NUCLEAR REACTIONS ON THE PALLADIUM ISOTOPES

APPROVED:

Tom J. Gray
Major Professor

James A. Roberts
Minor Professor

Director of the Department of Physics

Robert B. Toulouse
Dean of the Graduate School

White, Ronald Lee, Nuclear Reactions on the Palladium Isotopes, Master of Science (Physics), December, 1970, 71 pp., 2 tables, 8 figures, references, 30 titles.

The problem of interest in this investigation was to determine the cross sections of five nuclear reactions which occur when irradiating natural palladium with neutrons which have energy values of 15.1, 15.4, 15.9, and 16.3 MeV. The cross sections were measured relative to a copper monitor which was "sandwitched" in with the palladium target. The palladium and copper samples were counted to determine the gamma ray activity and the energy of each gamma ray present. These gamma ray spectra were analyzed with the computer program SAMPO. An explanatory exposition about SAMPO for the beginning and average user is included as an appencix.

A development of the statistical model of nuclear reactions is included along with a discussion of the model in which its strengths and weaknesses are pointed out. The computer program COMPNUC was written to carry out the computations involved in calculating cross sections by the use of the statistical model. The form assumed for the nuclear temperature was that of A. G. W. Cameron. COMPNUC is listed in an appendix. Graphical comparisons of the cross sections predicted by this model and the cross sections experimentally determined by this and other authors are given.

The cross sections at the neutron energies listed above were found to be: (1) $^{102}Pd(n,2n)^{101}Pd$--1.35±.126, 1.54±.12b, 1.80±.12b and 1.66±.12b. (2) $^{108}Pd(n,2n)^{107m}Pd$--1.85±.26b, 1.63±.26b, 1.96±.26b and 2.24±.26b. (3) $^{110}Pd(n,2n)^{109m}Pd$--3.00±.33b, 3.80±.33b, 4.45±.33b and 4.90±.33b. (4) $^{104}Pd(n,p)^{104m}Rh$-- 117±30mb, 161±30mb, 185±30mb and 183±30mb. (5) $^{108}Pd(n,p)^{108m}Rh$--7.0±1.0mb, 7.6±1.0mb, 9.6±1.0mn and 11.0±1.0mb.

NUCLEAR REACTIONS ON THE PALLADIUM ISOTOPES

THESIS

Presented to the Graduate Council of the
North Texas State University in Partial
Fulfillment of the Requirements

For the Degree of

MASTER OF SCIENCE

By

Ronald Lee White, B. A.
Denton, Texas
December, 1970

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

CHAPTER I

INTRODUCTION

It was the intent of this investigation to bring together the procedures and methods involved in the analysis of complex gamma ray spectra and their relationship to the measurement of (n,2n) and (n,p) reaction excitation functions. In determining the relationship between a particular photopeak in the gamma spectra and a reaction cross section, certain intrinsic properties of the material under study, such as beta-branching ratios and half-lives, must either be accepted from the literature or ascertained experimentally. If the variation of the cross section with energy for (n,2n) and (n,p) reactions are known, then the reactions may be employed as fast neutron detectors and monitors.

Natural palladium was chosen as the sample upon which to conduct studies for three major reasons: (1) Natural palladium is composed of six naturally occuring isotopes of which five are more than ten percent abundant. (2) Very little excitation function work has been done in this mass range. (3) Neutron reactions on palladium lead to a wide range of reaction products, many with half-lives in a realm suitable to obtain good counting statistics within twenty to thirty minutes. Of interest in this paper are the $^{108}Pd(n,2n)^{107m}Pd$, $^{110}Pd(n,2n)^{109m}Pd$, $^{104}Pd(n,p)^{104m}Rh$,

$^{102}Pd(n,2n)^{101}Pd$, and $^{108}Pd(n,p)^{108}Rh$ reactions between 15.1 MeV and 16.3 MeV. All previous cross section measurements made on palladium isotopes have been done using neutron energies between 14.0 MeV and 14.7 MeV. The most intensively studied reaction on a palladium isotope is the $^{110}Pd(n,2n)^{109}Pd$ reaction. For the $^{110}Pd(n,2n)^{109m}Pd$ reaction Mangal and Gill[1] report 971±12mb at 14.0 MeV, Lu, RanaKumar and Fink[2] report 510±35mb at 14.4 MeV, and Minetti and Pasquarelli[3] report 510±30mb at 14.7 MeV. For the $^{110}Pd(n,2n)^{109m+g}Pd$ reaction Lu, RanaKumar and Fink[2] report 1926±185mb at 14.4 MeV, and at 14.7 MeV Minetti and Pasquarelli[3] report 1590±140mb, Bonazzola, Brovetto, Chiavassa, Spinoglio and Pasquarelli[4] report 2570±160mb, and Paul and Clarke[5] report 1948±1000mb. For the $^{108}Pd(n,2n)^{107m}Pd$ reaction Minetti and Pasquarelli[3] report 517±80mb at 14.7 MeV. Finally, at 14.4 MeV Lu, RanaKumar and Fink[2,6] report 637±45 for the $^{102}Pd(n,2n)^{101}Pd$ reaction and 8.3±1.5mb for the $^{108}Pd(n,p)^{108}Rh$ reaction.

In order to compare experimental measurements with theoretical predictions, a computer program has been written to do theoretical calculations based on the statistical model. The statistical theory gives an account of (n,2n) and (n,p) reactions for nuclei with a mass number greater than approximately 50, in terms of a rather simple model. Computation of theoretical cross section values by use of this model requires knowledge of two parameters, the nuclear temperature and the cross section for the emission of the

first neutron from the compound nucleus, which are generally not known.

CHAPTER II

EXPERIMENTAL METHOD

Introduction

In order to obtain activation cross sections, the parallel disc method was used. This method embodies the juxtaposition of a disc of the sample material and a disc of the monitor material. The criterion for the selection of the neutron flux monitor is that it must have an accurately measured excitation function over the energy range of interest and that its reaction products should have half-lives comparable to the half-lives of the reaction products to be studied. One of the principal advantages of this method is that natural elements, generally, can be used as samples instead of enriched or separated isotopes, since with high resolution Ge(Li) detectors all of the gamma rays can be identified and their half-lives followed simultaneously.

Sample Preparation and Activation

Copper was chosen for the monitor material, since the (n,2n) excitation functions of its two naturally occurring isotopes, ^{63}Cu and ^{64}Cu, have been extensively studied[7-12]. Also, the half-lives of the two reaction products, ^{62}Cu ($T_{1/2}$ = 9.9 min.) and ^{64}Cu($T_{1/2}$ = 13.59 hr.), allow cross section measurements to be made for sample reaction products with half-lives ranging from a few seconds to many hours.

Copper monitors were obtained from Reactor Experiments, Incorporated in the form of discs having a radius of .375 in., a thickness of .010 in., and a chemical purity of 99.99 per cent. The palladium sample was obtained from Alpha Inorganic, Incorporated in the form of a foil having a thickness of .025 in. and a chemical purity of 99.99 per cent. Activation samples were prepared by cutting discs of .375 in. radius from this foil.

