AUG 9 1967

COO-1546-4

MASTER

LEMA: A PROGRAM FOR CALCULATING NEUTRON CROSS SECTIONS WITH THE
R-MATRIX FORMALISM AND WITH THE COMPLEX POLE EXPANSION OF
THE COLLISION MATRIX

D. B. Adler

and

F. T. Adler

Department of Physics

University of Illinois, Urbana

March 1967

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

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.

GESTI PRICES

H.C. \$ 3.00 MN .65

LEMA: A PROGRAM FOR CALCULATING NEUTRON CROSS SECTIONS WITH THE
R-MATRIX FORMALISM AND WITH THE COMPLEX POLE EXPANSION OF
THE COLLISION MATRIX

D. B. Adler

and

F. T. Adler

Department of Physics

University of Illinois, Urbana

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

1.	Introduction	1
2.	Method of calculation	
	a) Inversion of the level matrix	
	with real algebra.....	6
	b) Calculation of $\sigma(E) \sqrt{E}$	7
3.	The poles of the Collision Matrix and the	
	evaluation of the corresponding Complex Widths..	12
4.	The case of two interfering levels	15
5.	The case of more than two interfering levels....	18
6.	Preliminary results	19
7.	References.....	28
8.	Appendices	
	Appendix A. Input and Output.....	30
	Appendix B. Listing of LEMA 1.....	35
	Appendix C. Listing of LEMA 2.....	42
	Appendix D. Listing of LEMA 3.....	53
9.	Acknowledgment.....	69

ABSTRACT

The program LEMA calculates neutron cross sections in fissionable elements with the alternate approaches of the Wigner level matrix and of the generalized Kapur-Peierls expansion using the complex poles of the collision matrix. The calculation proceeds from a given set of R-matrix parameters, and the complex eigenvalues of the level matrix are hence derived. Two alternate evaluations of the cross section curves are provided, and parameters for the multilevel expansion are produced.

Two FORTRAN II versions of LEMA and a preliminary FORTRAN IV version are given as appendices.

As an illustration of the potential uses of these programs, preliminary results of calculations of interference effects in closely spaced resonances are presented.

I. Introduction

The generalized Kapur-Peierls formalism has proved useful to describe neutron cross sections 1,2 / of U-233 and U-235 in the low energy region (1-30 eV). The cross section analysis involving this formalism provides a phenomenological description of the cross sections of the fissile elements in a form useful for practical applications in reactor physics. To obtain a better insight into the physical meaning of the parameters derived in such an analysis it is useful to investigate quantitatively the connection between these parameters and those used in the classical R-matrix theory.

Therefore, it appears useful to establish numerical methods, and associated computer programs, which can lead from the R-matrix formulation to the multilevel parameters of the generalized Kapur-Peierls formalism.⁺

In this report the computer program LEMA is described which has been written for this purpose. That is, to evaluate cross sections from a given set of R-matrix parameters and to derive the complex poles and the complex partial widths for the alternate description in terms of the generalized Kapur-Peierls expansion. With this approach, parameters are obtained which lead to cross section formulae given as formally simple superpositions of symmetrical and asymmetrical Breit-Wigner type lines. As a by-product, this formulation also allows for a much faster evaluation of temperature and resolution effects than the direct level matrix approach permits, which has been given in the literature. 3,4 /

The first part of this calculation, concerning the cross section evaluation and the level matrix approach, is incorporated in the program LEMA 1, described in Section 2. Because the FORTRAN II system at the University of Illinois does not include complex algebra routines, the

⁺ Literature relevant to the theoretical aspects is referenced elsewhere. 1 /

calculation has been arranged using only real matrices. For the same reason of limitation to real algebra, the calculation related to the evaluation of the poles of the collision matrix and of the related cross section parameters was first developed limited to the case of two groups of two interfering levels. The relevant program LEMA 2 is described in Section 3.

For the case of more than two interfering levels an associated routine has been used previously which is applicable to 25 interfering levels having channel interactions weak enough to be treated by perturbation methods. This associated routine (BWSIG 1) is not reported here, since it dates back to a time (1962) when the work was initiated without support, other than free machine time granted by the Graduate College of the University of Illinois. It should be remarked, however, that BWSIG 1 has been used effectively to compare cross sections evaluated with the level matrix (LEMA) and by perturbation methods. In particular, the Pattenden and Harvey ^{5/} compilation for U-233 has provided a successful test case.

The second part of Program LEMA can be generalized to evaluate complex eigenvalues of the level matrix for any number of levels on computers utilizing FORTRAN IV (with complex algebra routines) such as IBM 360, or FORTRAN 63, such as the CDC 1604, and an IBM 7094 operating under the IBSYS system.

This generalization has been accomplished in the preliminary version of the program LEMA 3 which is discussed in Section 5.

The possible uses of the program LEMA, so generalized could encompass the study of groups of interfering levels in number larger than

two, as well as a detailed analysis of the distribution of the poles of the collision matrix, starting from given R-matrix parameter distributions.

So far, the existing programs LEMA have been used to study the role played by groups of strongly interfering levels, when closely spaced, and in these cases to realize the effects of such interferences on the asymmetries of the capture cross section. This point becomes, in fact, important, when one tries to correlate fission and capture data. The latter, being expectedly less asymmetrical, provide in general a better definition of the resonance energies; thus it has been essential to establish the existence of cases in which distortions occur as a consequence of the interferences. 2

Another matter of interest is the theoretical comparison between cross sections evaluated with the two methods (level matrix inversion, on one side, poles expansion on the other) starting from the same set of R-matrix parameters. Here, two questions have needed answers, as to the validity of the Kapur-Peierls expansion:

- (1) What is the imprecision induced by the assumption that the poles and the residues of the collision matrix are energy independent?
- (2) In using the assumption of energy independence, is the presence of a smoothly varying background made obvious by numerical disagreements between the two sets of cross section values.

To accommodate possible numerical discrepancies due to the dependence of the poles on the neutron energy first order perturbation corrections have been introduced in the program. The results are discussed in Section 6. It should be pointed out, however, that, for all cases studied numerically, the contribution of the first order perturbation is negligible, the agreement between the two evaluations being in all cases good to the 5-th significant figure without inclusion of the perturbation terms. Besides, for all these cases, no obvious "background" seems to be needed to compensate for the energy dependency of the parameters, such as the Rosenfeld-Humblert theory would indicate.

It should be recalled, however, that these numerical studies have been made for the present in the region between 1 and 30 eV, and that the Vogt assumption has been used explicitly in the definition of the level matrix.

In the study of pairs of interfering, closely spaced resonances, cases shown by Lynn ^{6/} have been investigated. Quasi-resonances produced by the use of LEMA have been further analyzed by means of available cross section fitting codes, ^{7/} to ascertain whether such resonance curves lead to double or single levels by the fitting process, and to counter-check the validity of the Kapur-Peierls parameters obtained from LEMA 2.

Three related versions of the program LEMA are presented here:

LEMA 1: originally written for the ILLIAC II, with attached graphing routines, for the evaluation of resonance cross sections with the level matrix inversion. This is accomplished with the double precision subroutine DINVL of the University of Illinois. ^{8/}

Except for the number of the fission channels involved (limited to two , although generalization to 3 is straightforward) this version could be considered equivalent to the ORNL Muffle ^{9/} code. The listing of LEMA 1 is given in Appendix B.

LEMA 2: besides the content of LEMA 1, the case of two interfering levels per spin state is treated with real arithmetic. Cross section parameters are evaluated, for further calculations with the multi-level formalism. The listing of this program is given in Appendix C.

Both LEMA 1 and LEMA 2 are in FORTRAN II, and the versions given here are operational on the IBM 7094 of the University of Illinois.

LEMA 3: is the FORTRAN IV version of LEMA 2, generalized to accommodate 25 interfering resonances per spin state. This version will presently become operable on the IBM 360 at Illinois and has been tested on the IBM 7094 of the Purdue University Computer Center. The listing of LEMA 3 is given in Appendix D.

Appendix A contains the description of input and output of the three LEMA programs.

2. Method of Calculation

a) Inversion of the level matrix with real algebra

Here, the problem is limited to s-wave neutrons, so that only two independent spin states are taken into account.

Let

$$A_{\lambda\lambda'}^{-1} = (E_{\lambda} - E) \delta_{\lambda\lambda'} - i\Gamma_{\lambda\lambda'} \quad (1)$$

represent the inverse level matrix, for a finite set of levels, having the same spin state. For the other spin state similarly another matrix of the same form is given. Neglecting the effects of the distant levels, which are not included in the finite arrays of Eq.(1) one has

$$\Gamma_{\lambda\lambda'} = \frac{1}{2} \sum_c (\Gamma_{\lambda c} \Gamma_{\lambda' c})^{1/2} \quad (2)$$

where $\Gamma_{\lambda c}$ represents the partial width for the decay of the λ -th level into the channel c .

The method of inversion used to obtain $A_{\lambda\lambda'}$ is outlined below.

Let

$$B = C - iG \quad (3)$$

be a complex matrix, and let

$$U = D + iO \quad (4)$$

be its reciprocal. The four real matrices, C , G , D , O have in general energy-dependent elements. Applying the condition that U be the reciprocal of B one can write two real matrix equations

$$OC - DG = 0 \text{ (zero matrix)} \quad (5)$$

$$DC + OG = I \text{ (unit matrix)} \quad (6)$$

In the case in consideration $C_{\lambda\lambda'} = (E_{\lambda} - E)\delta_{\lambda\lambda'}$, so that in using (5) and (6) to derive D and O it is desirable to avoid the inversion of C which leads to singularities. Thus, one can write from (5)

$$D = OCG^{-1} \quad (7)$$

which, brought into (6) gives

$$O(CG^{-1}C + G) = I. \quad (8)$$

From (8) and (7) one can solve for D, and obtain:

$$D = (CG^{-1}C + G)^{-1} CG^{-1}. \quad (9)$$

Component-wise and explicitly, (8) and (9) can be rewritten as

$$O_{\lambda\lambda'}^{-1} = G_{\lambda\lambda'} + G_{\lambda\lambda'}^{-1} (E_{\lambda} - E) (E_{\lambda'} - E) \quad (8a)$$

$$D_{\lambda\lambda'} = \sum_{\mu} O_{\lambda\mu} (E_{\mu} - E) G_{\mu\lambda'}^{-1} \quad (9a)$$

Thus, identifying D and O with the real and imaginary parts of A, one can write

$$A_{\lambda\lambda'} = \sum_{\mu} O_{\lambda\mu} [(E_{\mu} - E) G_{\mu\lambda'}^{-1} + i\delta_{\mu\lambda'}] \quad (10)$$

In this manner A is evaluated by two successive inversions of the real matrices G and O^{-1} , for any value of E.

b) Calculation of $\sigma(E) \sqrt{E}$

In terms of A the elements of the collision matrix U needed for the evaluation of the neutron cross sections can be written as

$$\begin{aligned} U_{nc}^J &= e^{-i(\omega_n + \omega_c)} (\delta_{nc} + i \sum_{\lambda\lambda'} A_{\lambda\lambda'}^J \sqrt{\Gamma_{\lambda n} \Gamma_{\lambda' c}}) \\ &= e^{-i(\omega_n + \omega_c)} [\delta_{nc} + i \sum_{\lambda\lambda'} (D_{\lambda\lambda'}^J + iO_{\lambda\lambda'}^J) \sqrt{\Gamma_{\lambda n} \Gamma_{\lambda' c}}] \end{aligned} \quad (11)$$

where the index J refers to the spin state, ω_n represents the phase shift and n refers to neutron channel and c to a generic channel c. In particular, one can introduce the notation

$$U_{nn}^J = e^{-2i\omega_n} (1 + iQ_{nn}^J) \quad (12)$$

where Q_{nn}^J is defined as

$$Q_{nn}^J = \sum_{\lambda\lambda'} \sqrt{\Gamma_{\lambda n} \Gamma_{\lambda' n}} A_{\lambda\lambda'} \quad (13)$$

or through the quantities

$$Q_{1nn}^J = \frac{2g_J}{\sqrt{E}} \operatorname{Re} Q_{nn}^J \equiv \sum_{\lambda\lambda'} \sqrt{\Gamma_{\lambda n}^o \Gamma_{\lambda' n}^o} D_{\lambda\lambda'}^J \quad (14)$$

$$Q_{2nn}^J = \frac{2g_J}{\sqrt{E}} \operatorname{Im} Q_{nn}^J \equiv \sum_{\lambda\lambda'} \sqrt{\Gamma_{\lambda n}^o \Gamma_{\lambda' n}^o} O_{\lambda\lambda'}^J \quad (15)$$

Total and scattering cross sections are evaluated from these matrix elements:

$$\sigma_{nT}(E) = \frac{2C}{\sqrt{E}} \sum_J g_J \operatorname{Re}(1 - U_{nn}^J) \quad (16)$$

$$= \frac{C}{\sqrt{E}} \left[2 \frac{(1 - \cos 2\omega_n)}{\sqrt{E}} - \sum_J (\sin 2\omega_n Q_{1nn}^J - \cos 2\omega_n Q_{2nn}^J) \right]$$

$$\begin{aligned} \sigma_{nn}(E) &= \frac{C}{\sqrt{E}} \sum_J g_J |1 - U_{nn}^J|^2 \\ &= \frac{C}{\sqrt{E}} \left\{ 2 \frac{(1 - \cos 2\omega_n)}{\sqrt{E}} - \sum_J [(1 - \cos 2\omega_n) Q_{2nn}^J + \sin 2\omega_n Q_{1nn}^J] \right\} \\ &\quad + \frac{C}{2} \sum_J [(Q_{1nn}^J)^2 + (Q_{2nn}^J)^2] \frac{1}{g_J} \end{aligned} \quad (17)$$

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

For the evaluation of the reaction cross section the complex vector V is introduced:

$$v_{\lambda n}^J = \eta_{\lambda n}^J + i \zeta_{\lambda n}^J \quad (18)$$

where

$$\eta_{\lambda n}^J = \sum_{\lambda'} \sqrt{\Gamma_{\lambda' n}^0} D_{\lambda \lambda'}^J \quad (19)$$

$$\zeta_{\lambda n}^J = \sum_{\lambda'} \sqrt{\Gamma_{\lambda' n}^0} O_{\lambda \lambda'}^J \quad (20)$$

so that for $c \neq n$

$$U_{nc}^J = i \sum_{\lambda'} \sqrt{\Gamma_{\lambda' c}} v_{\lambda n}^J \frac{\sqrt{E}}{2g_J} e^{-i(\omega_n + \omega_c)} \quad (21)$$

and

$$\begin{aligned} \sigma_n^{(x)} \sqrt{E} &= \frac{C}{\sqrt{E}} \sum_J g_J \sum_{c(x)} |U_{nc}^J|^2 \\ &= \frac{C}{2} \sum_{\lambda \lambda'} \sum_J v_{\lambda n}^J v_{\lambda' n}^{*J} \sum_{c(x)} \sqrt{\Gamma_{\lambda c} \Gamma_{\lambda' c}} \end{aligned} \quad (22)$$

Under the assumption of Vogt^{3/} that the capture channels do not contribute to the off-diagonal elements of the level matrix, i.e.

$$\sum_{c(Y)} \sqrt{\Gamma_{\lambda c} \Gamma_{\lambda' c}} = \Gamma_{\lambda Y} \delta_{\lambda \lambda'} \quad (23)$$

the expression for the capture cross section acquires the simpler form

$$\sigma_n^{(Y)} \sqrt{E} = \frac{C}{2} \sum_J \sum_{\lambda} \Gamma_{\lambda Y} [(\eta_{\lambda n}^J)^2 + (\xi_{\lambda n}^J)^2] \quad (24)$$

For the resonance scattering the corresponding expression (pure resonance contribution) is represented by the last term of eq. (17):

$$\sigma_{nn} \sqrt{E} = \frac{C}{2g_J} \sqrt{E} \sum_J [(Q_{1nn}^J)^2 + (Q_{2nn}^J)^2] \quad (25)$$

while for the fission cross section equation (22) gives:

$$\begin{aligned} \sigma_n^{(F)} \sqrt{E} = C \sum_J \left\{ \sum_{\lambda \lambda'} v_{\lambda n}^J v_{\lambda' n}^{*J} \left[G_{\lambda \lambda'}^J - \frac{1}{2} \Gamma_{\lambda Y} \delta_{\lambda \lambda'} \right] \right. \\ \left. - \frac{\sqrt{E}}{2} \sum_J \frac{1}{g_J} [(Q_{1nn}^J)^2 + (Q_{2nn}^J)^2] \right\} \end{aligned} \quad (26)$$

where

$$\sum_{c(f)} \sqrt{\Gamma_{\lambda c} \Gamma_{\lambda' c}} = 2G_{\lambda \lambda'}^J - \delta_{\lambda \lambda'} \Gamma_{\lambda Y} - \sum_{c(n)} \sqrt{\Gamma_{\lambda n} \Gamma_{\lambda' n}} \quad (27)$$

and

$$2G_{\lambda\lambda'}^J \equiv 2\Gamma_{\lambda\lambda'} = \sum_c \sqrt{\Gamma_{\lambda c} \Gamma_{\lambda' c}} \quad (\text{sum over all reaction channels}) \quad (27a)$$

Under the assumption that also

$$\sum_{c(n)} \sqrt{\Gamma_{\lambda n} \Gamma_{\lambda' n}} = \delta_{\lambda\lambda'} \Gamma_{\lambda n} \quad (28)$$

Eq. (26) reduces to

$$\sigma_n^{(F)} \sqrt{E} = C \sum_J \sum_{\lambda\lambda'} v_{\lambda n}^J v_{\lambda' n}^{*J} G_{\lambda\lambda'}^J - \frac{C}{2} \sum_J \sum_{\lambda} v_{\lambda n}^J v_{\lambda n}^{*J} (\Gamma_{\lambda n} + \Gamma_{\lambda' n}) \quad (29)$$

Eq. (29) can be reduced to the expression

$$\sigma_n^{(F)} \sqrt{E} = C \sum_J Q_{2nn}^J - \sigma_n^{(\gamma)} \sqrt{E} - \sigma_{nn} \sqrt{E} \quad (29a)$$

by combining equations (24) and (25), and making use of the equalities:

$$\begin{aligned} \sum_{\lambda\lambda'} v_{\lambda n}^J v_{\lambda' n}^{*J} G_{\lambda\lambda'}^J &\equiv \frac{-1}{2i} \left\{ \sum_{\lambda'} v_{\lambda' n}^{*J} \sum_{\lambda} v_{\lambda n}^J A_{\lambda\lambda'}^{-1} - \sum_{\lambda} v_{\lambda n}^J \sum_{\lambda'} v_{\lambda' n}^{*J} A_{\lambda\lambda'}^{-1*} \right\} \\ &= \frac{-1}{2i} \left\{ \sum_{\lambda'} v_{\lambda' n}^{*J} \sum_{\lambda, \lambda''} \sqrt{\Gamma_{\lambda'' n}^0} A_{\lambda\lambda''} A_{\lambda\lambda'}^{-1} - \sum_{\lambda} v_{\lambda n}^J \sum_{\lambda', \lambda''} \sqrt{\Gamma_{\lambda'' n}^0} A_{\lambda' \lambda''}^* A_{\lambda\lambda'}^{-1*} \right\} \\ &= \frac{-1}{2i} \sum_{\lambda'} (v_{\lambda' n}^{*J} \sqrt{\Gamma_{\lambda' n}^0} - v_{\lambda' n}^J \sqrt{\Gamma_{\lambda' n}^0}) \\ &\equiv \sum_{\lambda', \lambda''} \sqrt{\Gamma_{\lambda' n}^0} \sqrt{\Gamma_{\lambda'' n}^0} O_{\lambda' \lambda''}^J \\ &\equiv Q_{2nn}^J \end{aligned} \quad (30)$$

3. The poles of the Collision Matrix and the evaluation of the corresponding Complex Widths.

The equations of Section 2 are not sufficient to satisfy realistic needs, such as the evaluation of temperature dependent cross sections, and, for comparison with experimental data, resolution effects. Although the broadening effects can be evaluated by direct integration, this procedure is time consuming and not recommendable for repetitive calculations.

This practical problem can be circumvented, if instead of using the level matrix formulation one turns to the cross section expansion in terms of the complex poles of the collision matrix. These are provided by the equation

$$\det (A^{-1} (E)) = 0 \quad (31)$$

which, in general, can be solved by complex algebra. A complex, orthogonal transformation S , diagonalizing A^{-1} , is thus derived from the equations:

$$\sum_{\lambda'} S_{\lambda'k} A_{\lambda\lambda'}^{-1} (E \equiv C_k) = 0 \quad (32)$$

and from the normalization condition

$$\sum_k S_{\lambda k} S_{\lambda'k} = \delta_{\lambda\lambda'} \quad (33)$$

From S are evaluated the complex partial widths, $\tilde{\Gamma}_{kc}^{1/2}$

$$\tilde{\Gamma}_{kc}^{1/2} = \sum_{\lambda} S_{\lambda k} \sqrt{\Gamma_{\lambda c}} \quad (34)$$

which provide the basis for the calculation of the relevant cross section parameters.

