETURN	ROUN	46
	SXA	SB,4		ROUN	47
	CALL	SYSWOT	PRINT ERROR MESSAGE	ROUN	48
	TIX	M,,4		ROUN	49
SB	AXT	**,4		ROUN	50
	TRA	AA		ROUN	51
"	BCI	4,0DSQRT ARG NOT NORMAL		ROUN	52
RR	SXA	SA,4		ROUN	53
	CALL	SYSWOT		ROUN	54
	TIX	MA,,4		ROUN	55
SA	AXT	**,4		ROUN	56
	TRA	3,4		ROUN	57
MA	BCI	4,0DSQRT ARG NEGATIVE		ROUN	58
	DEC	.9960861B2	.25*SQRT(127/128)	ROUN	59

DEC	.9882118B2		ROUND	60
DEC	.9802742B2		ROUND	61
DEC	.9722718B2		ROUND	62
DEC	.9642030B2		ROUND	63
DEC	.9560662B2		ROUND	64
DEC	.9478594B2		ROUND	65
DEC	.9395810B2		ROUND	66
DEC	.9312290B2		ROUND	67
DEC	.9228014B2		ROUND	68
DEC	.9142962B2		ROUND	69
DEC	.9057110B2		ROUND	70
DEC	.8970438B2		ROUND	71
DEC	.8882919B2		ROUND	72
DEC	.8794530B2		ROUND	73
DEC	.8705243B2		ROUND	74
DEC	.8615030B2		ROUND	75
DEC	.8523864B2		ROUND	76
DEC	.8431711B2		ROUND	77
DEC	.8338540B2		ROUND	78
DEC	.8244316B2		ROUND	79
DEC	.8149003B2		ROUND	80
DEC	.8052562B2		ROUND	81
DEC	.7954951B2		ROUND	82
DEC	.7856128B2		ROUND	83
DEC	.7756046B2		ROUND	84
DEC	.7654655B2		ROUND	85
DEC	.7551904B2		ROUND	86
DEC	.7447735B2		ROUND	87
DEC	.7342088B2		ROUND	88
DEC	.7234898B2		ROUND	89
DEC	.7126096B2		ROUND	90
DEC	.7015608B2		ROUND	91
DEC	.6903351B2		ROUND	92
DEC	.6789238B2		ROUND	93
DEC	.6673174B2		ROUND	94
DEC	.6555055B2		ROUND	95
DEC	.6434769B2		ROUND	96
DEC	.6312191B2		ROUND	97
DEC	.6187184B2		ROUND	98
DEC	.6059600B2		ROUND	99
DEC	.5929271B2		ROUND	100
DEC	.5796012B2		ROUND	101
DEC	.5659616B2		ROUND	102
DEC	.5519851B2		ROUND	103
DEC	.5376453B2	ETC	ROUND	104
DEC	.5229125B2	.25*SQRT(35/128)	ROUND	105
DEC	.5077524B2	.25*SQRT(33/128)	ROUND	106
TABLE COMMON	BSS	TEMP STORAGE	ROUND	107
END	3		ROUND	108

REFERENCES

1. S. E. Atta and J. H. Harvey. Numerical analysis of neutron resonances ORNL-3205 Oak Ridge, Tennessee, January 1962.
2. D. W. Drawbaugh and G. Gibson. A single level analysis of ^{235}U based on recent σ_t , σ_f and σ_c measurements. Conference of Nuclear Data-Microscopic Cross Sections, Paris, 1966. CN-23/44.
3. D. B. Adler and F. T. Adler. Analysis of neutron resonances in fissile elements: programs CODILLI, CURVEPLOT, and SIGMA. COO-1546-3, 1966.
4. N. Draper and H. Smith. Applied Regression Analysis, J. Wiley and Sons, 1966.
5. H. H. Rosenbrock. The Computer Journal, 3, 175-183 (1960).
6. M. J. D. Powell. The Computer Journal, 7, 155-162 (1964).
7. G. deSaussure, L. W. Weston, R. Gwin, R. W. Ingle, J. Todd, R. W. Hockenburry, R. R. Fullwood, A. Lottin. Measurement of the neutron capture and fission cross sections and their ratios for ^{233}U , ^{235}U ; and ^{239}Pu . Conference on Nuclear Data-Microscopic Cross Sections, Paris 1966.
8. D. B. Adler, F. T. Adler. Low energy level structure of U-233 and U-235 with the multi-level formalism. Technical Report #1, AEC-BNL 006977, 1964.
and D. B. Adler and F. T. Adler. Multilevel analysis of the U-235 total and fission cross sections in the energy region below 37 eV. Final report AEC-BNL 006977, 1966.
9. R. G. Fluharty, N. H. Marshall, O. D. Simpson. Analytical description of the subcadmium cross sections of U^{235} , U^{233} , Pu^{239} , and Pu^{241} . Conference on Neutron Cross Section Technology, Washington, 1966. Conf 660303, Book 2, page 985.

10. V. J. Law and R. V. Bailey. Chemical Engineering Science 18, 189-202, (1963). Pergamon Press Ltd., Oxford, Printed in Great Britain.
11. R. Fletcher. The Computer Journal, 8, 33 (1965)
12. J. Orear. Notes on Statistics for Physicists. UCRL 8417, 1958.
13. D. McCall. Auxiliary routines for use with programs CURVEPLOT, TANDEM and CODILLI. COO 1546-6 (1967).