The neutron fluxes used in this work were produced on the Regional Nuclear Physics Laboratory's[13] 2MV Van de Graff accelerator, utilizing the $^{3}T(D,n)^{4}He$ reaction.

$$^{3}T_{1} + ^{2}D_{1} \longrightarrow ^{4}He_{2} + ^{1}n_{0} + 17.6 \text{ MeV} \qquad (1)$$

The deuteron bombarding energy was varied from 300 KeV to 1000 KeV. At these bombarding energies, the neutron energies at 0^{0}, range from 15.38 MeV to 15.77 MeV. The activation sample was placed at a distance .4 cm. from the neutron source during irradiation and therefore subtended an angle from 0^{0} to 67^{0}. Since the neutron energy is dependent upon the angle of emission, program MARSHEN[14] was used to calculate the neutron energies entering the sample, the weighted average neutron energy, and the standard deviation of the energies entering the sample.

Due to the tenuity of the activation samples, the attenuation of the neutron flux upon passage through either the copper or palladium discs was negligible. This was verified

by constructing a sample which was composed of a palladium disc "sandwiched" between two copper discs, irradiating the sample, and then counting the positron annihilation activity in the two copper discs separately. This measurement showed the same amount of gamma activity in each disc, within the statistical error of the experiment.

Gamma Spectra

After an irradiation period of approximately forty minutes the activation samples were taken from the neutron beam and counted separately for gamma activity. The palladium sample was counted first, since it has many short-lived product nuclei. The apparatus used in counting the samples was a Canberra true coaxial Ge(Li) detector which has a resolution of 2.4 KeV and an efficiency of 3.5 per cent, relative to a 3x3 inch NaI(Tl) detector at 25 cm., for the 1.33 MeV radiation from ^{60}Co, an Ortec 440A Selectable Active Filter Amplifier and a 1024 channel Nuclear Data-150 Analyzer which is interfaced to an IBM-24 keypunch for direct readout onto punched cards. A relative photopeak efficiency curve for gamma rays in the geometries used was constructed by normalizing several curves to a curve constructed from the known intensities of the ^{133}Ba decay.

All gamma ray spectra taken in conjunction with this work were analyzed with SAMPO[15]. SAMPO is a general-purpose semiconductor spectral analysis code. The program itself is

capable of determining the necessary peak shape parameters, provided the user can supply strong, "clean", singlet photo peaks in the region of interest so that the empirical peak shapes may be reduced to functional form. The general formalism presumed to describe photopeaks consists of a central gaussian, with smoothly joined high and low energy exponential tails. The peak shapes are normally found to vary continuously as a function of energy; therefore, the exact shape parameters at each point in a spectrum can in principle be obtained by interpolation between points for which standard peakshapes have been determined. Given accurate peak shape parameters, the program is easily capable of resolving very close-lying multiplets that could be visually detected only with great difficulty. Appendix B contains the instructions for the use of SAMPO on the IBM 360/50 computer.

Of particular interest in this work are the photopeaks at 97.1, 188.0, 212.0, 581.0, and 590.3 KeV, which come from the reactions $^{104}Pd(n,p)^{104m}Rh$, $^{110}Pd(n,2n)^{109m}Pd$, $^{108}Pd(n,2n)^{107m}Pd$, $^{108}Pd(n,p)^{108m}Rh$, and $^{102}Pd(n,2n)^{101}Pd$ respectively. There are many other photopeaks in the spectrum which can be readily identified, but either their areas were too small for an accurate measurement or they were multiplets which were so closely spaced that accurate intensity determinations were impossible. In the latter case the use of separated isotopes would provide much more reliable results.

Equations for Data Analysis

The rate at which a radioactive substance decays is proportional to the remaining number of radioactive atoms,

$$dN/dt = -\lambda N, \tag{2}$$

in which

$$\lambda = \ln(2)/T_{\frac{1}{2}}, \tag{3}$$

where λ is the decay constant for the radioactive substance, and $T_{\frac{1}{2}}$ is the half-life. Integrating, and substituting initial conditions yields

$$N = N_0 \exp(-\lambda t), \tag{4}$$

where N_0 is the number of radioactive nuclei at time t = 0. If all the decays in a certain time interval from ts to t0 are counted, we obtain

$$N = N_0\{\exp(-\lambda ts) - \exp(-\lambda t0)\}. \tag{5}$$

Since Ge(Li) detectors are not 100 per cent efficient, and since the detector does not subtend 4π steradians, an efficiency factor, f, and a geometry factor, G, must be entered into the equation to account for these physical limitations,

$$N_0 = fGN = fGN_0\{\exp(-\lambda ts) - \exp(-\lambda t0)\}, \tag{6}$$

where f and G must be less than 1 for a real detector. If

more than one nuclear state is populated in the decay of the product nucleus and if the nuclear state depopulates by more than one mode, factors, B and D, must be included to account for these multiple branchings. Thus the number of gamma rays counted, N_{cl}, is

$$N_{cl} = BN_{c/D} = BfGN_0\{\exp(-\lambda ts) - e(-\lambda to)\}/D, \qquad (7)$$

where B and D must be less than or equal to 1.

In many nuclei during gamma-decay, an interaction takes place between the emitted gamma and an inner shell electron. This process, known as internal conversion, causes the electron to be ejected from the atom after absorbing the gamma ray's energy. Thus the true intensity of a particular gamma ray is actually a sum of its observed intensity and the number of electrons emitted, and therefore, a correction factor is needed. These correction factors may be found in the literature for many nuclei. The true number of gamma emissions is related to the number of gammas counted, N_{cp}, by

$$(1 + \alpha)N_{cp} = BfGN_0\{\exp(-\lambda ts) - \exp(-\lambda to)\}/D, \qquad (8)$$

where α is the ratio of the number of conversion electrons emitted to the number of gamma rays emitted. Solving Eq. (8) for N_o gives

$$N_o = \frac{N_{cp}(1+\alpha)D}{BfG\{\exp(-\lambda ts) - \exp(-\lambda to)\}} \qquad (9)$$

Using activation analysis theory which says,

$$I = N_a \phi \sigma \{1 - \exp(-\lambda t i)\}, \tag{10}$$

where I is the intensity of radioactive decay at the end of the irradiation period; N_a is the number of target nuclei; σ is the cross section for the reaction; ϕ is the neutron flux during irradiation, assumed constant; λ is the decay constant of the product nucleus; and ti is the time of irradiation. As shown previously in Eq. (2), the intensity at the instant neutron irradiation stops is

$$(dN/dt)\Big|_{t=0} = -\lambda N_o. \tag{11}$$

Therefore,

$$-\lambda N_o = N_a \phi \sigma \{1 - \exp(-\lambda t i)\}. \tag{12}$$

Taking the ratio of equations of the above type, the unknown cross section may be measured relative to the monitor reaction, whose cross section is well known. Letting the subscript 1 denote the sample material and 2 denote the monitor material, it is found that

$$\frac{-\lambda_1 N_{01}}{-\lambda_2 N_{02}} = \frac{N_{a1} \phi_1 \sigma_1 \{1 - \exp(-\lambda_1 t i)\}}{N_{a2} \phi_2 \sigma_2 \{1 - \exp(-\lambda_2 t i)\}} \tag{13}$$

or, rearranging

$$\frac{\sigma_1}{\sigma_2} = \frac{\lambda_1 N_{01} N_{a2} \phi_2 \{1 - \exp(-\lambda_2 t i)\}}{\lambda_2 N_{02} N_{a1} \phi_1 \{1 - \exp(-\lambda_1 t i)\}}. \tag{14}$$