For fission or capture cross sections, the auxiliary complex matrix

$$(\tilde{\Gamma}_{jn}^o \tilde{\Gamma}_{kn}^{o*})^{1/2} \sum_{c(x)} (\tilde{\Gamma}_{jc} \tilde{\Gamma}_{kc}^*)^{1/2} = \phi_{jk}^{(x)} + i\sigma_{jk}^{(x)} \quad (35)$$

is formed where (x) stands for either fission or capture, and where the $\sum_{c(x)}$ refers to all the channels intervening into the reaction (x) . The imaginary part, σ_{jk} is zero only if the j -th and the k -th levels are not interfering. Therefore the matrix (35) is in general complex, also for the case of the capture cross section, where the assumption contained in the condition (23) only partly simplifies the evaluation of (35).

The parameters describing a given reaction cross section (x) can be calculated using the complex matrix given in eq. (35) according to the relations:

$$G_j^{(x)} = \sum_k \frac{(v_j + v_k) \phi_{jk}^{(x)} + (\mu_j - \mu_k) \sigma_{jk}^{(x)}}{(\mu_j - \mu_k)^2 + (v_j + v_k)^2} \quad (36)$$

$$H_j^{(x)} = \sum_k \frac{(v_j + v_k) \sigma_{jk}^{(x)} - (\mu_j - \mu_k) \phi_{jk}^{(x)}}{(\mu_j - \mu_k)^2 + (v_j + v_k)^2} \quad (37)$$

For the total cross section, the calculated parameters α^o, β^o , are defined as

$$\alpha_j^o + i\beta_j^o = \tilde{\Gamma}_{jn}^o \quad (38)$$

Properties of these parameters, derived from the formal properties of the collision matrix $\frac{1}{S}$ and of the orthogonal transformation, S , are summarized here:

$$\sum_{(x)} \overline{G_j^{(x)}} = \alpha_j^o \quad \sum_{(x)} \overline{H_j^{(x)}} = \beta_j^o \quad (39)$$

$$\sum_j \overline{\alpha_j^o} = \sum_{\lambda} \overline{\Gamma_{\lambda n}^o} \quad \sum_j \overline{\beta_j} = 0 \quad (40)$$

$$\sum_j \overline{\tilde{\Gamma}_{jc}} = \sum_{\lambda} \overline{\Gamma_{\lambda c}} \quad (41)$$

With this approach, the cross section formulae equivalent to those of the previous section, which are given as superpositions of interfering Breit-Wigner lines,

$$\sqrt{E} \sigma_n^{(T)} = C \sum_{J,k} \frac{[v_k (\alpha_k^{oJ} \cos 2\omega_n - \beta_k^{oJ} \sin 2\omega_n) + (\mu_k - E) (\beta_k^{oJ} \cos 2\omega_n + \alpha_k^{oJ} \sin 2\omega_n)]}{[(\mu_k - E)^2 + v_k^2]} \quad (42)$$

$$+ \frac{C}{\sqrt{E}} 2 (1 - \cos 2\omega_n)$$

$$\sqrt{E} \sigma_n^{(x)} = C \sum_{J,k} \frac{[G_k^{(x)J} v_k + H_k^{(x)J} (\mu_k - E)]}{[(\mu_k - E)^2 + v_k^2]} \quad (43)$$

where, as in equations (16) and (22), J stands for the spin state index, C is the conversion constant given in the previous sections, and $\mu_k - i v_k$ represents the k -th pole of the collision matrix for the J -th state.

The evaluation of the eigenvalues of the level matrix, from equation (31), requires in general, access to complex arithmetic routines, not available at present, at Illinois. In the absence of complex arithmetic, the case of two interfering levels has been coded in an independent version of LEMA (LEMA 2). The method of inversion of the level matrix is the same described in Section 2, while the actual matrix inversion and the calculation of the complex eigenvalues have been directly incorporated in the program, so that it can be used either on ILLIAC II or on the IBM 7094. (+)

4. The case of two interfering levels

Indicating with E_1, E_2 , the resonance levels, Γ_{11}, Γ_{22} , the corresponding half-widths, and with Γ_{12} the interference term, the eigenvalues of A^{-1} , $\mu_i - i\nu_i$, are expressed by

$$\mu_1 - i\nu_1 = E_0 - i\Gamma_0 - (\eta + i\gamma) \quad (44)$$

$$\mu_2 - i\nu_2 = E_0 - i\Gamma_0 + (\eta + i\gamma)$$

$$\text{where } E_0 = (E_1 + E_2)/2, \quad \Gamma_0 = (\Gamma_{11} + \Gamma_{22})/2 \quad (45)$$

The shift $\eta + i\gamma$ is given by

$$\eta^2 = \frac{1}{2} \left[d^2 - a^2 - \Gamma_{12}^2 + \sqrt{(d^2 + a^2)^2 + \Gamma_{12}^4 + 2\Gamma_{12}^2 (a^2 - d^2)} \right] \quad (46)$$

$$\gamma = -ad/\eta \quad (47)$$

$$\text{where } d = (E_2 - E_1)/2, \quad a = (\Gamma_{22} - \Gamma_{11})/2 \quad (48)$$

(+) The machine language of Illiac II, NICAP, is in fact not compatible with the SCATRE of the 7094, so that library subroutines are not interchangeable.

The evaluation of the transformation S , and of the complex partial widths follows immediately, using the equations (32) to (41).

In the special case in which the widths are identical,

$$a = 0,$$

and

$$d = \Gamma_{12}$$

one has the result

$$\eta = \gamma = 0 \quad (49)$$

This is a degenerate case, in which the calculation of the cross sections with the poles expansion seems to fail. In this case, in fact, it is not possible to define the transformation S , and the poles coalesce into one double pole. A further consideration on the meaning of this degeneracy, however, indicates that its existence is more formal than physical. Bearing in mind the fact that Γ_{ii} contain an energy dependent part, Γ_{ni} , it appears that the identity

$$\Gamma_{11} = \Gamma_{22}$$

in this calculation occurs only for the Γ_{ni} evaluated at E_i , which are the values conventionally adopted to form A^{-1} , before diagonalizing.

One can, in these cases, replace the initial values of Γ_{ni} by the values calculated at the intermediate energy E_0 . This step suffices to remove the degeneracy, and to provide values of η and γ , which allow to proceed in the calculation.

In these cases the resulting parameters for the two (very close) levels are such that the resulting line is practically symmetrical: the

"even" parameters, such as $\alpha^0, G^{(x)}$, are practically identical, the "odd" parameters, such as $\beta^0, H^{(x)}$, are almost equal and opposite. As a consequence, the resonance line presents a structured maximum which has a dip corresponding to the intermediate energy, E_0 .

Two related quasi-degenerate cases arise for $a = 0$, but $d \neq \Gamma_{12}$. The case $d > \Gamma_{12}$ leads to a separation in the resonance energies, with identical values of ν , $\nu_1 = \nu_2$. For $d < \Gamma_{12}$ the widths ν_1, ν_2 are different but the resonance energies coincide ($\mu_1 = \mu_2$). Also, for these cases the resulting lines are symmetrical, but the maximum appears deformed, by the appearance of a "dip" due to the interplay of the interference terms. In the section 6 is given a more detailed discussion of these results.

The importance of the energy dependence of the parameters has also been considered. This version of LEMA, (LEMA-2) contains an optional branch, which evaluates, for each value of E , the first order perturbation corrections for the elements of the matrix S and for the multilevel parameters.

At each value of E the corrective terms are given by $\frac{10}{}$

$$S_{\lambda k} = S_{\lambda k}^0 - \frac{i}{2} \sum_{k' \neq k} \frac{S_{\lambda k'}^0 \sum_{\lambda'} S_{\lambda' k}^0 S_{\lambda' k'}^0 \Gamma_{\lambda' n}^0}{c_k - c_{k'}} \left(\frac{\sqrt{E}}{2g_J} - \frac{\sqrt{E_{\lambda'}}}{2g_J} \right). \quad (50)$$

It is seen that the effects of these corrections affect only the 4th or 5th significant digit, whereas the agreement between the cross section values obtained by the poles expansion, with the values derived from the level matrix inversion is of the same order, or better, without inclusion of perturbation corrections.

5. The case of more than two interfering levels

The case of more than two interfering levels can be treated readily only if complex arithmetic routines can be incorporated into the computer program.

The imminent installation of an IBM 360 system on the campus of the University of Illinois has stimulated the preparation of a preliminary version of LEMA (LEMA 3) in FORTRAN IV. This version of LEMA can handle 25 interfering resonances per spin state. At present the corresponding eigenvalues are obtained by using the FORTRAN IV subroutine ALLMAT of Funderlic and Rizel of ORGDP^{11/}. For the real matrix inversion, the subroutine INV1^{12/} (subroutine MATINV, revised at the Department of Computer Science, University of Illinois) has been adapted to FORTRAN IV from FORTRAN II.*

LEMA 3 should not be considered to be in its definitive form. Less than 30 minutes of computer time has been available up to the present to perform the necessary compilations and code checks. Even more important, the use of routine ALLMAT should be restricted to the case of matrices with well separated eigenvalues since otherwise there is no assurance of convergence and of correct calculations of eigenvalues and eigenvectors.

Automatized studies of the statistical distribution of the poles of the collision matrix are therefore not yet possible with this program in its present form.

Users should also be aware that further code checking will be necessary to achieve reliable results within the indicated limitations.

*The double precision subroutine DNV1, written in SCATRE, could not easily be adjusted to FORTRAN IV.

6. Preliminary results.

From the cases studied numerically with LEMA 2 in the low energy region it appears that the direct cross section evaluation via the level matrix, and the expansion in the form of interfering Breit-Wigner lines give equivalent results without the need of introducing corrections due to the energy dependence of the Kapur-Peierls poles and complex parameters. Also, it does not appear necessary to introduce compensating smoothly varying terms, which may accompany the expansion in complex poles in the Rosenfeld-Humblet theory.

In all the cases considered, the two alternate expansions agree to the order of five significant digits, that is to the accuracy expectable from single precision calculations with FORTRAN II.

As preliminary results, cases of closely spaced levels are presented here, with strongly interfering fission channels, and also with no interference. The partial widths have been chosen identical, for the two levels, and such that, by varying the spacing, the degenerate situations of Section 4 are reproduced. The R-matrix parameters used as input are listed in Table I. The cases 1 to 3 correspond to a spacing larger than the interference term $2|\Gamma_{12}|$, the cases 6 to 8 to a spacing less than $2|\Gamma_{12}|$, and the cases 4 and 5 reproduce the actual degenerate situation of the two coalescing eigenvalues. The multilevel parameters obtained from LEMA are listed in Table II and Table III. The values of the "odd" parameters are listed only for the level 1, since for the level 2 the corresponding parameters are equal and opposite.

TABLE I

R-matrix parameters of cases investigated (in units of eV)

Case	E_2	Γ_{12}
1	8.78	.03
2	8.78	0
3	8.78	- .03
4	8.76	.03
5	8.76	0
6	8.76	- .03
7	8.74	.03
8	8.74	0
9	8.74	- .03

Parameters common to all nine cases

$$E_1 = 8.70$$

$$\Gamma_{11} = \Gamma_{22} = .051$$

$$\Gamma_Y = .04$$

$$\Gamma_F = .06$$

$$\Gamma_n = .02$$

TABLE II

Eigenvalues of the level matrix (in units of eV)

Case	μ_1	μ_2	ν_1	ν_2
1	8.71354	8.76646	.0510	.0510
2	8.70	8.78	.0510	.0510
3	8.71354	8.76646	.0510	.0510
4	8.72977	8.73023	.05123	.05077
5	8.70	8.76	.0510	.0510
6	8.72977	8.73023	.05077	.05123
7	8.72	8.72	.07336	.02864
8	8.72	8.72	.051	.051
9	8.72	8.72	.07336	.02864

TABLE III
Multilevel parameters obtained from LEMA (in units of $\text{eV}^{1/2}$)

Case	α_1	α_2	G_1^F	G_2^F	G_1^Y	G_2^Y	β_1	H_1^F	H_1^Y
1	.67856-03	.67418-03	.45249-03	.44974-03	.21559-03	.21375-03	.76710-03	.66462-03	.97600-04
2	.678067-03	.67497-03	.39886-03	.39704-03	.26591-03	.26469-03	0	0	0
3	.678861-03	.674180-03	.194266-03	.191512-03	.46152-03	.459684-03	-.767101-03	-.371815-03	-.376464-03
4	.45421-01	-.44067-01	.37603-01	-.36672-01	.74434-02	-.704122-02	.44589-01	.370513-01	.71847-02
5	.67807-03	.67574-03	.39886-03	.39749-03	.26591-03	.26499-03	0	0	0
6	.45267-01	-.43913-01	.15548-01	-.15273-01	.28305-01	-.27278-01	-.44742-01	-.15497-01	-.27846-01
7	.158596-02	-.231387-03	.119997-02	-.235914-03	.367607-03	.431168-05	-.699917-06	-.411688-06	-.274462-06
8	.678063-03	.676502-03	.398863-03	.397942-03	.265909-03	.265295-03	0	0	0
9	-.231387-03	.158596-02	-.971473-04	.248848-03	-.127847-03	.127344-02	-.699893-06	-.411699-06	-.274470-06

The calculations presented here have been done taking the potential scattering contribution as of the order of 10 barns, and the atomic mass number as 235. In all of the cases studied, the resonance scattering contribution appears large, somewhat larger than 100 barns.

It is also interesting to observe the relative locations of the maxima of the calculated cross sections. The closeness of the levels, in combination with the values of the interference term Γ_{12} , in several cases leads to a "dip" between the two closely spaced maxima, while in other instances only one single maximum appears.

The detailed behavior of the total, fission, and capture cross sections depends on the actual spacing and on the magnitude and sign of Γ_{12} . The scattering cross section shows maxima which are very large compared to the maxima for isolated single levels, and which are shifted from the maxima of the other cross sections. This shift, which appears to be less than the half-width (as it would be the case for isolated single levels) depends again on the spacing and on the value of Γ_{12} .

For the practical analysis of experimental data, considerations on the location of the maxima are, of course, only of academic interest since the experimental imprecisions may well be so large that the relative locations of the maxima cannot be determined with sufficient accuracy to make the required distinction.

With regard to the evaluation of experimental data it is, of course, of interest to establish whether the "quasi-resonances" could be spotted, at least in some cases. To investigate this point, the nine cases presented here were used as follows: temperature and resolution broadening were introduced, corresponding to the U-235 fission measurements of Michaudon^{13/} and the resulting broadened cross sections were calculated from the parameters given in Tables II and III.

Figures 1A to 9C show the total (A), fission (B) and capture (C) cross sections for the cases listed in Table I. It is interesting to note that for $\Gamma_{12} > 0$ and $T = 0$, a dip appears between the maxima in all cases. The dip is, of course, reduced in Fig.7, where the level spacing d is less than Γ_{12} as compared to Fig.1 where $d > \Gamma_{12}$. This result can readily be understood because in the cases 1,4 $\beta_1 > 0$ and $\beta_2 < 0$ so that the asymmetric contributions are positive for $E < \mu_1$ and $E > \mu_2$, while they are negative for $\mu_1 \leq E \leq \mu_2$, so that the dip is formed by "negative interference." For the limiting case 7 ($d < \Gamma_{12}$) the values of β_1 and β_2 are so small that their effect is negligible. Here, the role of "negative interference" is instead to be attributed to the negative value of the parameter α_2 for the narrower of the two(coalescing) levels !

By reversing the sign of Γ_{12} the eigenvalues are unchanged, while the elements of the orthogonal transformation matrix S and the resulting parameters change value. The case 3, as counterpart of 1, shows β values of signs opposite to the case 1. Hence the interference is constructive, and results in a positive superposition of effects, between μ_1 and μ_2 . For case 6 a similar consideration can be repeated, with the additional comment that here the

marked central maximum is in effect the result of two very large α values of opposite signs, enhanced by equally large β values. The case 9 is instead represented by values of α which have reversed their role with respect to case 7, thus the narrower level has the highest positive α and the effect of the negative α for the wider level shows in the sharp depression of the wings.

The cases of non-interfering resonances ($\Gamma_{12} = 0$), such as 5 and 8 show the coalescence of the maxima for total fission and capture cross section, as expectable from the overlap of two single levels, while the resonance scattering reaches its maximum at an energy slightly higher than the location of the maximum common to σ_T , σ_Y , and σ_F . In case 2 a small dip appears, because the spacing is just sufficiently large that the overlap is not complete. The maxima for σ_T , σ_Y , and σ_F coincide however again.

Similar aspects of the behavior of quasi-resonances have also been investigated by Lynn^{6/} and parametric studies have been reported by Garrison.^{14/} Results such as those presented above or discussed by other authors cannot be accepted at face value unless they are brought closer to the reality of the experimental measurements by including the effects of Doppler and resolution broadening. Such calculations have also been performed for the cases discussed above.

If Doppler and resolution broadening are included, the dips observed for $T = 0$ are eliminated. One observes, however, in the cases 1 and 4 an unusual structure, with a "mesa" top, which is recognizably different from the familiar single level line shape. In case 7 the flattening of the single level top also appears, but it is much less pronounced. For the cases 3, 6, and 9 ($\Gamma_{12} < 0$) the single sharp maximum appears to suggest one single Breit-Wigner line, as pointed out by Lynn^{6/}.

Returning now to the question whether quasi-resonances can be resolved, starting from experimental data, the previous results seem to indicate the necessity of analyzing in a coordinated manner three out of four cross sections.

In particular, the comparison of the "absorption" cross section, as sum of capture and fission, with the total cross section can indicate whether the scattering cross section is anomalously large. In this case no single level can be used to explain the difference, while the assumption of two very close levels can easily explain large values of the resonance scattering, as seen in these examples. In Figures 1D to 9D are plotted total and scattering cross sections for the cases investigated here.

Moreover, the appearance or nonappearance of a "mesa" top could qualitatively reveal the sign of Γ_{12} , provided no more than two levels are to be considered. The suggestion that for some of the U-235 levels pairs of overlapping single levels are needed has already been made by Michaudon.^{13/} In practice, one example can be recalled here (the 8.78 level of U-235) for which the analysis of capture and fission data of DeSaussure^{15/} provides an absorption peak of about 100 barns less high than the appropriately broadened total cross section of Michaudon.* It appears worthwhile to pursue detailed analysis of this and other U-235 levels along these lines.

Finally, the analysis of the time-dependence of the decaying state might in these cases support the investigation. The degenerate case should in fact decay with an exponential law corrected with a linear term for the case of a double pole of the collision matrix.^{16,17/}

* We are indebted to Dr. DeSaussure for this comment.

The preliminary results presented here may indicate the usefulness of program LEMA, and of associated programs^{7/} for the realistic study of quasi-resonances, including the effects of Doppler and resolution broadening. Finally, LEMA 3 may provide in the future a useful tool to study the distributions of the poles and residues of the collision matrix and for work on synthetic cross sections.

References

1. F. T. Adler and D. B. Adler
"Interpretation of neutron cross sections of the fissionable materials for the resolved resonance region." Conference on Neutron Cross Section Technology, 1966, Washington, D.C. CONF-660303, Book 2, pp.873-893.
2. 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, Subcontract AEC-BNL-006977, 1966 (to appear as a BNL Report).
3. E. Vogt. Phys. Rev. 118, 724 (1960).
4. E. J. Brooks, Jr., E. L. Slaggie. "Mule," a program for multilevel-multichannel calculation of neutron cross sections with Doppler and Resolution broadening. Knolls Atomic Power Lab. Nov. 1964, KAPL-M-6424.
5. N. J. Pattenden and J. A. Harvey. Proceedings International Conference on Nuclear Structure, Kingston, Canada, Univ. of Toronto Press, Toronto 1960, pp.882.
6. J. E. Lynn. Proceedings International Conference on the Study of Nuclear Structure with Neutrons, Antwerp 1965, North Holland Pub. Co., pp.125-155.
7. D. B. Adler and F. T. Adler. "Analysis of neutron resonances in fissile elements: Programs CODILLI, CURVEPLOT, and SIGMA - C00 1546-3, 1966.
8. DNV1: Double Precision Matrix Inversion, etc. F4-UOI-DNV1-61-SR University of Illinois, Digital Computer Laboratory, 1964.
9. C. A. Preskitt. MUFFLE: a code for the multilevel analysis of neutron cross sections. ORNL-TM-1180, 1965.
10. F. T. Adler and D. B. Adler. Proceedings International Conference on the Study of Nuclear Structure with Neutrons, Antwerp 1965, North Holland Publishing Co., p. 522.
11. R. E. Funderlic and J. Rinzel: Eugenvalues and Eigenvectors of complex matrices - or - AMAT ALLMAT: A Fortran IV arbitrary matrix Eugensystem Solver, Union Carbide Corp., ORGDB, Oak Ridge, Tenn. 1966.
12. MATINV: Matrix inversion with accompanying solution of linear equations. F4-UOI-MATINV-51-FR. University of Illinois, Department of Computer Science, 1966.

13. A. Michaudon, CEA-2552, 1964.
14. J. D. Garrison, "Two level interference effects in neutron cross sections," in Reactor Physics, Vol. II, Resonance Absorption, pp. 3-35, The MIT Press, 1966.
15. G. DeSaussure, L. W. Eston, R. Gwin, J. E. Russell, and R. W. Hockenbury, Nuclear Sci. Eng. 23, 45-57, 1965.
16. R. G. Newton, "Scattering theory of waves and particles," pp.607-608-International Series in pure and applied physics, McGraw-Hill, Inc., 1966.
17. A. Baracca and A. Minguzzi, Nuovo Cimento, X,39, 687, 1965.

Appendix A - Input and Output

List of symbols used in the input of Routine LEMA

A-1. Indices controlling input and output operations

- ISW: This index causes the read order to search for two possible input format for the fission channels interference. With ISW = 1 LEMA reads the angle, θ_λ of the λ -th level.
- With ISW = 2, LEMA expects to read as input the actual matrix $\cos(\theta_\lambda - \theta_{\lambda'})$. These two choices have been made necessary to accommodate the parameters obtained by different authors within Vogt's method.
- IPUNCH: decides if the calculated cross sections are to be punched on cards (IPUNCH = 1) or only printed (IPUNCH = 0)
- IPER: (only for LEMA 2). When =1, the code evaluates the contributions of first order perturbation terms, when =0, such calculations are skipped.
- J: this index is used to distinguish spin states. In particular, J = 1, 2 correspond to the two possible spin states, for the multilevel-multichannel case. The values J = 3, 4 characterize the spin states corresponding to J = 1, 2 respectively, for that set of additional single levels, which is desirable to include.
- NO(J): the number of resonance levels belonging to the two spin states. Precisely, NO(1) refers to the interfering set, NO(3) to the single level set for spin state 1, and similarly NO(2) and NO(4) for the spin state 2.

A-2. Floating point constants

ATM = atomic mass number of the nucleus

WN = nuclear radius in units of $eV^{-1/2}$. Such that, when E is in eV $\sqrt{E} WN = ak$ where a is the nuclear radius and k the neutron momentum.

CC = 6.52×10^5 barns x eV.

EMIN = lowest energy point in eV.

EMAX = highest energy point in eV.

DELE = energy interval, dE, in eV.

A-3. Resonance parameters

For a given spin state $J = 1, 2, 3, 4$.

ER(K,J) = resonance energy of the k-th level

GON(K,J) = Γ_{kn}^0 reduced neutron width for the k-th level

GG(K,J) = Γ_{ky} capture width for the k-th level

GF(K,J) = Γ_{kf} fission width for the k-th level

GN(K,J) = Γ_{kn} , neutron width for the k-th level

For a given spin state $J = 1, 2$,

TETA(K,J) = θ_K angle of the K-th fission width vector with a reference direction given for ISW = 1

GO(K,L,J) = $\cos(\theta_K - \theta_L)$ given for ISW = 2

GFACT(J) = statistical factor g, for the spin state characterized by J.

A-4. Input format for LEMA

Card 1: ISW, IPUNCH, IPER* FORMAT (2I5)
 2: ATM, WN, CC FORMAT (3E 15.4)
 3: EMIN, EMAX, DELE FORMAT (3E 15.4)

These cards are followed by four sets of parameter cards corresponding to the index J.

J=1 Wigner Eisenbud parameters of levels of the 1st spin state
 J=2 Wigner Eisenbud parameters of levels of the 2nd spin state
 J=3 Single level parameters of levels of the 1st spin state
 J=4 Single level parameters of levels of the 2nd spin state

Parameters set (1 set for each value of J from 1 to 4)

Card 1: GFACT(J), NO(J) FORMAT (E 15.4, I5)

Note: GFACT(1) = GFACT(3)

GFACT(2) = GFACT(4)

If NO(J) = 0 this is the only card in the parameters set.

Parameters cards {K = 1, NO(J)}

ER(K,J), GON(K,J), GG(K,J), GF(K,J) FORMAT (4E 15.4)

GN(K,J), TETA (K,J)

Note: for ISW = 2 and J = 3,4, TETA (K,J) is not input.

for ISW = 2 with J = 1,2 the previous 2 x NO(J) cards are followed by the matrix {K = 1, NO(J), L=1, NO(J)}.

GO(K,L,J) FORMAT (4E 15.4)

4 to a card.

* only LEMA 2

P

So that for this matrix for NO(J)= 2, 3, 4 there are NO(J) cards
for NO(J) =5, 6, 7 there are 2 x NO(J) cards, etc.

1. LEMA .1

A. The input is listed

Punched output (if IPUNCH $\neq$ 0)

C. One card per line of printed table

E, σ_{TOT} , σ_{FIS} , σ_{CAP} , σ_{SC} FORMAT (5E 15.4)

2. LEMA 2 and LEMA 3

A. The input is listed

B. The Kapur-Peierls eigenvalues and corresponding parameters are listed in the order

$$\mu, \nu \text{ (eigenvalues of } A^{-1})$$

$$\tilde{\Gamma}_n^{1/2}, \tilde{\Gamma}_Y^{1/2}, \tilde{\Gamma}_F^{1/2} \quad \text{(real and imaginary parts)} \\ \text{for each level.}$$

Finally, the multilevel cross section parameters are listed for each level as follows:

α, β	(total)
G^F, H^F	(fission)
G^Y, H^Y	(capture)

- C. Tabulated E-dependent values of the cross sections, in double columns for each E, where the column marked LMA gives the values obtained from the inversion of the Level matrix (as LEMA ONE) and the column marked P.E. gives the values as obtained from the generalized Kapur-Peierls expansion.

Punched output

- D. as for LEMA 1 (IPUNCH $\neq$ 0)

- E. cards of parameters useful for use in other associated programs

μ , ν , GR, GI for each cross section FORMAT (4E 15.4)

Comments for LEMA input

The option provided by the index IPER has not been included, thus the first card of LEMA 3 is the same as the first card of LEMA 2.

The format of the floating point input constants has been changed, from E 15.4 to E 12.3. Hence, the sequence of the input cards is modified as follows:

Card 1:	ISW, IPUNCH	FORMAT (2I5)
2:	ATM, WN, CC, EMIN, EMAX, DELE	FORMAT (6E 12.3)
3:	GFACT(J), NO(J)	FORMAT(E 12.3,I12)
4:	ER(K,J) GON(K,J), GG(K,J) GF(K,J),GN(K,J), TETA(K,J)	FORMAT (6E 12.3)

where for ISW = 2 TETA (K,J) is not input and additional cards containing GO(K,J) FORMAT (5E 12.3)

(5 to a card) are added in a manner analogous to the one used for LEMA 1.

C PROGRAM LEMA-----JULY 13, 1965 (ILLIAC II VERSION)....February 1966	LMA1	1
LEMA 1	LMA1	2
C THIS VERSION USEFUL TO CALCULATE CROSS SECTIONS BY INVERSION OF THE	LMA1	3
C LEVEL MATRIX FOR AT MOST 25 INTERFERING LEVELS FOR EACH SPIN STATE	LMA1	4
C	LMA1	5
C THIS VERSION HAS A NEATER OUTPUT THAN THE ORIGINAL L E M A ONE,	LMA1	6
C AND CONTAINS DOUBLE PRECISION MATRIX INVERSION FOR USE ON THE 7094-	LMA1	7
C	LMA1	8
DIMENSION GO(25,25,2),G(25,25),Q(25,25),OINV(25,25),D(25,25),VRE(25,25),VIM(25,4),TETA(25,4),ER(25,4),GON(25,4),GG(25,4),GF(25,4),GN(25,4),Q1(4),Q2(4),NO(4),GFACT(4)	LMA1	9
DIMENSION A(50,25),R(50),PIVOT(50),DET(2)	LMA1	10
COMMON A,R,PIVOT,DET,NR	LMA1	11
1000 READ 102,ISW,IPUNCH	LMA1	12
READ 100,ATM,WN,CC,EMIN,EMAX,DELE	LMA1	13
WOT6,2000	LMA1	14
WOT6,2010,ATM,WN,CC,EMIN,EMAX,DELE	LMA1	15
LINES = 11	LMA1	16
DO 1 J=1,4	LMA1	17
READ 103,GFACT(J),NO(J)	LMA1	18
WOT6,2020,J,J,GFACT(J),J,NO(J)	LMA1	19
LINES = LINES + 4	LMA1	20
NR=NO(J)	LMA1	21
IF(NR) 1,1,50	LMA1	22
50 GO TO(51,51,52,52),J	LMA1	23
51 GO TO(53,54),ISW	LMA1	24
52 IF (56 - LINES - NR) 372,373,373	LMA1	25
372 WOT 6, 2013	LMA1	26
LINES = 5	LMA1	27
373 WOT 6, 2030	LMA1	28
LINES = LINES + 4	LMA1	29
DO 35 K=1,NR	LMA1	30
DO 35 L=1,NR	LMA1	31
GO(K,L,J)=0.0	LMA1	32
TETA(K,J)=0.0	LMA1	33
READ 101,ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J)	LMA1	34
LINES = LINES + 1	LMA1	35
35 WOT6,2011,K,ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J)	LMA1	36
GO TO 1	LMA1	37
53 IF (56 - LINES - NR) 374,375,375	LMA1	38
374 WOT 6, 2013	LMA1	39
LINES = 5	LMA1	40
375 WOT 6, 208	LMA1	41
LINES = LINES + 4	LMA1	42
DO 42 K=1,NR	LMA1	43
READ 101, ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J),TETA(K,J)	LMA1	44
LINES = LINES + 1	LMA1	45
42 WOT6,2012,K,ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J),TETA(K,J)	LMA1	46
GO TO 1	LMA1	47
54 IF (56 - LINES - NR) 376,377,377	LMA1	48
376 WOT 6, 2013	LMA1	49
LINES = 5	LMA1	50
377 WOT 6, 209	LMA1	51
LINES = LINES + 4	LMA1	52
DO 36 K=1,NR	LMA1	53
READ 101, ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J)	LMA1	54
LINES = LINES + 1	LMA1	55
36 WOT6,2011,K,ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J)	LMA1	56
DO 37 K = 1,NR	LMA1	57
	LMA1	58
	LMA1	59
	LMA1	60

37	READ 101, (GO(K,L,J), L = 1,NR)	LMA1	61
	T1 = 5HGO(K,	LMA1	62
	T2 = 3H,J)	LMA1	63
	NROWS = 1 + (NR-1)/5	LMA1	64
	DO 1 NROW = 1,NROWS	LMA1	65
	NCOLS = 5 + NR - 5*NROW	LMA1	66
	IF (NCOLS-5) 301,301,302	LMA1	67
302	NCOLS = 5	LMA1	68
301	K1 = 5*(NROW-1) + 1	LMA1	69
	K2 = 5*(NROW-1) + NCOLS	LMA1	70
	GO TO (303,304,304,304,304), NROW	LMA1	71
303	IF (56 - LINES - NR) 390,391,391	LMA1	72
390	WOT 6, 2013	LMA1	73
	LINES = 5	LMA1	74
391	WOT 6, 350, (T1,I,T2, I = K1,K2)	LMA1	75
	WOT 6, 601	LMA1	76
	LINES = LINES + 4	LMA1	77
	GO TO 305	LMA1	78
304	IF (57 - LINES - NR) 392,393,393	LMA1	79
392	WOT 6, 2013	LMA1	80
	WOT 6, 350, (T1,I,T2, I = K1,K2)	LMA1	81
	WOT 6, 601	LMA1	82
	LINES = 9	LMA1	83
	GO TO 305	LMA1	84
393	WOT 6, 351, (T1,I,T2, I = K1,K2)	LMA1	85
	WOT 6, 601	LMA1	86
	LINES = LINES + 3	LMA1	87
305	DO 306 K = 1,NR	LMA1	88
	LINES = LINES + 1	LMA1	89
306	WOT 6, 204, K, (GO(K,L,J), L = K1,K2)	LMA1	90
	1 CONTINUE	LMA1	91
	J=1	LMA1	92
62	NR=NO(J)	LMA1	93
	IF(NR) 61,61,2	LMA1	94
	2 CONTINUE	LMA1	95
	DO 4 L =1,NR	LMA1	96
	DO 4 LL=1,NR	LMA1	97
	Y=GF(L,J)*GF(LL,J)	LMA1	98
	YY=SQRT(Y)	LMA1	99
	GO TO (39,40), ISW	LMA1	100
39	COSINE=COS(TETA(L,J)-TETA(LL,J))	LMA1	101
	GO(L,LL,J)=YY*COSINE/2.0	LMA1	102
	GO TO 41	LMA1	103
40	GO(L,LL,J)=YY*GO(L,LL,J)/2.0	LMA1	104
41	IF(L-LL) 4,5,4	LMA1	105
	5 GO(L,L,J)=GO(L,L,J)+GG(L,J)/2.0	LMA1	106
	4 CONTINUE	LMA1	107
61	J=J+1	LMA1	108
	IF(J-4) 62,62,63	LMA1	109
	BEGIN ENERGY DEPENDENT CALCULATION OF LEVEL MATRIX	LMA1	110
63	WOT6,501	LMA1	111
	LINES = 6	LMA1	112
	3 E=EMIN	LMA1	113
30	S=SQRT(E)	LMA1	114
	DO 60 J=1,4	LMA1	115
	6 NR=NO(J)	LMA1	116
	IF(NR) 60,60,55	LMA1	117
55	DO 17 L=1,NR	LMA1	118
	VRE(L,J)=0.0	LMA1	119
17	VIM(L,J)=0.0	LMA1	120
	Q1(J)=0.0	LMA1	121

	Q2(J)=0.0	LMA1 122
	GO TO (57,57,58,58),J	LMA1 123
7	DO 7 L=1,NR	LMA1 124
	G(L,L)=G0(L,L,J)+S*GON(L,J)/(4.0*GFACT(J))	LMA1 125
	LMAX=L-1	LMA1 126
	IF(LMAX) 7,7,8	LMA1 127
8	DO 10 LL=1,LMAX	LMA1 128
	G(L,LL)=G0(L,LL,J)	LMA1 129
10	G(LL,L)=G(L,LL)	LMA1 130
7	CONTINUE	LMA1 131
	DO 11 L=1,NR	LMA1 132
	DO 11 LL=1,NR	LMA1 133
11	Q(L,LL)=G(L,LL)	LMA1 134
	DO 1200 I=1,NR	LMA1 135
	DO 1201 II=1,NR	LMA1 136
201	A(I,II)=Q(I,II)	LMA1 137
200	R(I)=0.0	LMA1 138
	R(1)=1.0	LMA1 139
	I=INVERT(NR)	LMA1 140
	DO 1202 I=1,NR	LMA1 141
	DO 1202 II=1,NR	LMA1 142
1202	Q(I,II)=A(I,II)	LMA1 143
	DO 12 L=1,NR	LMA1 144
	DO 12 LL=1,NR	LMA1 145
12	OINV(L,LL)=G(L,LL)+Q(L,LL)*(ER(L,J)-E)*(ER(LL,J)-E)	LMA1 146
	DO 1203 I=1,NR	LMA1 147
	DO 1204 II=1,NR	LMA1 148
1204	A(I,II)=OINV(I,II)	LMA1 149
203	R(I)=0.0	LMA1 150
	R(1)=1.0	LMA1 151
	I=INVERT(NR)	LMA1 152
	DO 1205 I=1,NR	LMA1 153
	DO 1205 II=1,NR	LMA1 154
205	OINV(I,II)=A(I,II)	LMA1 155
	DO 14 L=1,NR	LMA1 156
	DO 14 LL=1,NR	LMA1 157
14	D(L,LL)=0.0	LMA1 158
	DO 15 L=1,NR	LMA1 159
	DO 15 LL=1,NR	LMA1 160
	DO 15 M=1,NR	LMA1 161
15	D(L,LL)=D(L,LL)+OINV(L,M)*(ER(M,J)-E)*Q(M,LL)	LMA1 162
	DO 20 L=1,NR	LMA1 163
	DO 20 LL=1,NR	LMA1 164
	Y=SQRT(GON(LL,J))	LMA1 165
	VRE(L,J)=VRE(L,J)+D(L,LL)*Y	LMA1 166
20	VIM(L,J)=VIM(L,J)+OINV(L,LL)*Y	LMA1 167
	DO 21 L=1,NR	LMA1 168
	Y=SQRT(GON(L,J))	LMA1 169
	Q1(J)=Q1(J)+VRE(L,J)*Y	LMA1 170
21	Q2(J)=Q2(J)+VIM(L,J)*Y	LMA1 171
	GO TO 60	LMA1 172
58	DO 59 L=1,NR	LMA1 173
	GAM=GG(L,J)+GF(L,J)+GON(L,J)*S/(4.0*GFACT(J))	LMA1 174
	W=(ER(L,J)-E)**2+GAM*GAM	LMA1 175
	Y=SQRT(GON(L,J))	LMA1 176
	VRE(L,J)=Y*(ER(L,J)-E)/W	LMA1 177
	Q1(J)=Q1(J)+Y*VRE(L,J)	LMA1 178
	VIM(L,J)=Y*GAM/W	LMA1 179
59	Q2(J)=Q2(J)+Y*VIM(L,J)	LMA1 180
60	CONTINUE	LMA1 181
	CALCULATE UNBROADENED CROSS SECTIONS	LMA1 182

19	SIGG=0.0	LMA1 183
	SIGA=0.0	LMA1 184
	SIGS=0.0	LMA1 185
	ARG=2.0*WN*S	LMA1 186
	SIGT=(1.0-COS(ARG))/S*2.0	LMA1 187
	DO 22 J=1,4	LMA1 188
	NR=NO(J)	LMA1 189
	IF(NR) 22,22,66	LMA1 190
66	DO 23 L=1,NR	LMA1 191
	SIGG=SIGG+GG(L,J)*(VRE(L,J)**2+VIM(L,J)**2)	LMA1 192
	GN(L,J)=GON(L,J)*S/(2.0*GFACT(J))	LMA1 193
23	SIGS=SIGS+GN(L,J)*(VRE(L,J)**2+VIM(L,J)**2)	LMA1 194
	SIGA=SIGA+Q2(J)	LMA1 195
	SIGT=SIGT-Q1(J)*SIN(ARG)+Q2(J)*COS(ARG)	LMA1 196
22	CONTINUE	LMA1 197
	TOT=SIGT*CC/S	LMA1 198
	CAP=SIGG*CC/(2.0*S)	LMA1 199
	SIGS=SIGS*CC/(2.0*S)	LMA1 200
	FIS=SIGA*CC/S-SIGS-CAP	LMA1 201
	SCAT=TOT-FIS-CAP	LMA1 202
	IF (IPUNCH) 64,65,64	LMA1 203
64	PUNCH 600,E,TOT,FIS,CAP,SCAT	LMA1 204
65	WOT6,500,E,TOT,FIS,CAP,SCAT	LMA1 205
	LINES = LINES + 1	LMA1 206
	IF (LINES - 60) 370, 371, 371	LMA1 207
371	WOT 6, 502	LMA1 208
	LINES = 6	LMA1 209
370	E = E + DELE	LMA1 210
	E=E+DELE	LMA1 211
	IF(E-EMAX) 30,30,24	LMA1 212
24	WOT 6, 602	LMA1 213
	GO TO 1000	LMA1 214
100	FORMAT(3E15.8)	LMA1 215
102	FORMAT(2I5)	LMA1 216
101	FORMAT(4E15.8)	LMA1 217
103	FORMAT(E15.8,I15)	LMA1 218
204	FORMAT (1H 48X, I2, 5E16.6)	LMA1 219
08	FORMAT (1H-, 29X, 5HISW=1, 7X, 1HK, 8X, 2HER, 13X, 3HGON, 12X,	LMA1 220
	2 2HGG, 13X, 2HGF, 13X, 2HGN, 12X, 4HTETA/)	LMA1 221
209	FORMAT (1H-, 29X, 5HISW=2, 15X, 1HK, 9X, 2HER, 14X, 3HGON, 13X,	LMA1 222
	2 2HGG, 14X, 2HGF, 14X, 2HGN/)	LMA1 223
350	FORMAT (1H-,29X,16MATRIX GO(K,L,J),4X,1HK,5(6X,A5,I2,A3))	LMA1 224
351	FORMAT (1H0, 50X, 5(6X, A5, I2, A3))	LMA1 225
500	FORMAT(5F20.4)	LMA1 226
501	FORMAT (1H1 39X, 32H---UNBROADENED CROSS SECTIONS---//14X, 6HENELMA1 227	
	2RGY 15X, 5HTOTAL 13X, 7HFISSION 13X, 7HCAPTURE 10X, 10HSCATTERING/LMA1 228	
	3/)	LMA1 229
502	FORMAT (1H1 33X, 44H---UNBROADENED CROSS SECTIONS (CONTINUED)---//LMA1 230	
	2/14X, 6HENERGY 15X, 5HTOTAL 13X, 7HFISSION 13X, 7HCAPTURE 10X,	LMA1 231
	3 10HSCATTERING//)	LMA1 232
600	FORMAT(5F15.4)	LMA1 233
601	FORMAT (1H)	LMA1 234
602	FORMAT (1H1)	LMA1 235
2000	FORMAT (1H1, 55X,23HP R O G R A M L E M A//32X, 68HCALCULATION OLMA1 236	
	2F RESONANCE CROSS SECTIONS WITH LEVEL MATRIX INVERSION, / 39X, 53HLMA1 237	
	3USING SUBROUTINE DNV1 FOR DOUBLE PRECISION ARITHMETIC//)	LMA1 238
2010	FORMAT (7H ATM = F7.2,10X, 5HWN = 1PE10.4,10X, 5HCC = 1PE10.4,10X,LMA1 239	
	27HEMIN = OPF6.3,10X,7HEMAX = F6.3, 8X, 7HDELE = F5.4//)	LMA1 240
1011	FORMAT (1H 48X, I2, 5E16.6)	LMA1 241
2012	FORMAT (1H 40X, I2, 6E15.6)	LMA1 242
2013	FORMAT (1H1, 43X, 47HP R O G R A M L E M A (C O N T I N U E DLMA1 243	