As mentioned in a previous section, ϕ_1 is approximately equal

to ϕ_2, and also

$$\frac{\lambda_1}{\lambda_2} = \frac{\ln(2)/T_{\frac{1}{2},1}}{\ln(2)/T_{\frac{1}{2},2}} = \frac{T_2}{T_1} \quad , \tag{15}$$

where T_1 is the half-life for the sample, and T_2 is the half-life for the monitor, so,

$$\frac{\sigma_1}{\sigma_2} = \frac{T_2 N_{01} N_{a2}\{1 - \exp(-\lambda_2 ti)\}}{T_1 N_{02} N_{a1}\{1 - \exp(-\lambda_1 ti)\}} \quad . \tag{16}$$

The number of nuclei in the sample can be found directly from the mass by the equation

$$N_a = \frac{m}{A} N_A \quad , \tag{17}$$

where N_A is Avogadro's number; A is the atomic number; and m is the mass. If the sample is a homogeneous mixture of many isotopes,

$$N_a = N_A m \sum_i (P_i/A_i), \tag{18}$$

where the P_i are the percentage abundances of the several isotopes in the sample and the A_i are their respective atomic numbers. Since a study can be made of only one isotope at a time, let

$$N_a = N_A P_1 m/A_1 \tag{19}$$

so that

$$\frac{N_{a1}}{N_{a2}} = \frac{A_1 P_2 m_2}{A_2 P_1 m_1} \quad . \tag{20}$$

Since the (n,2n) reactions in the copper monitor leave residual nuclei which both give off 511 KeV annihilation gamma radiation, a correction must be made to N_{cp2} to enable one to extract only that portion of the radiation which comes from the ^{62}Cu. This was done using the activation analysis described previously, which yielded the following equations:

$$X = \sigma_2(\%N_2)\{1 - \exp(-\lambda_2 ti)\}\{\exp(-\lambda_2 ts) - \exp(-\lambda_2 to)\} \quad (21)$$

$$Y = \sigma_3(\%N_3)\{1 - \exp(-\lambda_3 ti)\}\{\exp(-\lambda_3 ts) - \exp(-\lambda_3 to)\} \quad (22)$$

$$N_{cp2*} = \frac{X \cdot N_{cp2}}{X + Y} \quad . \quad (23)$$

Substituting Eqns. (9,20,23) into Eq. (16), the equation for the relative cross section becomes

$$\sigma_1 = \sigma_2 \frac{N_{cp1}(1+\alpha_1)A_1B_2f_2G_2T_2P_2m_2D_2\{1 - \exp(-\lambda_2 ti)\}}{N_{cp2*}(1+\alpha_2)A_2B_1f_1G_1T_1P_1m_1D_1\{1 - \exp(-\lambda_1 ti)\}}$$

$$\times \frac{\{\exp(-\lambda_2 ts_2) - \exp(-\lambda_2 to_2)\}}{\{\exp(-\lambda_1 ts_1) - \exp(-\lambda_1 to_1)\}} \quad . \quad (24)$$

All of the cross sectional values reported in this paper were computed using Eq. (24).

Experimental Errors

The error limits shown in Figs. 1-5 for the measured cross sections are root-mean-square errors and are composed of the following:

(i) Error in the relative photopeak efficiency of the detector. This represents a large contribution to the error, as it is

not possible to get a photopeak efficiency curve better than about 3 per cent accuracy, since the detector must be calibrated with sources calibrated to an accuracy of 1-2 per cent. Also, in the low energy region the curve becomes very steep and even a slight error in efficiency calibration can contribute large errors. In general this error was less than 6 per cent.

(ii) Statistical error. This error in counting statistics was generally less than 3 per cent for all peaks except the peak at 97.11 KeV which was about 10-15 per cent. The reason for this large error is that this peak was relatively small and was sitting on a very steep Compton edge.

(iii) Error due to self-absorption. In the case of low energy gamma rays, the error amounted to 1-2 per cent at most, but is considerably lower than this for the higher energy gamma rays.

(iv) Errors in timing. For long irradiation and counting times, the timing errors are negligible, except for short half-life activities where an error of 2 per cent was assigned. Spectra were taken with approximately 20 per cent dead time in the analyzer. In this manner the dead times should average out over the two counting intervals.

The errors in the monitor cross sections, the internal conversion coefficients, the branching and depopulation ratios, and the half-lives of the sample or monitor activities are not included in the reported error, because any revision in decay schemes and conversion coefficient values permits easy recalculation of the cross sections in the future.

CHAPTER III

STATISTICAL THEORY OF NUCLEAR REACTIONS

Introduction and Background

In a nuclear reaction, the nuclear processes do not begin until the incoming particle, b, and the nucleus, X, have come within the range of the nuclear forces. The processes have ended when the product particle, d, and the residual nucleus, Y, have separated by more than the range of the nuclear force. One model proposes that during the period between the entrance of particle b and the exit of particle d, a compound nucleus is formed whose properties decide the make up of d and Y.

It was Niels Bohr[16,17] who pointed out that it is useful to divide the nuclear reaction into two parts: first the formation of a compound nucleus, C, made up of b and X; and the subsequent disintegration of C into the reaction products d and Y. These two stages can be treated as independent processes, in the sense that the mode of disintegration of C depends only upon its energy, angular momentum, and parity, but not on the manner in which it has been produced. Thus, the two processes can be considered as completely separate processes. This is commonly called the Bohr assumption. It is based upon the supposition that the energy introduced by the arrival of the incident particle is statistically shared among all components of the compound nucleus until one or

more of its constituent particles acquire sufficient energy to escape.

Once the energy carried into X by b is shared with the other nucleons, it takes a great many energy exchanges, and thus, a considerable amount of time, before any one or more particles gain enough energy to be emitted. The large number of exchanges is the main reason for the validity, or approximate validity, of the Bohr assumption in many cases. A thorough mixing of the energy of the incident particle is expected so that the state of C, before emission, shows no traces depending on the special way in which the excitation energy was delivered.

There are two conditions which must be met for the validity of the Bohr assumption to hold. First, the mean free path, M, of an entering nucleon must be much smaller than the range of the nuclear forces,

$$M \approx 1.8 \times 10^{-15} \times (E-E_o) << R, \qquad (25)$$

where E is the bombarding energy in the center of mass system, E_o is the average kinetic energy of the nucleons within the nucleus--which is on the order of 20 MeV--and R is the nuclear radius. Second, the incident energy must be much less than the atomic mass of the target minus one multiplied by the separation energy of particle d,

$$E << (A-1) \times S_d. \qquad (26)$$

Both conditions are fulfilled for nuclei with an atomic number A greater than 10 as long as E is less than 50 MeV. It should also be noted that the above conditions are necessary, but not sufficient, since there may be mechanisms within the compound system which prevent the even distribution of the incident nucleon energy. Nevertheless, for the purpose of this investigation, the Bohr assumption will be considered totally valid.