2)/////)	LMA1 244
2C20 FORMAT (3H-J=11, 7X, 6HGFACT(11, 3H) =E10.4, 6X, 3HNO(11, 4H) = 12/)	LMA1 245
FORMAT (1H-, 49X, 1HK, 10X, 2HER, 9X, 3HGON, 10X, 2HGG, 10X, 2HGF,	LMA1 246
2 10X, 2HGN/)	LMA1 247
END	LMA1 248

FUNCTION INVERT(DUMMY)	LMA1 249
FUNCTION INVERT- INVERSION OF A REAL MATRIX BY DOUBLE PRECISION	LMA1 250
DIMENSION A(50,25),R(50),PIVOT(50),DET(2)	LMA1 251
COMMON A,R,PIVOT,DET,NR	LMA1 252
DIMENSION IPIVOT(25),INDEX(25,2)	LMA1 253
N=NR	LMA1 254
1 DO 2 I=1,NR	LMA1 255
A(2*N,I)=A(N,I)	LMA1 256
2 A(2*N-1,I)=0.0	LMA1 257
R(2*N)=R(N)	LMA1 258
R(3*N-1)=0.0	LMA1 259
N=N-1	LMA1 260
IF(N) 3,3,1	LMA1 261
3 INVERT=DNV1(NR,50,1,A,R,DET(2),PIVOT,IPIVOT,INDEX,25)	LMA1 262
DO 5 I=1,NR	LMA1 263
DO 4 II=1,NR	LMA1 264
4 A(I,II)=ROUNDF(A(2*I,II),A(2*I-1,II))	LMA1 265
5 R(I)=ROUNDF(R(2*I),R(2*I-1))	LMA1 266
RETURN	LMA1 267
END	LMA1 268

FUNCTION ROUND--ROUND DOUBLE PRECISION NUMBER TO SINGLE PRECISION	LMA1 269
ENTRY ROUND	LMA1 270
UND FRN	LMA1 271
TRA 1,4	LMA1 272
END	LMA1 273

* UOI	DOUBLE	PRECISICN	SQRT	LMA1	274
	ENTRY	DSQRT		LMA1	275
	EVEN			LMA1	276
U	BSS	2	TEMP STORAGE	LMA1	277
DSQRT	CLA*	1,4		LMA1	278
	TMI	ERR	ERROR RETURN, X NEGATIVE	LMA1	279
	SXA	AA,1		LMA1	280
	ACL	=0177000000000	(P+128+127)*2**-8	LMA1	281
	XCL			LMA1	282
	LGL	8		LMA1	283
	ALS	27	((P/2)-1)+128 = NEW CHARACTERISTIC	LMA1	284
	STO	COMMON+1		LMA1	285
	TQP	*+2	MQ IS + IF P IS ODD, USE Y/2 FOR Y	LMA1	286
	LLS	1	MQ IS - IF P IS EVEN, USE Y	LMA1	287
	LLS	5	6 BITS OF Y USED FOR TABLE LOOKUP	LMA1	288
	PAX	0,1	OF INITIAL APPROXIMATION	LMA1	289
	TXL	A,1,15	TRA IF ARG NOT NORMAL	LMA1	290
	LLS	25	AC=Y/16 OR -Y/16	LMA1	291
	SLW	COMMON		LMA1	292
	VDH	TABLE+16,1,19	DIVIDE BY INITIAL APPROX. (X0/4)	LMA1	293
	LLS	16	MQ=Y/4X0 OR -Y/4X0	LMA1	294
	XCL		AC=Y/4X0, MAYBE P BIT IS 1	LMA1	295
	ADD	TABLE+16,1	AC=((Y/2X0)+(X0/2))/2=X1/2	LMA1	296
	STO	COMMON+2		LMA1	297
	CLA	COMMON	AC=Y/16	LMA1	298
	DVH	COMMON+2		LMA1	299
	XCA		AC=Y/8X1	LMA1	300
	ALS	2		LMA1	301
	ADD	COMMON+2	AC=(Y/2X1)+(X1/2)=X2, FIXED POINT	LMA1	302
	LRS	8		LMA1	303
	RQL	28		LMA1	304
	ADD	COMMON+1	X2/2, FLOATING POINT	LMA1	305
*				LMA1	306
	DST	U		LMA1	307
	DLD*	1,4	ACMQ=A	LMA1	308
	DFDH	U	ACMQ=2A/X2	LMA1	309
	SUB	=0002000000000	ACMQ=A/2X2	LMA1	310
	DFAD	U	ACMQ=X3=SQRT(A)	LMA1	311
	DST*	2,4	STORE RESULT	LMA1	312
*				LMA1	313
AA	AXT	** ,1	RESTORE XR1	LMA1	314
	TOV	3,4	RETURN	LMA1	315
	TRA	3,4	RETURN	LMA1	316
*				LMA1	317
A	DLD*	1,4		LMA1	318
	TZE	AA-1	ZERO RETURN	LMA1	319
	SXA	SB,4		LMA1	320
	CALL	SYSWOT	PRINT ERROR MESSAGE	LMA1	321
	TIX	M, ,4		LMA1	322
SB	AXT	** ,4		LMA1	323
	TRA	AA		LMA1	324
M	BCI	4, ODSQRT ARG NOT NORMAL		LMA1	325
R	SXA	SA,4		LMA1	326
	CALL	SYSWOT		LMA1	327
	TIX	MA, ,4		LMA1	328
SA	AXT	** ,4		LMA1	329
	TRA	3,4		LMA1	330
MA	BCI	4, ODSQRT ARG NEGATIVE		LMA1	331
	DEC	.9960861B2	.25*SQRT(127/128)	LMA1	332
	DEC	.9882118B2		LMA1	333

DEC	.9802742B2		LMA1 334
DEC	.9722718B2		LMA1 335
DEC	.9642030B2		LMA1 336
DEC	.9560662B2		LMA1 337
DEC	.9478594B2		LMA1 338
DEC	.9395810B2		LMA1 339
DEC	.9312290B2		LMA1 340
DEC	.9228014B2		LMA1 341
DEC	.9142962B2		LMA1 342
DEC	.9057110B2		LMA1 343
DEC	.8970438B2		LMA1 344
DEC	.8882919B2		LMA1 345
DEC	.8794530B2		LMA1 346
DEC	.8705243B2		LMA1 347
DEC	.8615030B2		LMA1 348
DEC	.8523864B2		LMA1 349
DEC	.8431711B2		LMA1 350
DEC	.8338540B2		LMA1 351
DEC	.8244316B2		LMA1 352
DEC	.8149003B2		LMA1 353
DEC	.8052562B2		LMA1 354
DEC	.7954951B2		LMA1 355
DEC	.7856128B2		LMA1 356
DEC	.7756046B2		LMA1 357
DEC	.7654655B2		LMA1 358
DEC	.7551904B2		LMA1 359
DEC	.7447735B2		LMA1 360
DEC	.7342088B2		LMA1 361
DEC	.7234898B2		LMA1 362
DEC	.7126096B2		LMA1 363
DEC	.7015608B2		LMA1 364
DEC	.6903351B2		LMA1 365
DEC	.6789238B2		LMA1 366
DEC	.6673174B2		LMA1 367
DEC	.6555055B2		LMA1 368
DEC	.6434769B2		LMA1 369
DEC	.6312191B2		LMA1 370
DEC	.6187184B2		LMA1 371
DEC	.6059600B2		LMA1 372
DEC	.5929271B2		LMA1 373
DEC	.5796012B2		LMA1 374
DEC	.5659616B2		LMA1 375
DEC	.5519851B2		LMA1 376
DEC	.5376453B2	ETC	LMA1 377
DEC	.5229125B2	.25*SQRT(35/128)	LMA1 378
TABLE DEC	.5077524B2	.25*SQRT(33/128)	LMA1 379
C MMON BSS	3	TEMP STORAGE	LMA1 380
END			LMA1 381

PROGRAM LEMA-----13 JULY, 1965 (ILLIAC II VERSION) ...February 1966

LEMA 2

	LMA2	1
	LMA2	2
	LMA2	3
C CALCULATES INTERFERING RESONANCE CROSS SECTIONS BY LEVEL MATRIX	LMA2	4
C INVERSION AND BY KAPUR PEIERLS POLES EXPANSION.	LMA2	5
C THIS FORTRAN II VERSION IS LIMITED TO 2 INTERFERING RESONANCES PER	LMA2	6
C SPIN STATE, AND 2 ADDITIONAL NON-INTERFERING RESONANCES PER SPIN	LMA2	7
C STATE.	LMA2	8
C OPERATES ON EITHER ILLIAC II OR IBM 7094.	LMA2	9
	LMA2	10
C INDEX ISW = 1 REPRESENTS THE CASE OF GIVEN VALUES OF THE ANGLES TETA	LMA2	11
C INDEX ISW = 2 REPRESENTS THE CASE OF GIVEN VALUES OF THE COSINE OF	LMA2	12
C THE ANGLES BETWEEN TWO INTERFERING FISSION CHANNELS.	LMA2	13
	LMA2	14
C INDEX IPUNCH = 1 PROVIDES CALCULATED CROSS SECTIONS ON PUNCHED CARDS--	LMA2	15
C WITH IPUNCH = 0 THESE CARDS ARE NOT PUNCHED--	LMA2	16
	LMA2	17
C INDEX IPER = 1 PROVIDES FIRST ORDER PERTURBATION CORRECTIONS.	LMA2	18
C INDEX IPER = 0 SKIPS THE FIRST ORDER CORRECTIONS.	LMA2	19
	LMA2	20
DIMENSION GO(2,2,4),G(2,2),OINV(2,2),D(2,2),Q(2,2),VRE(2,4),	LMA2	21
1 VIM(2,4),TETA(2,4),ER(2,4),GON(2,4),GG(2,4),GF(2,4),GN(LMA2		22
22,4),Q1(4),Q2(4),NO(4),GFACT(4),W(2,2),WW(2,2),FU(2),GU(2)	LMA2	23
DIMENSION SRE(2,2),SIM(2,2),WRE(2),WGIM(2),WFRE(2),WFIM(2),PHIO(LMA2		24
12,2),SIGO(2,2),RHIO(2,2),TIGO(2,2),WGOR(2,2),WGOI(2,2),PHI(2,2),SILMA2		25
2G(2,2),RHI(2,2),TIG(2,2),ALFA(2),BETA(2),GFS(2),HFS(2),GGA(2),HGA(LMA2		26
32),ALG(2,4),BEG(2,4),CAPG(2,4),CAPH(2,4),FISG(2,4),FISH(2,4)	LMA2	27
DIMENSION GA(2,2),DE(2,4),DW(2,4),WCRE(2),WCIM(2),WS(4,4),US(4,4)	LMA2	28
DIMENSION STORS(2,2,4),STOIS(2,2,4),STUS(4,4,4)	LMA2	29
CALL FTRAP	LMA2	30
	LMA2	31
NRUN = 0	LMA2	32
1000 NRUN = NRUN + 1	LMA2	33
READ 102, ISW, IPUNCH, IPER	LMA2	34
READ 100, ATM, WN, CC, EMIN, EMAX, DELE	LMA2	35
KPUNCH=0	LMA2	36
WOT6,3000	LMA2	37
WOT 6, 3010, ATM, WN, CC, EMIN, EMAX, DELE	LMA2	38
DO 1 J=1,4	LMA2	39
READ 103,GFACT(J),NO(J)	LMA2	40
WOT 6, 3020, J, J, GFACT(J), J, NO(J)	LMA2	41
NR=NO(J)	LMA2	42
IF(NR) 1,1,50	LMA2	43
50 GO TO(51,51,52,52),J	LMA2	44
51 GO TO(53,54),ISW	LMA2	45
52 WOT 6, 3030	LMA2	46
DO 35 K = 1,NR	LMA2	47
DO 35 L=1,NR	LMA2	48
GO(K,L,J)=0.0	LMA2	49
TETA(K,J)=0.0	LMA2	50
READ 101,ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J)	LMA2	51
55 WOT6,3011,K,ER(K,J), GON(K,J), GG(K,J), GF(K,J), GN(K,J)	LMA2	52
GO TO 1	LMA2	53
53 WOT 6, 208	LMA2	54
DO 42 K = 1,NR	LMA2	55
READ 101, ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J),TETA(K,J)	LMA2	56
42 WOT6,3012,K,ER(K,J), GON(K,J), GG(K,J), GF(K,J), GN(K,J),TETA(K,J)	LMA2	57
GO TO 1	LMA2	58
54 WOT 6, 209	LMA2	59

	DO 36 K = 1,NR	LMA2	60
	READ 101, ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J)	LMA2	61
36	WOT6,3011,K,ER(K,J), GON(K,J), GG(K,J), GF(K,J), GN(K,J)	LMA2	62
	WOT 6, 210	LMA2	63
	DO 37 K=1,NR	LMA2	64
	READ 101,(GO(K,L,J),L=1,NR)	LMA2	65
37	WOT6,204,K, (GO(K,L,J), L = 1,NR)	LMA2	66
1	CONTINUE	LMA2	67
	J=1	LMA2	68
62	NR=NO(J)	LMA2	69
	IF(NR) 61,61,2	LMA2	70
2	CONTINUE	LMA2	71
	DO 4 L =1,NR	LMA2	72
	DO 4 LL=1,NR	LMA2	73
	Y=GF(L,J)*GF(LL,J)	LMA2	74
	YY=SQRT(Y)	LMA2	75
	GO TO (139,139,40,40), J	LMA2	76
139	GO TO (39,40),ISW	LMA2	77
39	COSINE=COS(TETA(L,J)-TETA(LL,J))	LMA2	78
	GO(L,LL,J)=YY*COSINE/2.0	LMA2	79
	GO TO 41	LMA2	80
40	GO(L,LL,J)=YY*GO(L,LL,J)/2.0	LMA2	81
41	IF(L-LL) 4,5,4	LMA2	82
5	GO(L,L,J)=GO(L,L,J)+GG(L,J)/2.0	LMA2	83
4	CONTINUE	LMA2	84
51	J=J+1	LMA2	85
	IF(J-4) 62,62,263	LMA2	86
		LMA2	87
	C CALCULATE EIGENVALUES FOR 2 X 2 CASE.	LMA2	88
		LMA2	89
263	NEW=0	LMA2	90
2010	J=1	LMA2	91
163	NR=NO(J)	LMA2	92
	IF(NR) 2011,2011,2012	LMA2	93
2012	IF(NEW) 164,164,2013	LMA2	94
2013	DO 2014 K=1,NR	LMA2	95
	DO2014 L=1,NR	LMA2	96
	SRE(K,L)=STORS(K,L,J)	LMA2	97
2014	SIM(K,L)=STOIS(K,L,J)	LMA2	98
	KMAX=2*NR	LMA2	99
	DO 2015 K=1,KMAX	LMA2	100
	DO 2015 L=1,KMAX	LMA2	101
2015	US(K,L)=STUS(K,L,J)	LMA2	102
	GO TO 2008	LMA2	103
164	EAV=(ER(1,J)+ER(2,J))/2.0	LMA2	104
	S=SQRT(ER(1,J))	LMA2	105
	DG=GON(1,J)*S/(2.0*GFACT(J))	LMA2	106
	GA(1,1) =GO(1,1,J)+DG/2.0	LMA2	107
	S=SQRT(ER(2,J))	LMA2	108
	DG=GON(2,J)*S/(2.0*GFACT(J))	LMA2	109
	GA(2,2) =GO(2,2,J)+DG/2.0	LMA2	110
	GOAV =(GA(1,1) + GA(2,2))/2.0	LMA2	111
	GAGA=(GA(1,1)-GA(2,2))	LMA2	112
	GA(1,2)=GO(1,2,J)	LMA2	113
	GA(2,1)=GA(1,2)	LMA2	114
	ESGA=ER(2,J)-ER(1,J)	LMA2	115
		LMA2	116
	C TEST ON DEGENERATE CASE	LMA2	117
		LMA2	118
	ABSGA = 2.0 * ABSF(GA(1,2))	LMA2	119

IF (ABSF(GAGA) - 1.0E-06) 2050,2050,2054	LMA2 120
2050 IF (ABSF(ESGA - ABSGA) - 1.0E-06) 2051,2051,2052	LMA2 121
C DEGENERATE CASE--SPACING EQUALS 2*GA(1,2).	LMA2 122
	LMA2 123
C51 SQE = SQRT (EAV)	LMA2 124
GA(1,1) = GA(1,1) + GON(1,J) * (SQE - SQRT(ER(1,J))) / (4.0 * 2 GFACT(J))	LMA2 125
GA(2,2) = GA(2,2) + GON(2,J) * (SQE - SQRT(ER(2,J))) / (4.0 * 2 GFACT(J))	LMA2 126
GAGA = GA(1,1) - GA(2,2)	LMA2 127
A = -GAGA / 2.0	LMA2 128
ETA = SQRT (A**4 + (2.0*GA(1,2)*A)**2) - A*A	LMA2 129
ETA = SQRT (ETA/2.0)	LMA2 130
GAL = -A*GA(1,2)/ETA	LMA2 131
FU(1) = EAV - ETA	LMA2 132
FU(2) = EAV + ETA	LMA2 133
GOAV = (GA(1,1) + GA(2,2)) / 2.0	LMA2 134
GU(1) = GOAV + GAL	LMA2 135
GU(2) = GOAV - GAL	LMA2 136
SCE = GA(1,2)**2 - (GA(1,2)-ETA)**2 + (A+GAL)**2	LMA2 137
SCI = 2.0*(GA(1,2)-ETA) * (A+GAL)	LMA2 138
WS = (GA(1,2)**2)*SCE / (SCE**2 + SCI**2)	LMA2 139
QS = -(GA(1,2)**2)*SCI / (SCE**2 + SCI**2)	LMA2 140
XS = WS + SQRT(WS**2 + QS**2)	LMA2 141
SRE(1,1) = SQRT(XS/2.0)	LMA2 142
SIM(1,1) = QS / (2.0*SRE(1,1))	LMA2 143
SRE(2,2) = SRE(1,1)	LMA2 144
SIM(2,2) = SIM(1,1)	LMA2 145
WS = (A+GAL) / GA(1,2)	LMA2 146
QS = (GA(1,2) - ETA) / GA(1,2)	LMA2 147
SRE(2,1) = SRE(1,1)*WS - SIM(1,1)*QS	LMA2 148
SIM(2,1) = SRE(1,1)*QS + SIM(1,1)*WS	LMA2 149
WDT 6, 302	LMA2 150
GO TO 2056	LMA2 151
C52 IF (ESGA - ABSGA) 2053,2051,2054	LMA2 152
	LMA2 153
C DEGENERATE CASE--SPACING IS LESS THAN 2*GA(1,2).	LMA2 154
	LMA2 155
C53 BETAS = -ESGA*ESGA + 4.0*GA(1,2)*GA(1,2)	LMA2 156
BET = SQRT (BETAS)	LMA2 157
FU(1) = EAV	LMA2 158
FU(2) = EAV	LMA2 159
GU(1) = GOAV + BET/2.0	LMA2 160
GU(2) = GOAV - BET/2.0	LMA2 161
EGU = -BET/2.0	LMA2 162
EFU=ER(2,J)-FU(1)	LMA2 163
AA = .5	LMA2 164
BB=-EFU/BET	LMA2 165
XX = .25 + BB**2	LMA2 166
X = SQRT(XX)	LMA2 167
SRE(1,1) = SQRT ((X+AA)/2.0)	LMA2 168
WDT 6, 303	LMA2 169
GO TO 2057	LMA2 170
2C54 ALAL=GAGA*GAGA + GA(1,2)*GA(1,2) *4.0	LMA2 171
ESSE =ESGA*ESGA	LMA2 172
BUZ=(ESSE-ALAL)/2.0	LMA2 173
GUTS=BUZ*BUZ+ESSE*GAGA*GAGA	LMA2 174
BETAS=BUZ+SQRT(GUTS)	LMA2 175
BET=SQRT(BETAS)	LMA2 176
	LMA2 177
	LMA2 178
	LMA2 179

FU(1)=EAV-BET/2.0	LMA2 180
FU(2)=EAV+BET/2.0	LMA2 181
GU(1)=GOAV+GAGA*ESGA/(2.0*BET)	LMA2 182
GU(2)=GOAV-GAGA*ESGA/(2.0*BET)	LMA2 183

C TEST ON SINGLE LEVEL CASE.

2C55 YY=ABSF(GA(1,2))	LMA2 186
IF(YY-1.0E-04) 70,70,71	LMA2 187
CALCULATE 2*2 TRANSFORMATION MATRIX	LMA2 188
71 GUGU = GU(2)-GU(1)	LMA2 189
IF (ABSF(GUGU) - 1.0E-06) 2058,2058,2060	LMA2 190

C DEGENERATE CASE--SPACING IS LARGER THAN 2*GA(1,2)

2C58 DD = ESGA/2.0	LMA2 192
ETA = FU(2) - EAV	LMA2 193
AA = (DD+ETA)/(2.0*ETA)	LMA2 194
SRE(1,1) = SQRT (AA)	LMA2 195
SIM(1,1) = 0.0	LMA2 196
SRE(2,1) = 0.0	LMA2 197
SIM(2,1) = SRE(1,1) * GA(1,2) / (DD + ETA)	LMA2 198
SRE(2,2) = SRE(1,1)	LMA2 199
SIM(2,2) = SIM(1,1)	LMA2 200
WOT 6, 304	LMA2 201
GO TO 2056	LMA2 202

C REGULAR CASE--THE EIGENVALUES OF THE LEVEL MATRIX ARE WELL SEPARATED.

2C60 FUFU=FU(2)-FU(1)	LMA2 203
EFU =ER(2,J)-FU(1)	LMA2 204
EGU =GA(2,2)-GU(1)	LMA2 205
RO =FUFU*FUFU+GUGU*GUGU	LMA2 206
AA=(EFU*FUFU+EGU*GUGU)/RO	LMA2 207
BB=(EFU*GUGU-FUFU*EGU)/RO	LMA2 208
XX=AA**2+BB**2	LMA2 209
X=(AA+SQRT(XX))/2.0	LMA2 210
SRE(1,1)=SQRT(X)	LMA2 211
2C57 SIM(1,1) = BB/(2.0*SRE(1,1))	LMA2 212
SRE(2,2)=SRE(1,1)	LMA2 213
SIM(2,2)=SIM(1,1)	LMA2 214
FEL = EGU * EGU + EFU * EFU	LMA2 215
RO1 = -EGU * GA(1,2) / FEL	LMA2 216
RO2 = EFU * GA(1,2) / FEL	LMA2 217
SRE (2,1) = RO1 * SRE (1,1) - RO2 * SIM (1,1)	LMA2 218
SIM (2,1) = RO1 * SIM (1,1) + RO2 * SRE (1,1)	LMA2 219
2C56 SRE(1,2)=-SRE(2,1)	LMA2 220
SIM(1,2)=-SIM(2,1)	LMA2 221

C FIRST ORDER PERTURBATION CORRECTIONS FOR THE MATRIX S

IF(KPUNCH) 72,72,2039	LMA2 222
2C39 IF(IPER) 72,72,2020	LMA2 223
20 DO 2001 K=1,4	LMA2 224
DO 2001 KK=1,4	LMA2 225
WS(K,KK)=0.0	LMA2 226
2001 US(K,KK)=0.0	LMA2 227
DO 2002 K=1,2	LMA2 228
DO 2002 KK=1,2	LMA2 229
DO 2002 LL= 1 , 2	LMA2 230

	X=SRE(LL,K)*SRE(LL,KK)-SIM(LL,K)*SIM(LL,KK)	LMA2 240
	XX=SIM(LL,K)*SRE(LL,KK)+SRE(LL,K)*SIM(LL,KK)	LMA2 241
	XGO=X*GON(LL,J)/(GFACT(J)*2.0)	LMA2 242
	XXGO=XX*GON(LL,J)/(GFACT(J)*2.0)	LMA2 243
	QRS=SQRT(ER(LL,J))	LMA2 244
	WS(K,KK)=WS(K,KK)+XGO*QRS	LMA2 245
	KP=K+2	LMA2 246
	WS(KP,KK)=WS(KP,KK)+XGO	LMA2 247
	KKP=KK+2	LMA2 248
	WS(K,KKP)=WS(K,KKP)+XXGO*QRS	LMA2 249
2C02	WS(KP,KKP)=WS(KP,KKP)+XXGO	LMA2 250
	DO 2003 L=1,2	LMA2 251
	DO 2003 K=1,2	LMA2 252
	DO 2003 KK=1,2	LMA2 253
	IF(K-KK) 2004,2005,2004	LMA2 254
C04	X=FU(K)-FU(KK)	LMA2 255
	XX=GU(K)-GU(KK)	LMA2 256
	DEX=(X*X+XX*XX)*2.0	LMA2 257
	A=(X*SRE(L,KK)-XX*SIM(L,KK))/DEX	LMA2 258
	B=(X*SIM(L,KK)+XX*SRE(L,KK))/DEX	LMA2 259
	LL=L+2	LMA2 260
	KP=K+2	LMA2 261
	KKP=KK+2	LMA2 262
	US(L,K)=US(L,K)+A*WS(K,KK)-B*WS(K,KKP)	LMA2 263
	US(LL,K)=US(LL,K)+A*WS(K,KKP)+B*WS(K,KK)	LMA2 264
	US(L,KP)=US(L,KP)+A*WS(KP,KK)-B*WS(KP,KKP)	LMA2 265
	US(LL,KP)=US(LL,KP)+A*WS(KP,KKP)+B*WS(KP,KK)	LMA2 266
2C05	CONTINUE	LMA2 267
2C03	CONTINUE	LMA2 268
C06	DO 2007 L=1,2	LMA2 269
	DO 2007 K=1,2	LMA2 270
	LL=L+2	LMA2 271
	KK=K+2	LMA2 272
	SRE(L,K)=US(LL,KK)*SQRT(EMIN)-US(LL,K)+SRE(L,K)	LMA2 273
2C07	SIM(L,K)=US(L,K)-US(L,KK)*SQRT(EMIN)+SIM(L,K)	LMA2 274
	GO TO 72	LMA2 275
C08	SQ1=SQRT(E-DELE)	LMA2 276
	DO 2009 L=1,2	LMA2 277
	DO 2009 K=1,2	LMA2 278
	LL=L+2	LMA2 279
	KK=K+2	LMA2 280
	SRE(L,K)=US(LL,KK)*(S-SQ1)+SRE(L,K)	LMA2 281
C09	SIM(L,K)=US(L,KK)*(SQ1-S)+SIM(L,K)	LMA2 282
	GO TO 72	LMA2 283
7C	SRE(1,1)=1.0	LMA2 284
	SRE(2,1)=0.0	LMA2 285
	SRE(1,2)=0.0	LMA2 286
	SRE(2,2)=1.0	LMA2 287
	DO 73 K=1,2	LMA2 288
	DO 73 KK=1,2	LMA2 289
	KO=K+2	LMA2 290
	KKO=KK+2	LMA2 291
	US(KO,KKO)=0.0	LMA2 292
	US(K,KK)=0.0	LMA2 293
	US(K,KO)=0.0	LMA2 294
	US(KO,K)=0.0	LMA2 295
73	SIM(K,KK)=0.0	LMA2 296
		LMA2 297
C	CALCULATE WIGGLED GAMMAS -- GAMMA ZERO N -- GAMMA CAPTURE AND	LMA2 298
	GAMMA FISSION.	LMA2 299

73	DO 74 K=1,2	LMA2 300
	WRE(K)=0.0	LMA2 301
	WGIM(K)=0.0	LMA2 302
	WCRE(K)=0.0	LMA2 303
	WCIM(K)=0.0	LMA2 304
	WFRE(K)=0.0	LMA2 305
74	WEIM(K)=0.0	LMA2 306
	DO 75 K=1,2	LMA2 307
	DO 75 L=1,2	LMA2 308
	X=SQRT(GF(L,J))	LMA2 309
	WFRE(K)=WFRE(K)+SRE(L,K)*X	LMA2 310
	WFIM(K)=WFIM(K)+SIM(L,K)*X	LMA2 311
	X=SQRT(GG(L,J))	LMA2 312
	WCRE(K)=WCRE(K)+SRE(L,K)*X	LMA2 313
	WCIM(K)=WCIM(K)+SIM(L,K)*X	LMA2 314
	Y=SQRT(GON(L,J))	LMA2 315
	WRE(K)=WRE(K)+Y*SRE(L,K)	LMA2 316
75	WGIM(K)=WGIM(K)+Y*SIM(L,K)	LMA2 317
	IF (KPUNCH) 470,470,471	LMA2 318
470	WOT 6, 3013	LMA2 319
	WOT 6, 211, FU(1), GU(1), FU(2), GU(2)	LMA2 320
	WOT 6, 400	LMA2 321
471	DO 76 K = 1,2	LMA2 322
	IF (KPUNCH) 472,472,473	LMA2 323
472	WOT 6, 402, K, WRE(K), WGIM(K)	LMA2 324
	WOT 6, 403, K, WCRE(K), WCIM(K)	LMA2 325
	WOT 6, 404, K, WFRE(K), WFIM(K)	LMA2 326
		LMA2 327
		LMA2 328
	C CALCULATE ALFA AND BETA FOR TOTAL CROSS SECTION.	LMA2 329
		LMA2 330
473	ALFA(K)=WRE(K)**2-WGIM(K)**2	LMA2 331
76	BETA(K)= 2.*WRE(K)*WGIM(K)	LMA2 332
	DO 77 K=1,2	LMA2 333
	DO 77 KK=1,2	LMA2 334
	WGOR (K,KK) = 0.	LMA2 335
	WGOI (K,KK) = 0.	LMA2 336
	PHIO(K,KK)=0.0	LMA2 337
	SIGO(K,KK)=0.0	LMA2 338
	RHIO(K,KK)=0.0	LMA2 339
77	TIGO(K,KK)=0.0	LMA2 340
		LMA2 341
	C CALCULATE MATRICES PHI, SIGMA FOR FISSION AND CAPTURE.	LMA2 342
		LMA2 343
	DO 78 K=1,2	LMA2 344
	DO 78 KK=1,2	LMA2 345
	DO 79 L=1,2	LMA2 346
	DO 79 LL=1,2	LMA2 347
	X=SRE(L,K)*SRE(LL,KK)+SIM(L,K)*SIM(LL,KK)	LMA2 348
	XX=SIM(L,K)*SRE(LL,KK)-SRE(L,K)*SIM(LL,KK)	LMA2 349
	IF (L-LL) 80,81,80	LMA2 350
81	PHIO(K,KK)=PHIO(K,KK)+X*GF(L,J)	LMA2 351
	SIGO(K,KK)=SIGO(K,KK)+XX*GF(L,J)	LMA2 352
	RHIO(K,KK)=RHIO(K,KK)+X*GG(L,J)	LMA2 353
	TIGO(K,KK)=TIGO(K,KK)+XX*GG(L,J)	LMA2 354
	WGOR (K,KK) = WGOR (K,KK) + X*GON(L,J)	LMA2 355
	WGOI (K,KK) = WGOI (K,KK) + XX*GON(L,J)	LMA2 356
	GO TO 79	LMA2 357
80	PHIO(K,KK)=PHIO(K,KK)+2.0*GA(1,2)*X	LMA2 358
	SIGO(K,KK)=SIGO(K,KK)+2.0*GA(1,2)*XX	LMA2 359

	YGO=SQRT(GON(L,J)*GON(LL,J))	LMA2 360
	WGOR(K,KK)=WGOR(K,KK)+X*YGO	LMA2 361
	WGOI(K,KK)=WGOI(K,KK)+XX*YGO	LMA2 362
79	CONTINUE	LMA2 363
7E	CONTINUE	LMA2 364
	DO 82 K=1,2	LMA2 365
	DO 82 KK=1,2	LMA2 366
	PHI(K,KK)=WGOR(K,KK)*PHIO(K,KK)-WGOI(K,KK)*SIGO(K,KK)	LMA2 367
	RHI(K,KK)=WGOR(K,KK)*RHIO(K,KK)-WGOI(K,KK)*TIGO(K,KK)	LMA2 368
	SIG(K,KK)=WGOR(K,KK)*SIGO(K,KK)+WGOI(K,KK)*PHIO(K,KK)	LMA2 369
82	TIG(K,KK)=WGOR(K,KK)*TIGO(K,KK)+WGOI(K,KK)*RHIO(K,KK)	LMA2 370
		LMA2 371
	C CALCULATE THE MULTILEVEL PARAMETERS FOR REACTION CROSS SECTIONS.	LMA2 372
		LMA2 373
	DO 83 K=1,2	LMA2 374
	GFS(K)=0.0	LMA2 375
	GGA(K)=0.0	LMA2 376
	HFS(K)=0.0	LMA2 377
83	HGA(K)=0.0	LMA2 378
	DO 84 K=1,2	LMA2 379
	DO 84 KK=1,2	LMA2 380
	GUGU=GU(K)+GU(KK)	LMA2 381
	FUFU=FU(K)-FU(KK)	LMA2 382
	FUNU=GUGU*GUGU+FUFU*FUFU	LMA2 383
	GFS(K)=GFS(K)+(GUGU*PHI(K,KK)+FUFU*SIG(K,KK))/FUNU	LMA2 384
	GGA(K)=GGA(K)+(GUGU*RHI(K,KK)+FUFU*TIG(K,KK))/FUNU	LMA2 385
	HFS(K)=HFS(K)+(GUGU*SIG(K,KK)-FUFU*PHI(K,KK))/FUNU	LMA2 386
84	HGA(K)=HGA(K)+(GUGU*TIG(K,KK)-FUFU*RHI(K,KK))/FUNU	LMA2 387
	DO 85 K=1,2	LMA2 388
	ALG(K,J)=ALFA(K)	LMA2 389
	BEG(K,J)=BETA(K)	LMA2 390
	CAPG(K,J)=GGA(K)	LMA2 391
	CAPH(K,J)=HGA(K)	LMA2 392
	FISG(K,J)=GFS(K)	LMA2 393
	FISH(K,J)=HFS(K)	LMA2 394
	DE(K,J)=FU(K)	LMA2 395
85	DW(K,J)=GU(K)	LMA2 396
	IF(KPUNCH) 2036,2036,2037	LMA2 397
2C36	WDI 6,405	LMA2 398
	DO 2038 K=1,NR	LMA2 399
	ALFCT = ALFA(K) / GU(K)	LMA2 400
	BEFCT = BETA(K) / GU(K)	LMA2 401
	GFFCT = GFS(K) / GU(K)	LMA2 402
	HFFCT = HFS(K) / GU(K)	LMA2 403
	GGFCT = GGA(K) / GU(K)	LMA2 404
	HGFCT = HGA(K) / GU(K)	LMA2 405
	PUNCH 1164, FU(K), GU(K), ALFCT, BEFCT, NRUN, K	LMA2 406
	PUNCH 1165, FU(K), GU(K), GFFCT, HFFCT, NRUN, K	LMA2 407
	PUNCH 1166, FU(K), GU(K), GGFCT, HGFCT, NRUN, K	LMA2 408
2C38	WDI 6, 406, K, ALFA(K), GFS(K), GGA(K), BETA(K), HFS(K), HGA(K)	LMA2 409
2C37	DO 2016 K=1,4	LMA2 410
	DO 2016 L=1,4	LMA2 411
16	STUS(K,L,J)=US(K,L)	LMA2 412
	DO 2017 K=1,2	LMA2 413
	DO 2017 L=1,2	LMA2 414
	STORS(K,L,J)=SRE(K,L)	LMA2 415
2C17	STOIS(K,L,J)=SIM(K,L)	LMA2 416
2C11	J=J+1	LMA2 417
	IF(J-4) 163,163,2035	LMA2 418
		LMA2 419

C BEGIN ENERGY-DEPENDENT CALCULATION OF LEVEL MATRIX.

		LMA2 420
		LMA2 421
35	IF (KPUNCH) 474,474,2022	LMA2 422
4	WOT 6, 501	LMA2 423
	LINES = 8	LMA2 424
	KPUNCH = 1	LMA2 425
	IF (IPER) 2022,2022,2010	LMA2 426
2022	IF(NEW) 3,3,30	LMA2 427
3	E=E MIN	LMA2 428
30	S=SQRT(E)	LMA2 429
	DO 60 J=1,4	LMA2 430
6	NR=NO(J)	LMA2 431
	IF(NR) 60,60,55	LMA2 432
55	DO 17 L=1,NR	LMA2 433
	VRE(L,J)=0.0	LMA2 434
17	VIM(L,J)=0.0	LMA2 435
	Q1(J)=0.0	LMA2 436
	Q2(J)=0.0	LMA2 437
	GO TO (57,57,58,58),J	LMA2 438
57	DO 7 L=1,NR	LMA2 439
	G(L,L)=GO(L,L,J)+S*GON(L,J)/(4.0*GFACT(J))	LMA2 440
	LMAX=L-1	LMA2 441
	IF(LMAX) 7,7,8	LMA2 442
8	DO 10 LL=1,LMAX	LMA2 443
	G(L,LL)=GO(L,LL,J)	LMA2 444
10	G(LL,L)=G(L,LL)	LMA2 445
7	CONTINUE	LMA2 446
	DO 11 L=1,NR	LMA2 447
	DO 11 LL=1,NR	LMA2 448
11	Q(L,LL)=G(L,LL)	LMA2 449
	DO 200 K=1,NR	LMA2 450
	DO 200 KK=1,NR	LMA2 451
200	W(K,KK) = Q(K,KK)	LMA2 452
	NCC=1	LMA2 453
	GO TO 220	LMA2 454
220	DET =W(1,1)*W(2,2)-W(2,1)*W(1,2)	LMA2 455
	WW(1,1)=W(2,2)/DET	LMA2 456
	WW(2,2)=W(1,1)/DET	LMA2 457
	WW(1,2)=-W(1,2)/DET	LMA2 458
	WW(2,1)=-W(2,1)/DET	LMA2 459
	IF(NCC-1) 221,221,222	LMA2 460
221	DO 202 K=1,NR	LMA2 461
	DO 202 KK=1,NR	LMA2 462
202	Q(K,KK)=WW(K,KK)	LMA2 463
	GO TO 223	LMA2 464
222	DO 203 K=1,NR	LMA2 465
	DO 203 KK=1,NR	LMA2 466
203	OINV(K,KK)=WW(K,KK)	LMA2 467
	GO TO 224	LMA2 468
223	DO 12 L=1,NR	LMA2 469
	DO 12 LL=1,NR	LMA2 470
12	OINV(L,LL)=G(L,LL)+Q(L,LL)*(ER(L,J)-E)*(ER(LL,J)-E)	LMA2 471
	DO 201 K=1,NR	LMA2 472
	DO 201 KK=1,NR	LMA2 473
201	W(K,KK) = OINV(K,KK)	LMA2 474
	NCC=2	LMA2 475
	GO TO 220	LMA2 476
224	DO 14 L=1,NR	LMA2 477
	DO 14 LL=1,NR	LMA2 478
14	D(L,LL)=0.0	LMA2 479

	DO 15 L=1,NR	LMA2 480
	DO 15 LL=1,NR	LMA2 481
	DO 15 M=1,NR	LMA2 482
15	D(L,LL)=D(L,LL)+OINV(L,M)*(ER(M,J)-E)*Q(M,LL)	LMA2 483
	DO 20 L=1,NR	LMA2 484
	DO 20 LL=1,NR	LMA2 485
	Y=SQRT(GON(LL,J))	LMA2 486
	VRE(L,J)=VRE(L,J)+D(L,LL)*Y	LMA2 487
20	VIM(L,J)=VIM(L,J)+CINV(L,LL)*Y	LMA2 488
	DO 21 L=1,NR	LMA2 489
	Y=SQRT(GON(L,J))	LMA2 490
	Q1(J)=Q1(J)+VRE(L,J)*Y	LMA2 491
1	Q2(J)=Q2(J)+VIM(L,J)*Y	LMA2 492
	GO TO 60	LMA2 493
58	DO 59 L=1,NR	LMA2 494
-	GAM=GG(L,J)+GF(L,J)+GON(L,J)*S/(4.0*GFACT(J))	LMA2 495
	Z=(ER(L,J)-E)**2+GAM*GAM	LMA2 496
	Y=SQRT(GON(L,J))	LMA2 497
	VRE(L,J)=Y*(ER(L,J)-E)/Z	LMA2 498
	Q1(J)=Q1(J)+Y*VRE(L,J)	LMA2 499
	VIM(L,J)=Y*GAM/Z	LMA2 500
59	Q2(J)=Q2(J)+Y*VIM(L,J)	LMA2 501
10	CONTINUE	LMA2 502
		LMA2 503
	C CALCULATE UNBROADENED CROSS SECTIONS FROM LEVEL MATRIX INVERSION.	LMA2 504
		LMA2 505
9	SIGG=0.0	LMA2 506
	SIGA=0.0	LMA2 507
	SIGS=0.0	LMA2 508
	ARG=2.0*WN*S	LMA2 509
	SIGT=(1.0-COS(ARG))/S*2.0	LMA2 510
	DO 22 J=1,4	LMA2 511
	NR=NO(J)	LMA2 512
	IF(NR) 22,22,66	LMA2 513
06	DO 23 L=1,NR	LMA2 514
	SIGG=SIGG+GG(L,J)*(VRE(L,J)**2+VIM(L,J)**2)	LMA2 515
	GN(L,J)=GON(L,J)*S/(2.0*GFACT(J))	LMA2 516
13	SIGS=SIGS+GN(L,J)*(VRE(L,J)**2+VIM(L,J)**2)	LMA2 517
	SIGA=SIGA+Q2(J)	LMA2 518
	SIGT=SIGT-Q1(J)*SIN(ARG)+Q2(J)*COS(ARG)	LMA2 519
12	CONTINUE	LMA2 520
	TOT=SIGT*CC/S	LMA2 521
	CAP=SIGG*CC/(2.0*S)	LMA2 522
	SIGS=SIGS*CC/(2.0*S)	LMA2 523
	FIS=SIGA*CC/S-SIGS-CAP	LMA2 524
	SCAT=TOT-FIS-CAP	LMA2 525
	IF (IPUNCH) 64,65,64	LMA2 526
14	PUNCH 600, E, TOT, FIS, CAP, SCAT	LMA2 527
65	TOT2 = TOT	LMA2 528
	FIS2 = FIS	LMA2 529
	CAP2 = CAP	LMA2 530
	SCAT2 = SCAT	LMA2 531
	J=1	LMA2 532
	TOT=0.0	LMA2 533
	CAP=0.0	LMA2 534
	FIS=0.0	LMA2 535
18	NR=NO(J)	LMA2 536
	IF(NR) 90,90,91	LMA2 537
91	DO 86 K=1,2	LMA2 538
	FU(K)=DE(K,J)	LMA2 539