Theory

According to the Bohr assumption, the cross section of the nuclear reaction X(b,d)Y may be written in the form

$$\sigma(b,d) = \sigma_c(b)\, G_c(d), \qquad (27)$$

where $\sigma_c(b)$ is the cross section for the formation of a compound nucleus formed by particle B and nucleus X, and $G_c(d)$ is the probability that the compound nucleus, once formed, will decay by emitting particle d, leaving a residual nucleus Y. This formulation applies to all but elastic scattering, which is treated separately.

To write an expression for $\sigma_c(b)$, some assumptions need to be made which define the structure of the nucleus for the present consideration. (1) The nucleus has a well defined surface which is a sphere of radius R, and the nuclear forces do not act between b and X if the distance between b and the center of the nucleus is larger than R. (2) If particle b penetrates the nuclear surface, it moves with an average

kinetic energy which is much higher than its energy, E, outside the nucleus. (3) Particle b is subject to strong interactions inside the nucleus so that it interchanges its energy rapidly with the other nucleons. (4) The number of open decay channels is very large. The fourth assumption, unlike the previous three, is peculiar to the continum theory of highly excited nuclei and is fulfilled when the incident energy, E, is much higher than the excitation energies of the first few excited states of the target nucleus.

The cross section for formation of the compound nucleus through entrance channel b may be broken down into a sum of terms,

$$\sigma_c(b) = \sum_{\ell=0}^{\infty} \sigma_{c,\ell}(b). \qquad (28)$$

In order to simplify the present considerations, the dependence of the compound system on parity and on the total angular momentum will be ignored. These terms introduce only small variations in the formation cross section and are, therefore, not essential in view of the very qualitative nature of the present considerations. Since the derivation of the expression for $\sigma_{c,\ell}(b)$ is lengthy and is not of importance to the discussion, it will be omitted. It may be found in several texts on nuclear physics[18-23]. The expression for compound nucleus formations for a given ℓ is

$$\sigma_{c,\ell}(b) = (2\ell + 1)\pi\lambda^2 \frac{-4S_\ell \mathrm{Im}(f_\ell)}{\{\mathrm{Re}(f_\ell)-\Delta_\ell\}^2 + \{\mathrm{Im}(f_\ell)-S_\ell\}^2}, \qquad (29)$$

where:

$$S_\ell = kRV_\ell \tag{30}$$

$$V_\ell^2 = \{F_\ell^2(R) + G_\ell^2(R)\}^{-1} \tag{31}$$

$$F_\ell(R) = kRj_\ell(kR) \tag{32}$$

$$G_\ell(R) = -kR\eta_\ell(kR) \tag{33}$$

$j_\ell(kR)$ = spherical Bessel function of order ℓ (34)

$\eta_\ell(kR)$ = spherical Neumann function of order ℓ (35)

$$k = 1/\lambda\!\!\!^{-} = (2M_b E)^{1/2}/\hbar \tag{36}$$

M_b = reduced mass of particle b (37)

$$R = r_o A^{1/3} \tag{38}$$

r_o = nuclear radius constant $\approx 1.4 \times 10^{-13}$cm. (39)

$$\Delta_\ell = V_\ell R\{G_\ell(dG_\ell/dr) + F_\ell(dF_\ell/dr)\}_{r=R} \tag{40}$$

$f_\ell = R\{(dU_\ell/dr)/U_\ell\}_{r=R}$ = logarithmic derivative of the radial wave function, $U_\ell(r)$, at the nuclear boundary. (41)

If assumption 4 is fulfilled, it is unlikely that the compound system will decay back into the entrance channel. It is thus assumed that the wave function, in the entrance channel, will have the form of an ingoing wave, since it does not return. Therefore, the ingoing wave function has the form,

$$U_\ell(r) \approx \exp(-iKr) \text{ for } r<R, \tag{42}$$

where K is the wave number of the particle inside the nuclear surface. Therefore, a comparison of U_ℓ^2 with the expression

for f_ℓ and the fact that f_ℓ must be continuous at the nuclear boundary give the fundamental assumption of the continuum theory:

$$f_\ell = -iKR. \tag{43}$$

According to Blatt and Weiskopf[18], K can be written as

$$K = (K_o^2 + k^2)^{1/2}, \tag{44}$$

where

$$K_o = (9\pi/8r_o^3)^{2/3}. \tag{45}$$

Combining Eqns. (28-45), the expression for the creation of a compound nucleus is

$$\sigma_c(b) = \pi\lambda\!\!\!^{-2} \sum_{\ell=o}^{\infty}(2\ell+1)\frac{4S_\ell KR}{\Delta_\ell^2 + (KR+S_\ell)^2}. \tag{46}$$

The value of K is the only information about the interior of the nucleus that is needed for the calculations of $\sigma_c(b)$.

According to Eq. (27), the above expression coupled with an expression for $G_c(d)$ will yield the desired formula for $\sigma(b,d)$. The derivation of $G_c(d)$ has been given by Blatt and Weisskopf[18] and Cuzzocrea, Notarrigo, and Perillo[24] for the types of reactions studied in this work. Of particular interest here are the expressions for $\sigma(n,p)$ and $\sigma(n,2n)$:

$$\sigma(n,p) = \sigma_c(n)\exp\{E^*/T\} \tag{47}$$

$$\sigma(n,2n) = \sigma_c(n)\{1 - (1 - E_c/T)\exp(E_c/T)\}, \tag{48}$$

where:

$$E^* = Q_{n,p} + \delta_R - \delta_T - K_p V_p \quad (49)$$

$Q_{n,p}$ = the Q-value for the (n,p) reaction (50)

δ_R = pairing energies for the residual nucleus (51)

δ_T = pairing energies for the target nucleus (52)

V_p = Coulomb potential for protons (53)

K_p = approximation to the penetrability of V_p (54)

$E_c = E + Q_{n,2n} = E - S_n$ (55)

S_n = separation energy of a neutron (56)

T = the nuclear temperature. (57)

If particle b enters the nucleus and quickly shares its energy with all other nucleons, the excited compound nucleus will be in a state where the energy levels are very closely spaced. Owing to experimental uncertainty in the beam energy and in the finite resolution of detectors, it is impossible to determine exactly which energy level is populated. It is thus more reasonable to speak of the density, D, of the nuclear energy levels at the approximate excitation energy. Assumption (3) argues that shortly after the entrance of particle b, all nucleons have the same energy; and therefore, the number of particles emitted is a measure of the statistical probability that a nucleon, or nucleons, will gain enough energy, through collisions with its neighbors, to escape the nuclear potential. This process is similar to that of evaporation from a liquid drop and is known, therefore, as the evaporation model. It

should be noted that the temperature determining the energy distribution of the emitted particles is not the temperature of the compound nucleus before emission, but the temperature of the residual nucleus after emission. In the thermodynamics of the liquid drop, the temperature before and after evaporation is assumed to remain constant, but evaporation of one particle from the compound nucleus constitutes a relatively large loss of energy which reduces the temperature considerably. Using this thermodynamic analogy, if S(E) is defined as the entropy of the system,