	GU(K)=DW(K,J)	LMA2 540
	ALFA(K)=ALG(K,J)	LMA2 541
	BETA(K)=BEG(K,J)	LMA2 542
	GFS(K)=FISG(K,J)	LMA2 543
	HFS(K)=FISH(K,J)	LMA2 544
	GGA(K)=CAPG(K,J)	LMA2 545
86	HGA(K)=CAPH(K,J)	LMA2 546
		LMA2 547
	C CALCULATE UNBROADENED CROSS SECTIONS FROM PARAMETERS.	LMA2 548
		LMA2 549
	ARG=2.0*WN*SQRT(E)	LMA2 550
	CO=COS(ARG)	LMA2 551
	SI = SIN(ARG)	LMA2 552
	PART1= .0	LMA2 553
	PART2=0.0	LMA2 554
	DO 87 K=1,2	LMA2 555
	X=FU(K)-E	LMA2 556
	BW=X*X+GU(K)*GU(K)	LMA2 557
	SEV=GU(K)/BW	LMA2 558
	SOD=X/BW	LMA2 559
	PART1=PART1 + SOD*ALFA(K) -SEV*BETA(K)	LMA2 560
	PART2= PART2 + SOD*BETA(K)+ SEV*ALFA(K)	LMA2 561
	FIS=FIS+GFS(K)*SEV+HFS(K)*SOD	LMA2 562
87	CAP=CAP+HGA(K)*SOD+GGA(K)*SEV	LMA2 563
	TOT=TOT+CO*PART2-SI*PART1	LMA2 564
90	J = J+1	LMA2 565
	IF(J-4) 88,88,89	LMA2 566
85	CAP=CAP*CC/S	LMA2 567
	FIS=FIS*CC/S	LMA2 568
	TOT=TOT*CC/S+ 2.0*CC*(1.0-COS(ARG))/E	LMA2 569
	SCAT=TOT-CAP-FIS	LMA2 570
	WOT 6, 500, E, TOT2, TOT, FIS2, FIS, CAP2, CAP, SCAT2, SCAT	LMA2 571
	LINES = LINES + 1	LMA2 572
	IF (LINES - 56) 420,421,421	LMA2 573
421	WOT 6, 3031	LMA2 574
	IF (IPER) 92, 92, 93	LMA2 575
92	WOT 6, 3032	LMA2 576
	GO TO 94	LMA2 577
93	WOT 6, 3033	LMA2 578
94	WOT 6, 501	LMA2 579
	LINES = 8	LMA2 580
420	E = E + DELE	LMA2 581
	NEW=1	LMA2 582
	S=SQRT(E)	LMA2 583
	IF(E-EMAX) 2021,2021,24	LMA2 584
2021	IF(IPER) 2022,2022,2010	LMA2 585
24	IF (LINES - 8) 1000, 1000, 97	LMA2 586
97	DO 95 K = LINES,56	LMA2 587
95	WOT 6, 450	LMA2 588
	WOT 6, 3031	LMA2 589
	IF (IPER) 98, 98, 99	LMA2 590
98	WOT 6, 3032	LMA2 591
	GO TO 1000	LMA2 592
99	WOT 6, 3033	LMA2 593
	GO TO 1000	LMA2 594
		LMA2 595
		LMA2 596
100	FORMAT (3E15.0)	LMA2 597
101	FORMAT (4E15.8)	LMA2 598
102	FORMAT (3I5)	LMA2 599

103	FORMAT (F15.0, 15)	LMA2 600
	FORMAT (1H 49X, 11, 2E16.6)	LMA2 601
208	FORMAT (1H-, 29X, 5HISW=1, 7X, 1HK, 8X, 2HER, 13X, 3HGON, 12X, 2 2HGG, 13X, 2HGF, 13X, 2HGN, 12X, 4HTETA/)	LMA2 602
		LMA2 603
209	FORMAT (1H-, 29X, 5HISW=2, 15X, 1HK, 9X, 2HER, 14X, 3HGON, 13X, 2 2HGG, 14X, 2HGF, 14X, 2HGN/)	LMA2 604
		LMA2 605
210	FORMAT (1H-, 29X, 16HMATRIX GO(K,L,J), 4X, 1HK, 6X, 9HGO(K,1,J), 2X, 9HGO(K,2,J)/)	LMA2 606
		LMA2 607
211	FORMAT (36H EIGENVALUES OF THE LEVEL MATRIX----//49X, 2HMU, 14X, 2 2HNU//31X, 3HK=1 F20.5, F15.5 //31X, 3HK=2 F20.5, F15.5//))	LMA2 608
		LMA2 609
302	FORMAT(132H-***ATTENTION*** THIS PROGRAM HAS PROCEEDED THROUGH 2THE BRANCH CONTAINING THE TEST ON THE DEGENERATE CASE, WITH GA(1,1 3) = GA(2,2)/20X, 28HAND ER(2)-ER(1) = 2*GA(1,2).)	LMA2 610
		LMA2 611
303	FORMAT(132H-***ATTENTION*** THIS PROGRAM HAS PROCEEDED THROUGH 2THE BRANCH CONTAINING THE TEST ON THE DEGENERATE CASE, WITH GA(1,1 3) = GA(2,2)/20X, 36HAND ER(2)-ER(1) LESS THAN 2*GA(1,2).)	LMA2 612
		LMA2 613
304	FORMAT(132H-***ATTENTION*** THIS PROGRAM HAS PROCEEDED THROUGH 2THE BRANCH CONTAINING THE TEST ON THE DEGENERATE CASE, WITH GA(1,1 3) = GA(2,2)/20X, 39HAND ER(2)-ER(1) GREATER THAN 2*GA(1,2).)	LMA2 614
		LMA2 615
400	FORMAT (1H 26HCOMPLEX PARTIAL WIDTHS----//49X, 4HREAL, 9X, 9HIMAGIL 2NARY/)	LMA2 616
		LMA2 617
402	FORMAT (1H ,30X, 4HGAM(, 11, 3H,N), E17.6, E16.6)	LMA2 618
		LMA2 619
403	FORMAT (1H ,30X, 4HGAM(, 11, 3H,G), E17.6, E16.6)	LMA2 620
		LMA2 621
404	FORMAT (1H ,30X, 4HGAM(, 11, 3H,F), E17.6, E16.6)	LMA2 622
		LMA2 623
405	FORMAT (25H6RESONANCE PARAMETERS----//)	LMA2 624
		LMA2 625
406	FORMAT (1H ,30X, 2HK=11, 13X, 5HALPHA, 12X, 4HGFIS, 12X, 4HGCAP/ 243X E12.6, 2(4X, E12.6)// 48X, 4HBETA, 12X, 4HHFIS, 12X, 4HFCAP/ 343X E12.6, 2(4X, E12.6)//)	LMA2 626
		LMA2 627
450	FORMAT (1H)	LMA2 628
		LMA2 629
500	FORMAT (1H , F6.3, 3(F17.5,F14.5), F16.6, F14.6)	LMA2 630
		LMA2 631
501	FORMAT (1H1, 50X, 30H--UNBROADENED CROSS SECTIONS--//7H ENERGY, 1 20X, 5HTOTAL, 9X, 5HTOTAL, 11X, 7HFISSION, 7X, 7HFISSION, 10X, 7HCA 3PTURE, 7X, 7HCAPTURE, 8X, 10HSCATTERING, 4X, 10HSCATTERING/ 17X, 6 4H(LMA)* 7X, 8H(P.E.)* 10X, 5H(LMA), 9X, 6H(P.E.), 11X, 5H(LMA), 9X 5,6H(P.E.), 11X, 5H(LMA), 8X, 6H(P.E.))//)	LMA2 632
		LMA2 633
600	FORMAT (5F15.4)	LMA2 634
		LMA2 635
1164	FORMAT (4E15.8, 214, 3X, 9HALFA,BETA)	LMA2 636
		LMA2 637
1165	FORMAT (4E15.8, 214, 5X, 7HGFS,HFS)	LMA2 638
		LMA2 639
1166	FORMAT (4E15.8, 214, 5X, 7HGGA,HGA)	LMA2 640
		LMA2 641
3000	FORMAT (1H1, 55X,23HP R O G R A M L E M A//38X, 57HCALCULATION O 2F RESONANCE CROSS SECTIONS WITH LEVEL MATRIX/38X, 58HINVERSION AND 3 WITH COMPLEX EIGENVALUES OF THE LEVEL MATRIX//))	LMA2 642
		LMA2 643
3010	FORMAT (7H ATM = F7.2,10X, 5HWN = 1PE10.4,10X, 5HCC = 1PE10.4,10X, 27HEMIN = OPF6.3,10X,7HEMAX = F6.3, 8X, 7HDELE = F5.4//))	LMA2 644
		LMA2 645
3011	FORMAT (1H 49X, 11, 5E16.6)	LMA2 646
		LMA2 647
3012	FORMAT (1H 41X, 11, 6E15.6)	LMA2 648
		LMA2 649
3013	FORMAT (1H1, 43X, 47HP R O G R A M L E M A (C O N T I N U E O 2)//))	LMA2 650
		LMA2 651
3020	FORMAT (3H-J=11, 8X, 6HGFACT(11, 3H) =E10.4, 6X,3HNO(11,3H) =I2/) 2 10X, 2HGN/)	LMA2 652
		LMA2 653
3030	FORMAT (1H-, 49X, 1HK, 10X, 2HER, 9X, 3HGON, 10X, 2HGG, 10X, 2HGF, 2 10X, 2HGN/)	LMA2 654
		LMA2 655
3031	FORMAT (1H- 1X,59H* (LMA) --- EVALUATED FROM INVERSION OF WIGNER L 2EVEL MATRIX)	LMA2 656
		LMA2 657
3032	FORMAT (1H 90H** (P.E.) -- EVALUATED FROM POLES EXPANSION (WITH 2OUT FIRST ORDER PERTURBATION CORRECTIONS))	LMA2 658
		LMA2 659
3033	FORMAT (1H 87H** (P.E.) -- EVALUATED FROM POLES EXPANSION (WITH 2 FIRST ORDER PERTURBATION CORRECTIONS))	LMA2 660
		LMA2 661
	END	LMA2 662

LMA3	1
LMA3	2
LMA3	3
LMA3	4
LMA3	5
LMA3	6
LMA3	7
LMA3	8
LMA3	9
LMA3	10
LMA3	11
LMA3	12
LMA3	13
LMA3	14

LMA3	16
2LMA3	17
(LMA3	18
LMA3	19
)LMA3	20
(LMA3	21
LMA3	22
LMA3	23
CLMA3	24
LMA3	25

LMA3	26
LMA3	27
LMA3	28
LMA3	29
LMA3	30
LMA3	31
LMA3	32
LMA3	33
LMA3	34
LMA3	35
LMA3	36
LMA3	37
LMA3	38
LMA3	39
LMA3	40
LMA3	41
LMA3	42
LMA3	43
LMA3	44
LMA3	45
LMA3	46
LMA3	47
LMA3	48
LMA3	49
LMA3	50
LMA3	51
LMA3	52
LMA3	53
LMA3	54
LMA3	55
LMA3	56
LMA3	57
LMA3	58
LMA3	59
LMA3	60

54	IF (56 - LINES - NR) 97,98,98	LMA3	61
97	WRITE (6,89)	LMA3	62
	LINES = 5	LMA3	63
98	WRITE (6,77)	LMA3	64
	LINES = LINES + 4	LMA3	65
	DO 36 K=1,NR	LMA3	66
	READ(5,101) ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J)	LMA3	67
	LINES = LINES + 1	LMA3	68
36	WRITE (6,73) K,ER(K,J),GON(K,J),GG(K,J),GF(K,J),GN(K,J)	LMA3	69
	DO 37 K=1,NR	LMA3	70
37	READ(5,101) (GO(K,L,J), L=1,NR)	LMA3	71
	T1 = 5HG0(K,	LMA3	72
	T2 = 3H,J)	LMA3	73
	NROWS = 1 + (NR-1)/5	LMA3	74
	DO 1 NROW = 1,NROWS	LMA3	75
	NCOLS = 5 + NR - 5*NROW	LMA3	76
	IF (NCOLS - 5) 78,78,79	LMA3	77
79	NCOLS = 5	LMA3	78
78	K1 = 5*(NROW-1) + 1	LMA3	79
	K2 = 5*(NROW-1) + NCOLS	LMA3	80
	GO TO (80,81,81,81,81), NROW	LMA3	81
80	IF (56 - LINES - NR) 82,83,83	LMA3	82
32	WRITE (6,89)	LMA3	83
	LINES = 5	LMA3	84
83	WRITE (6,84) (T1,I,T2, I = K1,K2)	LMA3	85
	WRITE (6,85)	LMA3	86
	LINES = LINES + 4	LMA3	87
	GO TO 86	LMA3	88
81	IF (57 - LINES - NR) 87,88,88	LMA3	89
87	WRITE (6,89)	LMA3	90
	WRITE (6,84) (T1,I,T2, I = K1,K2)	LMA3	91
	WRITE (6,85)	LMA3	92
	LINES = 9	LMA3	93
	GO TO 86	LMA3	94
88	WRITE (6,90) (T1, I, T2, I = K1,K2)	LMA3	95
	WRITE (6,85)	LMA3	96
	LINES = LINES + 3	LMA3	97
86	DO 91 K = 1,NR	LMA3	98
	LINES = LINES + 1	LMA3	99
91	WRITE (6,92) K, (GO(K,L,J), L = K1,K2)	LMA3	100
1	CONTINUE	LMA3	101
C		LMA3	102
C	FORM OFF-DIAGONAL ELEMENTS OF THE LEVEL MATRIX	LMA3	103
C		LMA3	104
	J=1	LMA3	105
62	NR=NO(J)	LMA3	106
	IF(NR) 61,61,2	LMA3	107
2	CONTINUE	LMA3	108
	DO 4 L =1,NR	LMA3	109
	DO 4 LL=1,NR	LMA3	110
	Y=GF(L,J)*GF(LL,J)	LMA3	111
	YY=SQRT(Y)	LMA3	112
	GO TO (39,40), ISW	LMA3	113
39	COSINE=COS(TETA(L,J)-TETA(LL,J))	LMA3	114
	GO(L,LL,J)=YY*COSINE/2.0	LMA3	115
	GO TO 41	LMA3	116
40	GO(L,LL,J)=YY*GO(L,LL,J)/2.0	LMA3	117
41	IF(L-LL) 4,5,4	LMA3	118
	5 GO(L,L,J)=GO(L,L,J)+GG(L,J)/2.0	LMA3	119
	4 CONTINUE	LMA3	120
61	J=J+1	LMA3	121

IF(J-2) 62,62,63

C

CALCULATION OF EIGENVALUES IN SINGLE PRECISION

63	J=1	LMA3 122
163	NA=NO(J)	LMA3 123
	EAV=0.0	LMA3 124
	IF(NR) 165,165,164	LMA3 125
164	DJ 166 K=1,NR	LMA3 126
	DJ 166 KK=1,NR	LMA3 127
166	GA(K,KK)=GO(K,KK,J)	LMA3 128
	DJ 167 K=1,NR	LMA3 129
	S=SQRT(ER(K,J))	LMA3 130
167	GA(K,K)=GA(K,K)+GON(K,J)*S/(GFACT(J)*4.0)	LMA3 131
	DJ 168 K=1,NR	LMA3 132
	EAV=EAV+ER(K,J)	LMA3 133
	DJ 168 KK=1,NR	LMA3 134
168	D(K,KK)=0.0	LMA3 135
	EAV=EAV/FLOAT(NR)	LMA3 136
	DJ 169 K=1,NR	LMA3 137
169	D(K,K)=ER(K,J)	LMA3 138
	DJ 170 K=1,NR	LMA3 139
	DJ 170 KK=1,NR	LMA3 140
	XZ=-GA(K,KK)	LMA3 141
	A(K,KK)=CMPLX(D(K,KK),XZ)	LMA3 142
	FACT=CMPLX(EAV,0.0)	LMA3 143
170	A(K,KK)=A(K,KK)/FACT	LMA3 144
	N=NR	LMA3 145
	IA=25	LMA3 146
	CALL ALLMAT(A,EIGVAL,N,IA,NCAL)	LMA3 147
	IF(54 - LINES - NR) 502,503,503	LMA3 148
502	WRITE(6,89)	LMA3 149
	LINES = 5	LMA3 150
503	WRITE(6,500)	LMA3 151
	LINES = LINES + 6	LMA3 152
	DJ 172 K=1,NR	LMA3 153
	EIGVAL(K)=EIGVAL(K)*FACT	LMA3 154
	LINES = LINES + 1	LMA3 155
172	WRITE(6,501) K, EIGVAL(K)	LMA3 156
C		LMA3 157
C	OBTAIN EIGENVECTOR FOR K-TH EIGENVALUE	LMA3 158
C		LMA3 159
	K=1	LMA3 160
460	MK=0	LMA3 161
	L=1	LMA3 162
400	IF(L-K) 402,411,402	LMA3 163
411	IF(K-VR) 401,409,401	LMA3 164
401	L=L+1	LMA3 165
	GO TO 400	LMA3 166
402	MK=MK+1	LMA3 167
	NK=0	LMA3 168
	LL=1	LMA3 169
403	IF(LL-K) 405,414,405	LMA3 170
414	IF(K-VR) 404,408,404	LMA3 171
404	LL=LL+1	LMA3 172
	GO TO 403	LMA3 173
405	NK=NK+1	LMA3 174
	G(MK,NK)=GA(L,LL)	LMA3 175
	IF(MK-NK) 407,406,407	LMA3 176
406	G(MK,MK)=G(MK,MK)+AIMAG(EIGVAL(K))	LMA3 177
	D(MK,MK)=ER(L,J)-REAL(EIGVAL(K))	LMA3 178
		LMA3 179
		LMA3 180
		LMA3 181
		LMA3 182

WG(MK)=CMPLX(0.0,GA(L,K))	LMA3 183
407 IF(LL-NR) 404,408,408	LMA3 184
408 IF(L-NR) 401,409,409	LMA3 185
C SOLVE FOR VECTOR Y	LMA3 186
409 NMAX=NR-1	LMA3 187
DJ 410 LK=1,NMAX	LMA3 188
DJ 410 LLK=1,NMAX	LMA3 189
410 Q(LK,LLK) =G(LK,LLK)	LMA3 190
C INVERT Q	LMA3 191
DJ 421 I=2,NMAX	LMA3 192
421 B(I,1)=0.0	LMA3 193
B(1,1)=1.0	LMA3 194
CALL MATINV(Q,NMAX,B,0,DETERM)	LMA3 195
DJ 420 LK=1,NMAX	LMA3 196
DJ 420 LLK=1,NMAX	LMA3 197
420 OINV(LK,LLK)=G(LK,LLK) +Q(LK,LLK)*D(LK, LK)*D(LLK,LLK)	LMA3 198
C INVERT OINV	LMA3 199
DJ 422 I=2,NMAX	LMA3 200
422 B(I,1)=0.0	LMA3 201
B(1,1)=1.0	LMA3 202
CALL MATINV(OINV,NMAX,B,0,DETERM)	LMA3 203
DJ 440 LK=1,NMAX	LMA3 204
DJ 440 LLK=1,NMAX	LMA3 205
440 AINV(LK,LLK)=0.0	LMA3 206
DJ 442 LK=1,NMAX	LMA3 207
DJ 442 LLK=1,NMAX	LMA3 208
DJ 441 MK=1,NMAX	LMA3 209
441 AINV(LK,LLK)= AINV(LK,LLK)+ OINV(LK,MK)*D(MK,MK)*Q(MK,LLK)	LMA3 210
442 A(LK,LLK)=CMPLX(AINV(LK,LLK),OINV(LK,LLK))	LMA3 211
DJ 443 LK=1,NMAX	LMA3 212
443 WA(LK)=(0.0,0.0)	LMA3 213
DJ 444 LK=1,NMAX	LMA3 214
DJ 444 LLK=1,NMAX	LMA3 215
444 WA(LK) = WA(LK) +A(LK,LLK)*WG(LLK)	LMA3 216
FACT=(1.0,0.0)	LMA3 217
DJ 445 LK=1,NMAX	LMA3 218
445 FACT = FACT +WA(LK)*WA(LK)	LMA3 219
ZZ=(1.0,0.0)	LMA3 220
ZZZ=ZZ/FACT	LMA3 221
SCAVE(K,K) = CSQRT(ZZZ)	LMA3 222
LK=1	LMA3 223
L=1	LMA3 224
451 IF(L-K) 452,450,452	LMA3 225
450 L=L+1	LMA3 226
452 IF(L-NR) 453,453,454	LMA3 227
453 SCAVE(L,K)=SCAVE(K,K)*WA(LK)	LMA3 228
L=L+1	LMA3 229
LK=LK+1	LMA3 230
IF(LK-NMAX) 451,451,454	LMA3 231
454 K=K+1	LMA3 232
IF(K-NR) 460,460,461	LMA3 233
461 DJ 462 K=1,NR	LMA3 234
DJ 462 KK=1,NR	LMA3 235
462 A(K,KK)=SCAVE(K,KK)	LMA3 236
	LMA3 237
	LMA3 238
	LMA3 239
	LMA3 240
	LMA3 241
	LMA3 242
	LMA3 243

ALCULATE COMPLEX PARTIAL WIDTHS

	DJ 180 K=1,NR	LMA3 244
	WC(K)=(0.0,0.0)	LMA3 245
	WG(K)=(0.0,0.0)	LMA3 246
	180 WA(K)=(0.0,0.0)	LMA3 247
	IF (56 - LINES - NR) 506,507,507	LMA3 248
	506 WRITE (6,89)	LMA3 249
	LINES = 5	LMA3 250
	507 WRITE (6,504)	LMA3 251
	LINES = LINES + 6	LMA3 252
	DJ 182 K=1,NR	LMA3 253
	DJ 181 L=1,NR	LMA3 254
	WG(K) = WG(K) + SQRT(GON(L,J))*A(L,K)	LMA3 255
	WC(K) = WC(K) + SQRT(GG(L,J))*A(L,K)	LMA3 256
	181 WA(K) = WA(K) + SQRT(GF(L,J))*A(L,K)	LMA3 257
	LINES = LINES + 1	LMA3 258
	182 WRITE (6,505) K, WG(K), WA(K), WC(K)	LMA3 259
	DJ 183 K=1,NR	LMA3 260
	DJ 183 KK=1,NR	LMA3 261
	FISVE(K, KK)=(0.0, 0.0)	LMA3 262
	CAPVE(K, KK)=(0.0, 0.0)	LMA3 263
	183 SCAVE(K, KK)=(0.0, 0.0)	LMA3 264

ALCULATE KAPUR-PEIERLS TYPE PARAMETERS

C

	DJ 184 K=1,NR	LMA3 265
	DJ 184 KK=1,NR	LMA3 266
	DJ 186 L=1,NR	LMA3 267
	DJ 186 LL=1,NR	LMA3 268
	ZZ=CONJG(A(LL, KK))	LMA3 269
	IF(L-LL) 188, 187, 188	LMA3 270
	188 FISVE(K, KK) = FISVE(K, KK) + 2.0*GA(L, LL)*A(L, K)*ZZ	LMA3 271
	GJ TO 186	LMA3 272
	187 CAPVE(K, KK) = CAPVE(K, KK) + GG(L, J)*A(L, K)*ZZ	LMA3 273
	FISVE(K, KK) = FISVE(K, KK) + GF(L, J)*A(L, K)*ZZ	LMA3 274
	186 CONTINUE	LMA3 275
	ZZ=CONJG(WG(KK))	LMA3 276
	FIN(K, KK) = FISVE(K, KK)*WG(K)*ZZ	LMA3 277
	184 CAN(K, KK) = CAPVE(K, KK)*WG(K)*ZZ	LMA3 278
	DJ 185 K=1,NR	LMA3 279
	ZZ=WG(K)*WG(K)	LMA3 280
	ALFA(K)=REAL(ZZ)	LMA3 281
	BETA(K)=AIMAG(ZZ)	LMA3 282
	FU(K)=REAL(EIGVAL(K))	LMA3 283
	GU(K) = -AIMAG(EIGVAL(K))	LMA3 284
	GFS(K)=0.0	LMA3 285
	HFS(K)=0.0	LMA3 286
	GGA(K)=0.0	LMA3 287
	185 HGA(K)=0.0	LMA3 288
	DJ 189 K=1,NR	LMA3 289
	DJ 189 KK=1,NR	LMA3 290
	GUGU=GU(K)+GU(KK)	LMA3 291
	FUFU=FU(K)-FU(KK)	LMA3 292
	FUNU=GUGU**2+ FUFU**2	LMA3 293
	PHI=REAL(FIN(K, KK))	LMA3 294
	SIG=AIMAG(FIN(K, KK))	LMA3 295
	RHI=REAL(CAN(K, KK))	LMA3 296
	TIG=AIMAG(CAN(K, KK))	LMA3 297
	GFS(K) = GFS(K) + (GUGU*PHI + FUFU*SIG)/FUNU	LMA3 298
	HFS(K) = HFS(K) + (GUGU*SIG - FUFU*PHI)/FUNU	LMA3 299

G3A(K)=G3A(K)+(GUGU*RHI+FUFU*TIG)/FUNU

189 H3A(K)=H3A(K)+(GUGU*TIG-FUFU*RHI)/FUNU

LMA3 305

LMA3 306

LMA3 307

LMA3 308

C

200 IF (55 - LINES - NR) 509,510,510

LMA3 309

LMA3 310

509 WRITE (6,89)

LMA3 311

LINES = 5

LMA3 312

510 WRITE (6,508)

LMA3 313

LINES = LINES + 5

LMA3 314

DJ 190 K=1,NR

LMA3 315

DE(K,J)=FU(K)

LMA3 316

DW(K,J)=GU(K)

LMA3 317

BEG(K,J)=BETA(K)

LMA3 318

ALG(K,J)=ALFA(K)

LMA3 319

CAPG(K,J)=GGA(K)

LMA3 320

CAPH(K,J)=HGA(K)

LMA3 321

FISG(K,J)=GFS(K)

LMA3 322

FISH(K,J)=HFS(K)

LMA3 323

LINES = LINES + 1

LMA3 324

190 WRITE(6, 505) K,ALFA(K),GFS(K),GGA(K),BETA(K),HFS(K),HGA(K)

LMA3 325

165 J=J+1

LMA3 326

IF(J-2) 163,163,173

LMA3 327

C

LMA3 328

C

SINGLE LEVEL EIGENVALUES

LMA3 329

C

LMA3 330

173 IF(J-4) 174,174,175

LMA3 331

174 NR=NO(J)

LMA3 332

IF(NR) 165,165,176

LMA3 333

176 IF (54 - LINES - NR) 520,521,521

LMA3 334

520 WRITE (6,89)

LMA3 335

LINES = 5

LMA3 336

521 WRITE (6,500)

LMA3 337

LINES = LINES + 6

LMA3 338

DJ 178 K=1,NR

LMA3 339

S= SQRT(ER(K,J))

LMA3 340

GA(K,K)=GF(K,J) +GON(K,J)*S/(GFACT(J)*4.0) +GG(K,J)

LMA3 341

XZ=-GA(K,K)

LMA3 342

EIGVAL(K)=CMPLX(ER(K,J),XZ)

LMA3 343

LINES = LINES + 1

LMA3 344

178 WRITE (6,501) K, EIGVAL(K)

LMA3 345

C

LMA3 346

C

SINGLE LEVEL PARAMETERS

LMA3 347

C

LMA3 348

DJ 191 K=1,NR

LMA3 349

FU(K)=ER(K,J)

LMA3 350

GU(K)=GA(K,K)

LMA3 351

BETA(K)=0.0

LMA3 352

HGA(K)=0.0

LMA3 353

HFS(K)=0.0

LMA3 354

G3A(K)= GON(K,J)*GG(K,J)/(2.0*GU(K))

LMA3 355

GFS(K)=GON(K,J)*GF(K,J)/(2.0*GU(K))

LMA3 356

191 ALFA(K)= GON(K,J)

LMA3 357

GJ TO 200

LMA3 358

C

LMA3 359

C

BEGIN ENERGY DEPENDENT CALCULATION OF LEVEL MATRIX

LMA3 360

175 WRITE (6,511)

LMA3 361

LINES = 8

LMA3 362

C

LMA3 363

3 E=EMIN

LMA3 364

30 S=SQRT(E)

LMA3 365

DJ 60 J=1,4	LMA3 366
6 NR=NO(J)	LMA3 367
IF(NR) 60,60,55	LMA3 368
55 DJ 17 L=1,NR	LMA3 369
VRE(L,J)=0.0	LMA3 370
17 VIM(L,J)=0.0	LMA3 371
Q1(J)=0.0	LMA3 372
Q2(J)=0.0	LMA3 373
GJ TO (57,57,58,58),J	LMA3 374
57 DJ 7 L=1,NR	LMA3 375
G(L,L)=G0(L,L,J)+S*GON(L,J)/(4.0*GFACT(J))	LMA3 376
LMAX=L-1	LMA3 377
IF(LMAX) 7,7,8	LMA3 378
8 DJ 10 LL=1,LMAX	LMA3 379
G(L,LL)=G0(L,LL,J)	LMA3 380
10 G(LL,L)=G(L,LL)	LMA3 381
7 CONTINUE	LMA3 382
DJ 111 K=2,NR	LMA3 383
111 B(K,1)=0.0	LMA3 384
B(1,1)=1.0	LMA3 385
DJ 11 L=1,NR	LMA3 386
DJ 11 LL=1,NR	LMA3 387
11 AINV(L,LL)=G(L,LL)	LMA3 388
CALL MATINV(AINV,NR,B,0,DETERM)	LMA3 389
DJ 12 L=1,NR	LMA3 390
DJ 12 LL=1,NR	LMA3 391
Q(L,LL)=AINV(L,LL)	LMA3 392
12 OINV(L,LL)=G(L,LL)+Q(L,LL)*(ER(L,J)-E)*(ER(LL,J)-E)	LMA3 393
DJ 112 K=2,NR	LMA3 394
112 B(K,1)=0.0	LMA3 395
B(1,1)=1.0	LMA3 396
DJ 120 L=1,NR	LMA3 397
DJ 120 LL=1,NR	LMA3 398
120 AINV(L,LL)=OINV(L,LL)	LMA3 399
CALL MATINV(AINV,NR,B,0,DETERM)	LMA3 400
DJ 121 L=1,NR	LMA3 401
DJ 121 LL=1,NR	LMA3 402
121 OINV(L,LL)=AINV(L,LL)	LMA3 403
DJ 14 L=1,NR	LMA3 404
DJ 14 LL=1,NR	LMA3 405
14 D(L,LL)=0.0	LMA3 406
DJ 15 L=1,NR	LMA3 407
DJ 15 LL=1,NR	LMA3 408
DJ 15 M=1,NR	LMA3 409
15 D(L,LL)=D(L,LL)+OINV(L,M)*(ER(M,J)-E)*Q(M,LL)	LMA3 410
DJ 20 L=1,NR	LMA3 411
DJ 20 LL=1,NR	LMA3 412
Y=SQRT(GJN(LL,J))	LMA3 413
VRE(L,J)=VRE(L,J)+D(L,LL)*Y	LMA3 414
20 VIM(L,J)=VIM(L,J)+OINV(L,LL)*Y	LMA3 415
DJ 21 L=1,NR	LMA3 416
Y=SQRT(GJN(L,J))	LMA3 417
Q1(J)=Q1(J)+VRE(L,J)*Y	LMA3 418
21 Q2(J)=Q2(J)+VIM(L,J)*Y	LMA3 419
GJ TO 60	LMA3 420
58 DJ 59 L=1,NR	LMA3 421
GAM=GS(L,J)+GF(L,J)+GON(L,J)*S/(4.0*GFACT(J))	LMA3 422
W=(ER(L,J)-E)**2+GAM*GAM	LMA3 423
Y=SQRT(GJN(L,J))	LMA3 424
VRE(L,J)=Y*(ER(L,J)-E)/W	LMA3 425
Q1(J)=Q1(J)+Y*VRE(L,J)	LMA3 426

VIM(L,J)=Y*GAM/W	LMA3 427
59 Q2(J)=Q2(J)+Y*VIM(L,J)	LMA3 428
60 CONTINUE	LMA3 429
C CALCULATE UNBROADENED CROSS SECTIONS	LMA3 430
C	LMA3 431
19 SIGG=0.0	LMA3 432
SIGA=0.0	LMA3 433
SIGS=0.0	LMA3 434
ARG=2.0*WN*S	LMA3 435
SIGT=(1.0-COS(ARG))/S*2.0	LMA3 436
DJ 22 J=1,4	LMA3 437
NR=NO(J)	LMA3 438
IF(NR) 22,22,66	LMA3 439
66 DJ 23 L=1,NR	LMA3 440
SIGG=SIGG+GG(L,J)*(VRE(L,J)**2+VIM(L,J)**2)	LMA3 441
GN(L,J)=GDN(L,J)*S/(2.0*GFACT(J))	LMA3 442
23 SIGS=SIGS+GN(L,J)*(VRE(L,J)**2+VIM(L,J)**2)	LMA3 443
SIGA=SIGA+Q2(J)	LMA3 444
SIGT=SIGT-Q1(J)*SIN(ARG)+Q2(J)*COS(ARG)	LMA3 445
22 CONTINUE	LMA3 446
TJT=SIGT*CC/S	LMA3 447
CAP=SIGG*CC/(2.0*S)	LMA3 448
SIGS=SIGS*CC/(2.0*S)	LMA3 449
FIS=SIGA*CC/S-SIGS-CAP	LMA3 450
SCAT=TJT-FIS-CAP	LMA3 451
PUNCHES DATA ON CARDS IF INDEX IPUNCH IS A POSITIVE INTEGER	LMA3 452
IF(IPUNCH) 1100,1100,1101	LMA3 453
1101 PUNCH 522, E, TOT, FIS, CAP, SCAT	LMA3 454
1100 TJT2=TJT	LMA3 455
FIS2=FIS	LMA3 456
CAP2=CAP	LMA3 457
SCAT2=SCAT	LMA3 458
J=1	LMA3 459
TJT=0.0	LMA3 460
CAP=0.0	LMA3 461
FIS=0.0	LMA3 462
201 NR=NO(J)	LMA3 463
IF(NR) 205,205,202	LMA3 464
202 DJ 203 K=1,NR	LMA3 465
FU(K)=DE(K,J)	LMA3 466
GU(K)=DW(K,J)	LMA3 467
ALFA(K)=ALG(K,J)	LMA3 468
BETA(K)=BEG(K,J)	LMA3 469
GFS(K)=FISG(K,J)	LMA3 470
HFS(K)=FISH(K,J)	LMA3 471
GGA(K)=CAPG(K,J)	LMA3 472
203 HGA(K)=CAPH(K,J)	LMA3 473
CALCULATE CROSS SECTIONS FROM PARAMETERS	LMA3 474
C	LMA3 475
S=SQRT(E)	LMA3 476
ARG=2.0*WN*S	LMA3 477
CJ=COS(ARG)	LMA3 478
SI=SIN(ARG)	LMA3 479
PART1=0.0	LMA3 480
PART2=0.0	LMA3 481
DJ 204 K=1,NR	LMA3 482
X=FU(K)-E	LMA3 483
	LMA3 484
	LMA3 485
	LMA3 486
	LMA3 487

	BW=X*X+GJ(K)*GU(K)	LMA3 488
	SEV=GU(K)/BW	LMA3 489
	SDD=X/BW	LMA3 490
	PART1=PART1+ SDD*ALFA(K)-SEV*BETA(K)	LMA3 491
	PART2=PART2+ SDD*BETA(K)+SEV*ALFA(K)	LMA3 492
	FIS=FIS+SEV*GFS(K)+SDD*HFS(K)	LMA3 493
204	CAP =CAP+SEV*GGA(K)+SDD*HGA(K)	LMA3 494
	TOT=TOT+CO*PART2-SI*PART1	LMA3 495
205	J=J+1	LMA3 496
	IF(J-4) 201,201,206	LMA3 497
206	CAP=CAP*CC/S	LMA3 498
	FIS=FIS*CC/S	LMA3 499
	TOT=TOT*CC/S + 2.0*CC*(1.0-CO)/E	LMA3 500
	SCAT=TOT - FIS - CAP	LMA3 501
	WRITE(6, 517) E,TOT2,TOT,FIS2,FIS,CAP2,CAP,SCAT2,SCAT	LMA3 502
	LINES=LINES+1	LMA3 503
	IF (LINES - 56) 207,512,512	LMA3 504
512	WRITE (6,513)	LMA3 505
	WRITE (6,511)	LMA3 506
	LINES = 8	LMA3 507
207	E=E+DELE	LMA3 508
	IF(E-EMAX) 30,30,24	LMA3 509
24	IF (LINES - 8) 1000,1000,514	LMA3 510
514	DO 515 K = LINES, 56	LMA3 511
515	WRITE (6,85)	LMA3 512
	WRITE (6,513)	LMA3 513
	GO TO 1000	LMA3 514
C		LMA3 515
70	FJRMAT (1H1, 55X,23HP R O G R A M L E M A//38X, 57HCALCULATION DLM	LMA3 516
	2F RESONANCE CROSS SECTIONS WITH LEVEL MATRIX/38X, 58HINVERSION ANDL	LMA3 517
	3 WITH COMPLEX EIGENVALUES OF THE LEVEL MATRIX///)	LMA3 518
71	FJRMAT (7H ATM = F7.2,10X, 5HWN = 1PE10.4,10X, 5HCC = 1PE10.4,10X,LMA3	LMA3 519
	27HEMIN = OPF6.3,10X,7HEMAX = F6.3, 8X, 7HDELE = F5.4///)	LMA3 520
72	FJRMAT (3H-J=11, 8X, 6HGFACT(11, 3H) =E10.4, 6X,3HNO(11,3H) =12/) LMA3	LMA3 521
73	FJRMAT (1H 48X, 12, 5E16.6)	LMA3 522
74	FJRMAT (1H-, 49X, 1HK, 10X, 2HER, 9X, 3HGON, 10X, 2HGG, 10X, 2HGF,LMA3	LMA3 523
	2 10X, 2HGN/)	LMA3 524
75	FJRMAT (1H-, 29X, 5HISW=1, 7X, 1HK, 8X, 2HER, 13X, 3HGON, 12X,	LMA3 525
	2 2HGG, 13X, 2HGF, 13X, 2HGN, 12X, 4HTETA/)	LMA3 526
76	FJRMAT (1H 40X, 12, 6E15.6)	LMA3 527
77	FJRMAT (1H-, 29X, 5HISW=2, 15X, 1HK, 9X, 2HER, 14X, 3HGON, 13X,	LMA3 528
	2 2HGG, 14X, 2HGF, 14X, 2HGN/)	LMA3 529
84	FJRMAT (1H-,29X,16HMATRIX GO(K,L,J),4X,1HK,5(6X,A5,12,A3))	LMA3 530
85	FJRMAT (1H)	LMA3 531
89	FJRMAT (1H1, 43X, 47HP R O G R A M L E M A (C O N T I N U E DLM	LMA3 532
	2)///)	LMA3 533
90	FJRMAT (1H0, 50X, 5(6X, A5, 12, A3))	LMA3 534
92	FJRMAT (1H 48X, 12, 5E16.6)	LMA3 535
100	FJRMAT(6E12.3)	LMA3 536
101	FJRMAT(5E12.3)	LMA3 537
102	FJRMAT (215)	LMA3 538
103	FJRMAT(E12.3,112)	LMA3 539
500	FJRMAT (36H-EIGENVALUES OF THE LEVEL MATRIX----//32X, 1HK 16X, 2HMLMA3	LMA3 540
	2U 14X, 24NU/)	LMA3 541
501	FJRMAT (31X, 12, F21.5, F15.5)	LMA3 542
504	FJRMAT (27H-COMPLEX PARTIAL WIDTHS----/48X, 8HGAM(K,N) 25X, 8HGAM(LMA3	LMA3 543
	2K,F) 25X, 8HGAM(K,G)/32X, 1HK 3(11X, 4HREAL 8X, 9HIMAGINARY 1X)/)	LMA3 544
505	FJRMAT (31X, 12, 3(6X, E12.6, 3X, E12.6))	LMA3 545
508	FJRMAT (25H-RESONANCE PARAMETERS----/32X, 1HK 10X, 5HALPHA 11X, 4HLM	LMA3 546
	2BETA 14X, 4HGFIS 11X, 4HHEIS 14X, 4HGCAP 11X, 4HHCAP/)	LMA3 547
511	FJRMAT (1H1, 50X, 30H--UNBROADENED CROSS SECTIONS--//7H ENERGY, 1LM	LMA3 548