$$S(E) = \log D(E), \tag{58}$$

then the nuclear temperature may be stated as

$$T = (dS/dE)^{-1}, \tag{59}$$

where the Boltzmann constant k has been omitted which leads to a temperature which is k times the conventional one. Consequently, the temperature has dimensions of an energy. With these definitions, it is found that the density of states may be expressed as

$$D(E) = \text{constant} \times \exp(2E/T). \tag{60}$$

Using a more rigorous approach, Cameron[25] has derived an expression for the density of states that has been found to compare well with the experimental data available. From Eqns. (47,48), it is seen that what is needed from D(E) is the

expression for the nuclear temperature. Cameron gives this as

$$T = \frac{3}{\pi G}\left[\frac{3}{2} + \left(\frac{9}{4} + \frac{2\pi^2 GU}{3}\right)^{1/2}\right], \tag{61}$$

where

$$U = E + P(Z) + P(N) \tag{62}$$

$$P(N) = \text{Pairing energy for neutrons} \tag{63}$$

$$P(Z) = \text{Pairing energy for protons} \tag{64}$$

$$G = G_N + G_Z \tag{65}$$

$$G_N = \text{density of neutron orbits at the Fermi level} \tag{66}$$

$$G_Z = \text{density of proton orbits at the Fermi level} \tag{67}$$

He has also given a formula by which G may be calculated,

$$G = \frac{\sum_{m=0}^{5} (V_{pm} + V_{nm})T^m}{\sum_{m=0}^{5} T^m}, \tag{68}$$

which includes constants, V_{RM}, that are tabulated in his paper. Since the expression for T involves G and vice versa, an iterative procedure is used to obtain the final values for these parameters.

The computer program, COMPNUC was written to carry out the above procedures and to calculate $\sigma(n,p)$ or $\sigma(n,2n)$ within a given energy range. This program is listed in Appendix A.

It should be remembered that the assumptions made in this theory greatly oversimplify the actual situation. Any

individual properties of the nuclei concerned are not taken into account. Accordingly, the results must be considered useful only as a first orientation regarding the orders of magnitude to be expected and the shape of the excitation function.

CHAPTER IV

DISSCUSION

Experimental and Theoretical Results

All theoretical calculations were carried out with the computer program COMPNUC (described in Chapter 3). The values used for the Coulomb potential, V_p, and the penetrability of the Coulomb potential barrier, K_p, were taken from Dostrovsky and Fraenkel[26], the Q-values from Mattauch and Thiele[27], and the pairing energies from Cameron[25]. These values were used with r_o equal to 1.5×10^{-13} cm. For the three (n,2n) reactions, the semi-empirical predictions of Pearlstein[28] will also be given.

Table I summarizes the values used in the experimental cross section determination. All beta-ray branching ratios

TABLE I

VALUES USED TO CALCULATE RELATIVE CROSS SECTIONS

	^{62}Cu	^{101}Pd	^{107m}Pd	^{109m}Pd	^{104m}Rh	^{108m}Rh
half-life	9.9m	8.4h	225	4.7m	4.41m	6m
conversion coeff.	0	0	.3	.52	.4	0
β-branching ratio	100%	39%	100%	100%	3.8%	80%
γ-branching ratio	100%	62%	100%	100%	100%	63.9%
γ-energy (KeV)	511	590.3	212	188	97.99	581.1

and gamma ray energies and branching ratios were taken from Lederer, Hollander, and Perlman[29], with the exception of the values for ^{108m}Rh, which were taken from Pinston, Schussler, and Moussa[30]. ^{108m}Rh is a recent discovery and has not yet been incorporated into any of the isotope data tables. All energy error bars represent one standard deviation of the neutron energies entering the sample.

The theoretical and experimental curves are in good agreement with the ^{108}Pd(n,2n)^{107m}Pd, ^{102}Pd(n,2n)^{101}Pd, and ^{108}Pd(n,p)^{108m}Rh reactions. The latter two reactions lack information about internal conversion coefficients for the 590.3 KeV and 581.1 KeV gamma rays, respectively, but they should be very small. Both the ^{104}Pd(n,p)^{104m}Rh reaction and the ^{110}Pd(n,2n)^{109m}Pd reaction excitation functions are well above the predicted values. The ^{104}Pd(n,p)^{104m}Rh excitation function has never been reported in the literature; thus no comparison can be made which would show the validity of these values. The ^{110}Pd(n,2n)^{109m}Pd has, however, been done by several experimenters between 14 MeV and 15 MeV, and their results seem to indicate that the experimental results reported here are too large by approximately a factor of three. This could be due to two things: (1) an unknown gamma ray of approximately the same energy could have made the 188 KeV gamma ray's intensity seem much larger than its true value, or (2) during the accumulation period, this gamma ray was too large for the memory capacity of the analyzer