20X, 5HTOTAL, 9X, 5HTOTAL, 11X, 7HFISSION, 7X, 7HFISSION, 10X, 7HCALMA3 549
3PTURE, 7X, 7HCAPTURE, 8X, 10HSCATTERING, 4X, 10HSCATTERING/ 17X, 6LMA3 550
4H(LMA)* 7X, 8H(P.E.)* 10X, 5H(LMA), 9X, 6H(P.E.), 11X, 5H(LMA), 9XLMA3 551
5, 5H(P.E.), 11X, 5H(LMA), 8X, 6H(P.E.)/) LMA3 552
513 FJRMAT (1H- 1X, 59H* (LMA) --- EVALUATED FROM INVERSION OF WIGNER LLMA3 553
2EVEL MATRIX/44H ** (P.E.) -- EVALUATED FROM POLES EXPANSION) LMA3 554
517 FJRMAT (1H , F6.3, 3(F17.5, F14.5), F16.6, F14.6) LMA3 555
522 FJRMAT (5E15.8) LMA3 556
END LMA3 557

C		AMAT	1
	SUBROUTINE ALLMAT(A,LAMBDA,M,IA,NCAL)	AMAT	2
		AMAT	3
C	PROG.AUTHORS JOHN RINZEL,R.E.FUNDERLIC,UNION CARBIDE CORP.	AMAT	4
C	NUCLEAR DIVISION,CENTRAL DATA PROCESSING FACILITY,	AMAT	5
C	OAK RIDGE TENNESSEE	AMAT	6
C		AMAT	7
	COMPLEX A(IA,25),H(30,30),HL(30,30),LAMBDA(25),VECT(30),	AMAT	8
	IMJLT(30),SHIFT(3),TEMP,SIN,COS,TEMP1,TEMP2	AMAT	9
	LOGICAL INTH(30),TWICE	AMAT	10
	INTEGER INT(30),R,RP1,RP2	AMAT	11
	N=M	AMAT	12
	NCAL=N	AMAT	13
	IF(N.NE.1)GO TO 1	AMAT	14
	LAMBDA(1)=A(1,1)	AMAT	15
	A(1,1)=1.	AMAT	16
	GO TO 57	AMAT	17
1	ICOUNT=0	AMAT	18
	SHIFT=0.	AMAT	19
	IF(N.NE.2)GO TO 4	AMAT	20
2	TEMP=(A(1,1)+A(2,2)+CSQRT((A(1,1)+A(2,2))**2-	AMAT	21
	14.*(A(2,2)*A(1,1)-A(2,1)*A(1,2)))/2.	AMAT	22
	IF(REAL(TEMP).NE.0..OR.AIMAG(TEMP).NE.0.)GO TO 3	AMAT	23
	LAMBDA(M)=SHIFT	AMAT	24
	LAMBDA(M-1)=A(1,1)+A(2,2)+SHIFT	AMAT	25
	GO TO 37	AMAT	26
3	LAMBDA(M)=TEMP+SHIFT	AMAT	27
	LAMBDA(M-1)=(A(2,2)*A(1,1)-A(2,1)*A(1,2))/(LAMBDA(M)-SHIFT)+SHIFT	AMAT	28
	GO TO 37	AMAT	29
C		AMAT	30
C	REDUCE MATRIX A TO HESSENBERG FORM	AMAT	31
C		AMAT	32
4	NM2=N-2	AMAT	33
	DO 15 R=1,NM2	AMAT	34
	RP1=R+1	AMAT	35
	RP2=R+2	AMAT	36
	ABIG=0.	AMAT	37
	INT(R)=RP1	AMAT	38
	DO 5 I=RP1,N	AMAT	39
	ABSSQ=REAL(A(I,R))**2+AIMAG(A(I,R))**2	AMAT	40
	IF(ABSSQ.LE.ABIG)GO TO 5	AMAT	41
	INT(R)=I	AMAT	42
	ABIG=ABSSQ	AMAT	43
5	CONTINUE	AMAT	44
	INTER=INT(R)	AMAT	45
	IF(INTER.EQ.RP1)GO TO 8	AMAT	46
	IF(ABIG.EQ.0.)GO TO 15	AMAT	47
	DO 6 I=R,N	AMAT	48
	TEMP=A(RP1,I)	AMAT	49
	A(RP1,I)=A(INTER,I)	AMAT	50
6	A(INTER,I)=TEMP	AMAT	51
	DO 7 I=1,N	AMAT	52
	TEMP=A(I,RP1)	AMAT	53
	A(I,RP1)=A(I,INTER)	AMAT	54
7	A(I,INTER)=TEMP	AMAT	55
8	DO 9 I=RP2,N	AMAT	56
	MULT(I)=A(I,R)/A(RP1,R)	AMAT	57
9	A(I,R)=MULT(I)	AMAT	58
	DO 11 I=1,RP1	AMAT	59
	TEMP=0.	AMAT	60

DJ 10 J=RP2,N	AMAT 61
10 TEMP=TEMP+A(I,J)*MULT(J)	AMAT 62
11 A(I,RP1)=A(I,RP1)+TEMP	AMAT 63
DJ 13 I=RP2,N	AMAT 64
TEMP=0.	AMAT 65
DJ 12 J=RP2,N	AMAT 66
12 TEMP=TEMP+A(I,J)*MULT(J)	AMAT 67
13 A(I,RP1)=A(I,RP1)+TEMP-MULT(I)*A(RP1,RP1)	AMAT 68
DJ 14 I=RP2,N	AMAT 69
DJ 14 J=RP2,N	AMAT 70
14 A(I,J)=A(I,J)-MULT(I)*A(RP1,J)	AMAT 71
15 CONTINUE	AMAT 72
C	AMAT 73
C CALCULATE EPSILON	AMAT 74
C	AMAT 75
EPS=0.	AMAT 76
DJ 16 I=1,N	AMAT 77
16 EPS=EPS+CABS(A(1,I))	AMAT 78
DJ 18 I=2,N	AMAT 79
SUM=0.	AMAT 80
IM1=I-1	AMAT 81
DJ 17 J=IM1,N	AMAT 82
17 SUM=SUM+CABS(A(I,J))	AMAT 83
18 IF(SUM.GT.EPS)EPS=SUM	AMAT 84
EPS=SQRT(FLOAT(N))*EPS*1.E-12	AMAT 85
IF(EPS.EQ.0.)EPS=1.E-12	AMAT 86
DJ 19 I=1,N	AMAT 87
DJ 19 J=1,N	AMAT 88
19 H(I,J)=A(I,J)	AMAT 89
20 IF(N.NE.1)GO TO 21	AMAT 90
LAMBDA(M)=A(1,1)+SHIFT	AMAT 91
GJ TO 37	AMAT 92
21 IF(N.EQ.2)GO TO 2	AMAT 93
22 MVL=M-N+1	AMAT 94
IF(REAL(A(N,N)).NE.0..OR.AIMAG(A(N,N)).NE.0.)	AMAT 95
1 IF(ABS(REAL(A(N,N-1)/A(N,N)))+ABS(AIMAG(A(N,N-1)/A(N,N)))-1.E-9)	AMAT 96
2 24,24,23	AMAT 97
23 IF(ABS(REAL(A(N,N-1)))+ABS(AIMAG(A(N,N-1))).GE.EPS)GO TO 25	AMAT 98
24 LAMBDA(MVL)=A(N,N)+SHIFT	AMAT 99
ICOUNT=0	AMAT 100
N=N-1	AMAT 101
GJ TO 21	AMAT 102
C	AMAT 103
C DETERMINE SHIFT	AMAT 104
C	AMAT 105
25 SHIFT(2)=(A(N-1,N-1)+A(N,N)+CSQRT((A(N-1,N-1)+A(N,N))**2	AMAT 106
1 -4.*(A(N,N)*A(N-1,N-1)-A(N,N-1)*A(N-1,N)))/2.	AMAT 107
IF(REAL(SHIFT(2)).NE.0..OR.AIMAG(SHIFT(2)).NE.0.)GO TO 26	AMAT 108
SHIFT(3)=A(N-1,N-1)+A(N,N)	AMAT 109
GJ TO 27	AMAT 110
26 SHIFT(3)=(A(N,N)*A(N-1,N-1)-A(N,N-1)*A(N-1,N))/SHIFT(2)	AMAT 111
27 IF(CABS(SHIFT(2)-A(N,N)).LT.CABS(SHIFT(3)-A(N,N)))GO TO 28	AMAT 112
INDEX=3	AMAT 113
GJ TO 29	AMAT 114
28 INDEX=2	AMAT 115
29 IF(CABS(A(N-1,N-2)).GE.EPS)GO TO 30	AMAT 116
LAMBDA(MVL)=SHIFT(2)+SHIFT	AMAT 117
LAMBDA(MVL+1)=SHIFT(3)+SHIFT	AMAT 118
ICOUNT=0	AMAT 119
N=N-2	AMAT 120
GJ TO 20	AMAT 121

30	SHIFT=SHIFT+SHIFT(INDEX)	AMAT 122
	DJ 31 I=1,N	AMAT 123
31	A(I,I)=A(I,I)-SHIFT(INDEX)	AMAT 124
		AMAT 125
C	PERFORM GIVEN ROTATIONS, QR ITERATES	AMAT 126
		AMAT 127
	IF(ICOUNT.LE.10)GO TO 32	AMAT 128
	NCAL=M-N	AMAT 129
	GO TO 37	AMAT 130
32	NM1=N-1	AMAT 131
	TEMP1=A(1,1)	AMAT 132
	TEMP2=A(2,1)	AMAT 133
	DJ 36 R=1,NM1	AMAT 134
	RP1=R+1	AMAT 135
	R40=SQRT(REAL(TEMP1)**2+AIMAG(TEMP1)**2+	AMAT 136
1	REAL(TEMP2)**2+AIMAG(TEMP2)**2)	AMAT 137
	IF(RH0.EQ.0.)GO TO 36	AMAT 138
	COS=TEMP1/RH0	AMAT 139
	SIN=TEMP2/RH0	AMAT 140
	INDEX=MAX0(R-1,1)	AMAT 141
	DJ 33 I=INDEX,N	AMAT 142
	TEMP=CONJG(COS)*A(R,I)+CONJG(SIN)*A(RP1,I)	AMAT 143
	A(RP1,I)=-SIN*A(R,I)+COS*A(RP1,I)	AMAT 144
33	A(R,I)=TEMP	AMAT 145
	TEMP1=A(RP1,RP1)	AMAT 146
	TEMP2=A(R+2,R+1)	AMAT 147
	DJ 34 I=1,R	AMAT 148
	TEMP=COS*A(I,R)+SIN*A(I,RP1)	AMAT 149
	A(I,RP1)=-CONJG(SIN)*A(I,R)+CONJG(COS)*A(I,RP1)	AMAT 150
34	A(I,R)=TEMP	AMAT 151
	INDEX=MIN0(R+2,N)	AMAT 152
	DJ 35 I=RP1,INDEX	AMAT 153
	A(I,R)=SIN*A(I,RP1)	AMAT 154
35	A(I,RP1)=CONJG(COS)*A(I,RP1)	AMAT 155
36	CONTINUE	AMAT 156
	ICOUNT=ICOUNT+1	AMAT 157
	GO TO 22	AMAT 158
		AMAT 159
C	CALCULATE VECTORS	AMAT 160
		AMAT 161
37	IF(NCAL.EQ.0)GO TO 57	AMAT 162
	N=M	AMAT 163
	NM1=N-1	AMAT 164
	IF(N.LE.2)GO TO 38	AMAT 165
	EPS=AMAX1(CABS(LAMBDA(1)),CABS(LAMBDA(2)))*1.E-8	AMAT 166
	IF(EPS.EQ.0.)EPS=1.E-12	AMAT 167
	H(1,1)=A(1,1)	AMAT 168
	H(1,2)=A(1,2)	AMAT 169
	H(2,1)=A(2,1)	AMAT 170
	H(2,2)=A(2,2)	AMAT 171
38	DJ 56 L=1,NCAL	AMAT 172
	DJ 40 I=1,N	AMAT 173
	DJ 39 J=1,N	AMAT 174
39	HL(I,J)=1(I,J)	AMAT 175
40	HL(I,I)=4L(I,I)-LAMBDA(L)	AMAT 176
	DJ 44 I=1,NM1	AMAT 177
	MULT(I)=0.	AMAT 178
	INTH(I)=.FALSE.	AMAT 179
	IP1=I+1	AMAT 180
	IF(CABS(HL(IP1,I)).LE.CABS(HL(I,I)))GO TO 42	AMAT 181
	INTH(I)=.TRUE.	AMAT 182

DJ 41 J=I,N	AMAT 183
TEMP=HL(I+1,J)	AMAT 184
HL(I+1,J)=HL(I,J)	AMAT 185
41 HL(I,J)=TEMP	AMAT 186
42 IF(REAL(-HL(I,I)).EQ.0..AND.AIMAG(HL(I,I)).EQ.0.)GO TO 44	AMAT 187
MULT(I)=-HL(I+1,I)/HL(I,I)	AMAT 188
DJ 43 J=I+1,N	AMAT 189
43 HL(I+1,J)=HL(I+1,J)+MULT(I)*HL(I,J)	AMAT 190
44 CONTINUE	AMAT 191
DJ 45 I=1,N	AMAT 192
45 VECT(I)=1.	AMAT 193
TWICE=.FALSE.	AMAT 194
46 IF(REAL(-HL(N,N)).EQ.0..AND.AIMAG(HL(N,N)).EQ.0.)HL(N,N)=EPS	AMAT 195
VECT(N)=VECT(N)/HL(N,N)	AMAT 196
DJ 48 I=1,NM1	AMAT 197
K=N-I	AMAT 198
DJ 47 J=K,NM1	AMAT 199
47 VECT(K)=VECT(K)-HL(K,J+1)*VECT(J+1)	AMAT 200
IF(REAL(-HL(K,K)).EQ.0..AND.AIMAG(HL(K,K)).EQ.0.)HL(K,K)=EPS	AMAT 201
48 VECT(K)=VECT(K)/HL(K,K)	AMAT 202
BIG=0.	AMAT 203
DJ 49 I=1,N	AMAT 204
SUM=ABS(REAL(VECT(I)))+ABS(AIMAG(VECT(I)))	AMAT 205
49 IF(SUM.GT.BIG)BIG=SUM	AMAT 206
DJ 50 I=1,N	AMAT 207
50 VECT(I)=VECT(I)/BIG	AMAT 208
IF(TWICE)GO TO 52	AMAT 209
DJ 51 I=1,NM1	AMAT 210
IF(.NOT.INTH(I))GO TO 51	AMAT 211
TEMP=VECT(I)	AMAT 212
VECT(I)=VECT(I+1)	AMAT 213
VECT(I+1)=TEMP	AMAT 214
51 VECT(I+1)=VECT(I+1)+MULT(I)*VECT(I)	AMAT 215
TWICE=.TRUE.	AMAT 216
GO TO 46	AMAT 217
52 IF(N.EQ.2)GO TO 55	AMAT 218
NM2=N-2	AMAT 219
DJ 54 I=1,NM2	AMAT 220
N1I=N-1-I	AMAT 221
N1I=N-I+1	AMAT 222
DJ 53 J=N1I,N	AMAT 223
53 VECT(J)=1(J,N1I)*VECT(N1I+1)+VECT(J)	AMAT 224
INDEX=INT(N1I)	AMAT 225
TEMP=VECT(N1I+1)	AMAT 226
VECT(N1I+1)=VECT(INDEX)	AMAT 227
54 VECT(INDEX)=TEMP	AMAT 228
55 DJ 56 I=1,N	AMAT 229
56 A(I,L)=VECT(I)	AMAT 230
57 RETURN	AMAT 231
END	AMAT 232

C

MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF LINEAR EQUATIONS

UOI INV1
INV1 001

SUBROUTINE MATINV(A,N,B,M,DETERM)

INV1 002
INV1 003

C

DIMENSION IPIVOT(25), A(25,25), B(25,1), INDEX(25,2), PIVOT(25)
EQUIVALENCE (IROW,JROW), (ICOLUMN,JCOLUMN), (AMAX, T, SWAP)INV1 004
INV1 005
INV1 006

INITIALIZATION

INV1 007
INV1 008

DETERM=1.0

INV1 009

DO 20 J=1,N

INV1 010

20 IPIVOT(J)=0

INV1 011

DO 550 I=1,N

INV1 012

C

SEARCH FOR PIVOT ELEMENT

INV1 013

AMAX=0.0

INV1 014
INV1 015
INV1 016

DO 105 J=1,N

INV1 017

IF (IPIVOT(J)-1) 60, 105, 60

INV1 018

60 DO 100 K=1,N

INV1 019

IF (IPIVOT(K)-1) 80, 100, 740

INV1 020

80 IF (ABS (AMAX)-ABS (A(J,K))) 85, 85, 100

INV1 021

85 IROW=J

INV1 022

ICOLUMN=K

INV1 023

AMAX=A(J,K)

INV1 024

100 CONTINUE

INV1 025

105 CONTINUE

INV1 026

IPIVOT(ICOLUMN)=IPIVOT(ICOLUMN)+1

INV1 027

C

INTERCHANGE ROWS TO PUT PIVOT ELEMENT ON DIAGONAL

INV1 028
INV1 029
INV1 030

C

IF (IROW-ICOLUMN) 140, 260, 140

INV1 031

140 DETERM=-DETERM

INV1 032

DO 200 L=1,N

INV1 033

SWAP=A(IROW,L)

INV1 034

A(IROW,L)=A(ICOLUMN,L)

INV1 035

200 A(ICOLUMN,L)=SWAP

INV1 036

IF(M) 260, 260, 210

INV1 037

210 DO 250 L=1, M

INV1 038

SWAP=B(IROW,L)

INV1 039

B(IROW,L)=B(ICOLUMN,L)

INV1 040

250 B(ICOLUMN,L)=SWAP

INV1 041

260 INDEX(1,1)=IROW

INV1 042

INDEX(1,2)=ICOLUMN

INV1 043

PIVOT(1)=A(ICOLUMN,ICOLUMN)

INV1 044

DETERM=DETERM*PIVOT(1)

INV1 045

IF(PIVOT(1)) 330,720,330

INV1 046

C

DIVIDE PIVOT ROW BY PIVOT ELEMENT

INV1 047
INV1 048

C

330 A(ICOLUMN,ICOLUMN)=1.0

INV1 049
INV1 050

DO 350 L=1,N

INV1 051

350 A(ICOLUMN,L)=A(ICOLUMN,L)/PIVOT(1)

INV1 052

IF(M) 380, 380, 360

INV1 053

360 DO 370 L=1,M

INV1 054

370 B(ICOLUMN,L)=B(ICOLUMN,L)/PIVOT(1)

INV1 055

C

REDUCE NON-PIVOT ROWS

INV1 056
INV1 057

C

		INV1 058
380	DJ 550 L1=1,N	INV1 059
	IF(L1-ICJLUM) 400, 550, 400	INV1 060
400	T=A(L1,ICJLUM)	INV1 061
	A(L1,ICJLUM)=0.0	INV1 062
	DJ 450 L=1,N	INV1 063
450	A(L1,L)=A(L1,L)-A(ICJLUM,L)*T	INV1 064
	IF(M) 550, 550, 460	INV1 065
460	DJ 500 L=1,M	INV1 066
500	B(L1,L)=3(L1,L)-B(ICJLUM,L)*T	INV1 067
550	CJNTINUE	INV1 068
C	INTERCHANGE COLUMNS	INV1 070
C		INV1 069
C		INV1 071
	DJ 710 I=1,N	INV1 072
	L=N+1-I	INV1 073
	IF (INDEX(L,1)-INDEX(L,2)) 630, 710, 630	INV1 074
630	JROW=INDEX(L,1)	INV1 075
	JCOLUM=INDEX(L,2)	INV1 076
	DJ 705 K=1,N	INV1 077
	SWAP=A(K,JROW)	INV1 078
	A(K,JROW)=A(K,JCOLUM)	INV1 079
	A(K,JCOLUM)=SWAP	INV1 080
705	CJNTINUE	INV1 081
710	CJNTINUE	INV1 082
	RETURN	INV1 083
720	WRITE(6,730)	INV1 084
730	FORMAT(20H MATRIX IS SINGULAR)	INV1 085
740	RETURN	INV1 086

ACKNOWLEDGMENT

The suggestion of Mr. Funderlic of ORGDP concerning the use of his subroutine ALLMAT has been appreciated.

We are indebted to the Computer Center of Purdue University for the kind cooperation in the code checking of LEMA 3, and for the use of some machine time. In particular, the help of Mr. Mike Pavloff of the Purdue Computer Center is gratefully acknowledged.

FIG. 1-A QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 2-A QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 3-A QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 4-A QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 5-A

QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 6-A QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 7-A

QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 9-H QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 1-3 QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 2-B QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 3-3 QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 1 B QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 5-B QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 6-B

QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 7-B

QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 8-B QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 9-B

QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 1 QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 2

QUASI-RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 3-C QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 4-C QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 5c QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 6-C

QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 7-C

QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 8-C QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 9-C

QUASI RESONANCES-DOPPLER AND RESOLUTION BROADENING

FIG. 1-D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.

FIG. 2-D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.

FIG. 3-D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.

FIG. 4-D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.

FIG. 1 D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.

FIG. 6-D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.

FIG. 7-D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.

FIG. 8-D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.

FIG. 9-D

QUASI RESONANCES-TOTAL AND SCATTERING C.S.