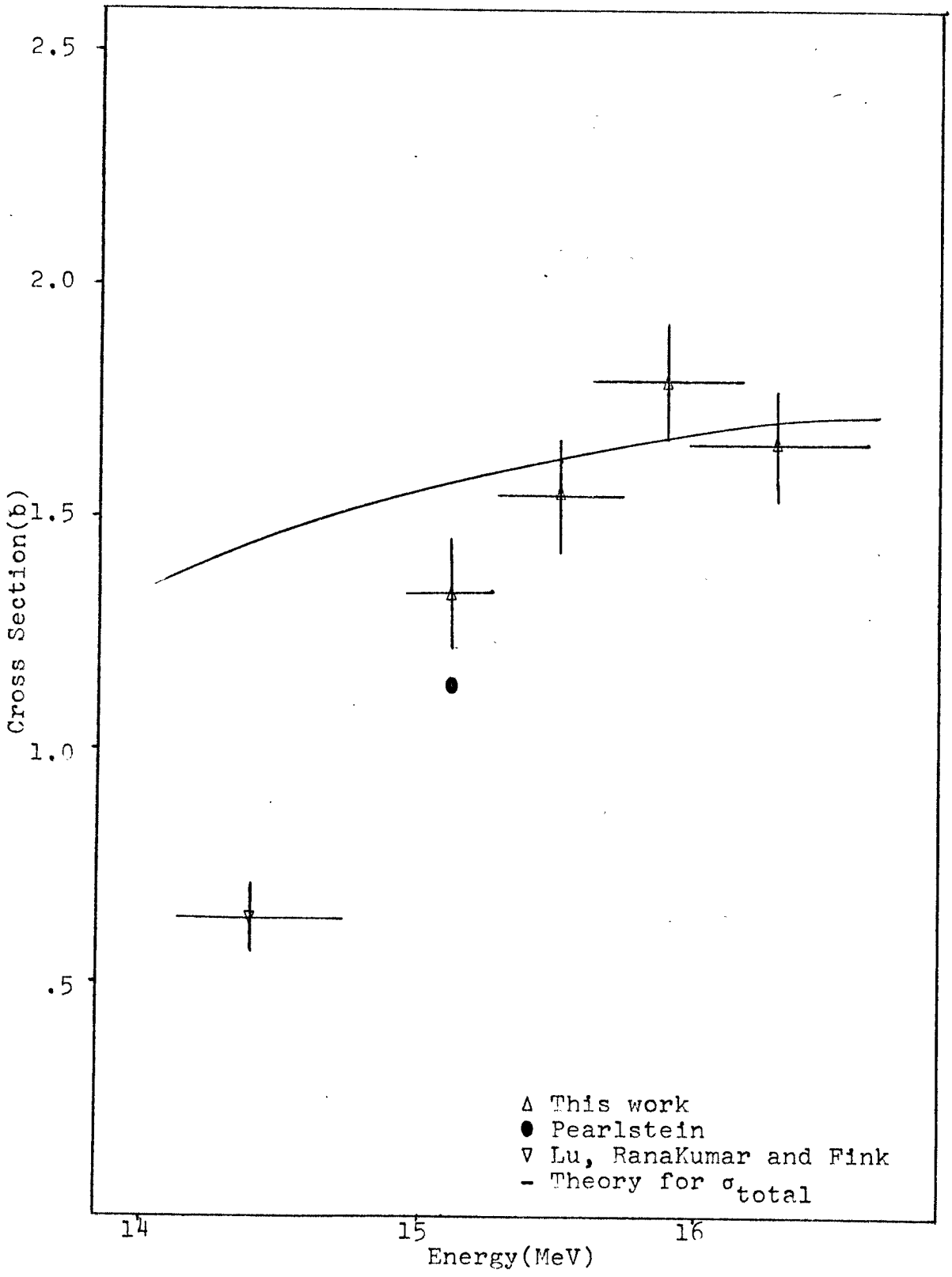

Fig. 1--$^{102}Pd(n,2n)^{101}Pd$ Cross Sections

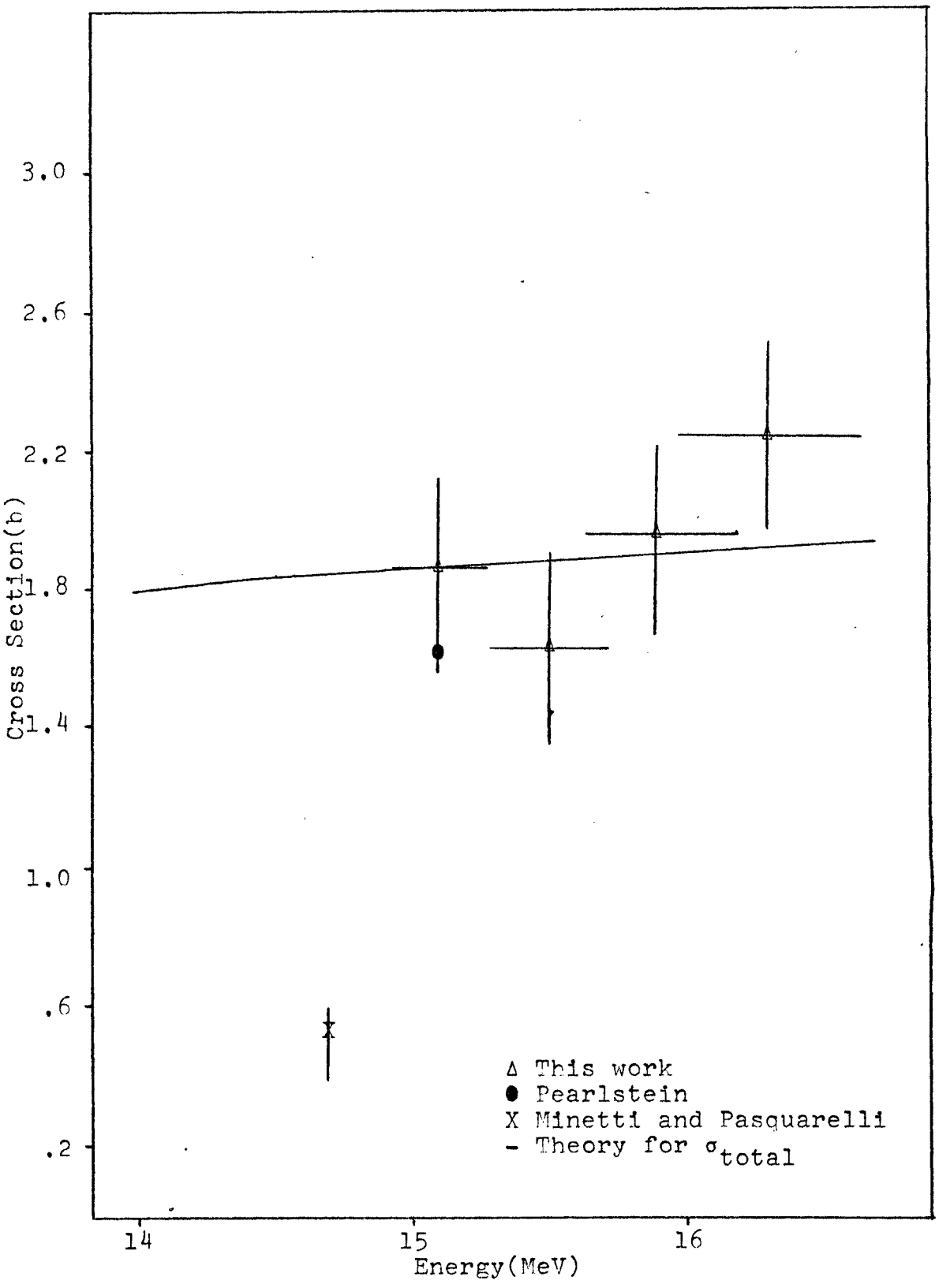

Fig. 2--$^{108}Pd(n,2n)^{107m}Pd$ Cross Sections

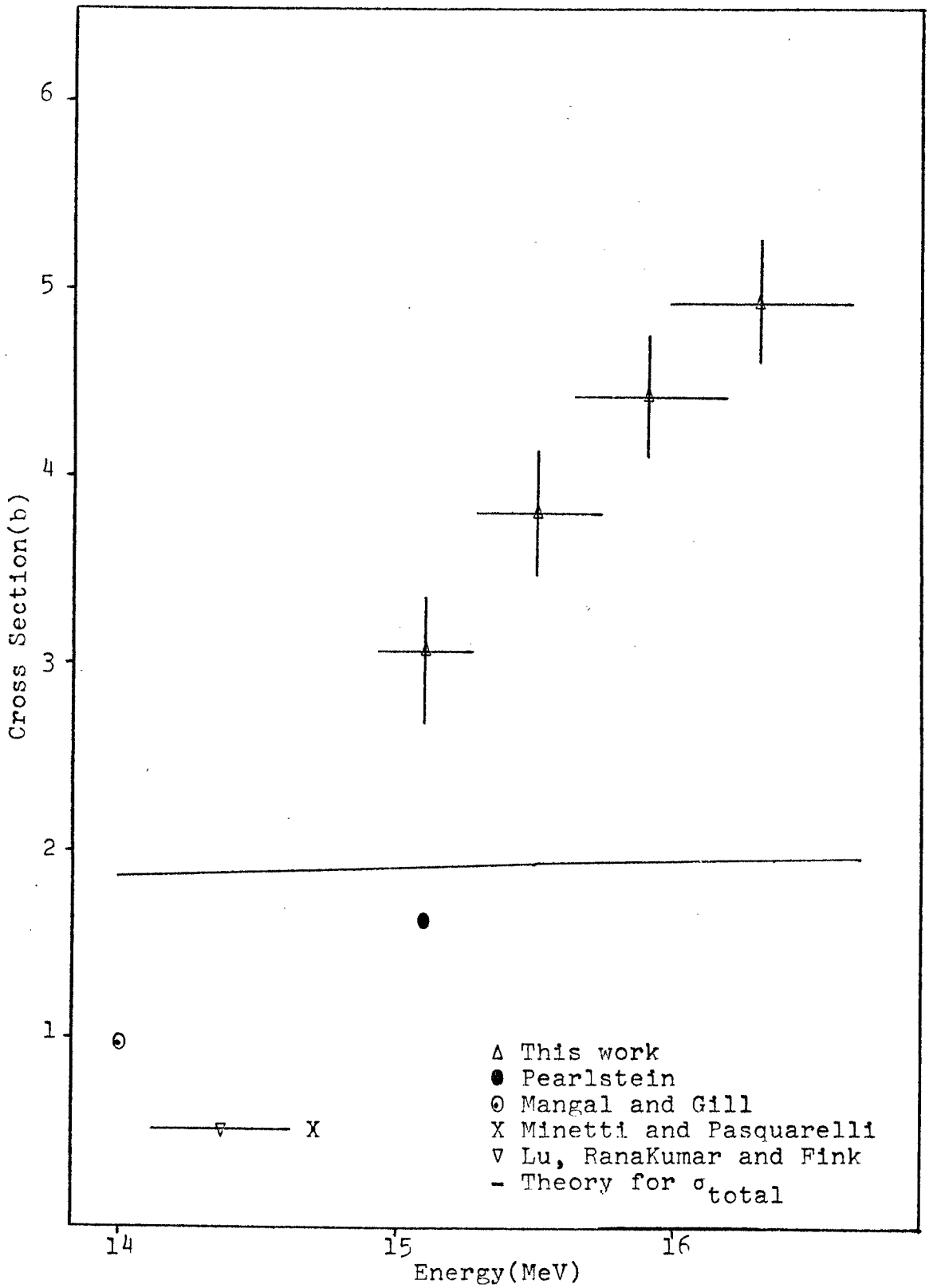

Fig. 3--$^{110}Pd(n,2n)^{109m}Pd$ Cross Sections

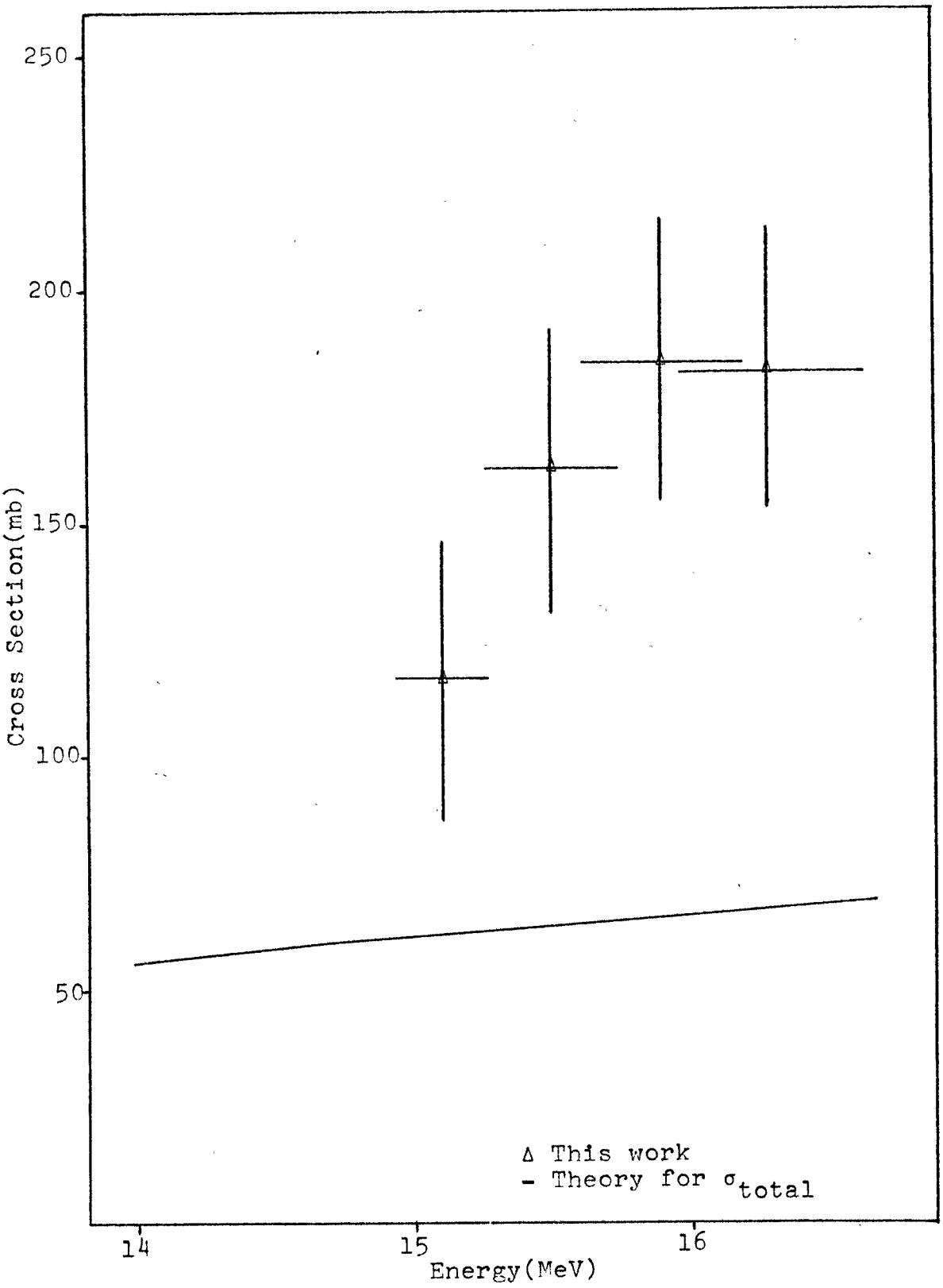

Fig. 4--^{104}Pd(n,p)^{104m}Rh Cross Sections

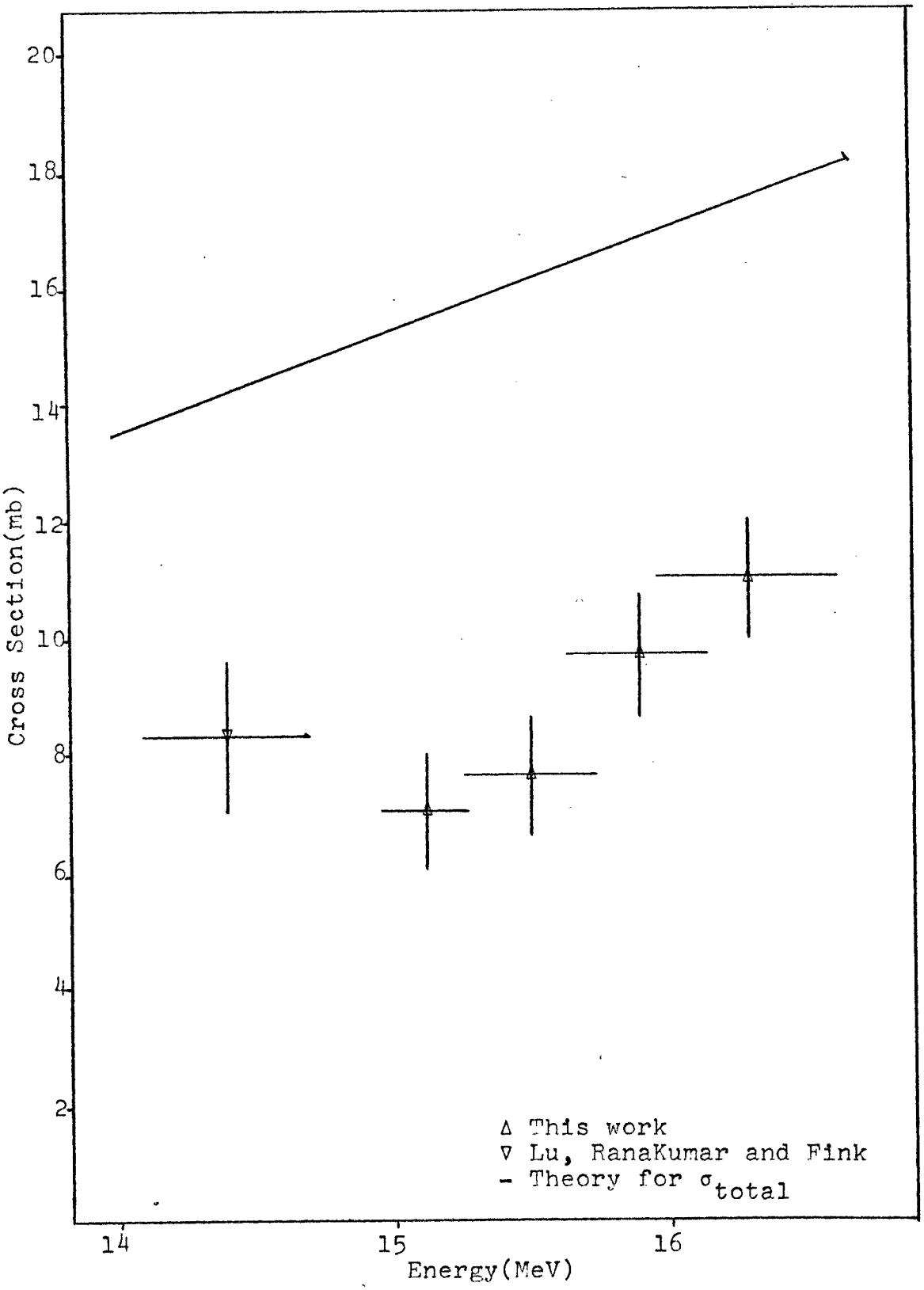

Fig. 5--$^{108}Pd(n,p)^{108m}Rh$ Cross Sections

and the overlap had to be unfolded by hand, which may have been done incorrectly, giving rise to a large peak error.

In general the slopes of the experimentally determined excitation functions seem to be steeper than those of the theoretical curve. This may be due to some unknown flaw in the experimental procedure, or else this is truly the shape in this region. This shape would coincide with the theoretical predictions if the peaks of the theoretical cross sections were shifted to a slightly higher energy or if the magnitude of the peaks in the theoretical cross sections were too small. Since four of the reaction products are metastable states only, the theoretical predictions, which are total cross sections, will give only a theoretical maximum limit on the metastable state cross sections.

Conclusions

This work was undertaken to gain insight and experience into the techniques and problems associated with the experimental data reduction and analysis of complex systems while at the same time gaining experimental and theoretical knowledge of nuclear reaction systematics. The results presented here are meant only to be a first orientation to the cross-sectional values. The gamma spectra are quite complex, containing many multiplets and extremely small photopeaks. This and many other problems mentioned previously point up the need for much work to be done in this area, using separated isotope techniques before definite assignments

should be made to the cross-sectional values. Once this work has been done on several of the isotopes in this mass region, the procedure followed in this work could be reversed, and complex systems made up of more than one element could be analyzed quickly, yielding very accurate determinations of the percentage abundance of all isotopes in the sample. This type of determination has an unlimited number of uses. If this type of analysis were done with charged particles, not only could isotope abundances be found, but also their locations in the sample could be pinpointed. This technique would be very useful to semiconductor manufacturers in that they could find precisely what impurities were in their products and where in the sample they were located. A study directed along these lines is already in progress.

It was found that a very useful aid in this type of work would have been a table of gamma rays by energy. If such a table were made, listing only a few of the most intense photopeaks from each element, the first stage of the analysis, identification of the elemental composition, could be sped up tremendously.

APPENDIX A

```
C    ********** PROGRAM COMPNUC **********
C    *                                     *
C    * THIS PROGRAM CALCULATES THE STATISTICAL MODEL (N,P), *
C    * (N,A), AND (N,2N) CROSS SECTIONS AS A FUNCTION OF   *
C    * BOMBARDING ENERGY.  THE FORM ASSUMED FOR THE NUCLEAR *
C    * TEMPERATURE IS THAT OF CAMERON WHICH WAS USED IN HIS *
C    * FORMULATION OF THE DENSITY OF NUCLEON ORBITS AT THE  *
C    * FERMI LEVEL.  THIS FORMULATION INCLUDES TERMS WHICH  *
C    * TAKE INTO ACCOUNT THE PAIRING OF NUCLEONS IN EVEN-   *
C    * EVEN, EVEN-ODD, AND ODD-EVEN NUCLEI.                *
C    **                                                   **
C    *******************************************************
C    **                                                   **
C    *                    LIST OF DEFINITIONS              *
C    *                    *******************              *
C    *                                                     *
C    * A = THE ATOMIC MASS NUMBER OF THE TARGET NUCLEUS     *
C    * Z = THE NUMBER OF PROTONS IN THE TARGET NUCLEUS      *
C    * MTARG = THE MASS OF THE TARGET NUCLEUS IN ATOMIC MASS*
C    *         UNITS (AMU)                                 *
C    * Q = THE Q-VALUE OF THE REACTION                     *
C    * ELAB = THE INITIAL BOMBARDING ENERGY IN THE LAB     *
C    *        SYSTEM                                       *
C    * EMAX = THE FINAL BOMBARDING ENERGY IN THE LAB SYSTEM *
C    * STEP = THE INCREMENTAL STEP IN THEBOMBARDING ENERGY  *
C    * KTYPE = THE TYPE OF REACTION TO BE CALCULATED       *
C    *       = 1 -- (N,P), = 2 -- (N,A),= 3 -- (N,2N)      **
C    * SPORA =THE BINDING ENERGY OF THE PROTON OR ALPHA     *
C    *         PARTICLE TO THE TARGET ((N,P) AND (N,A) ONLY)*
C    * R1 = THE VALUE OF RO USED IN THE EQUATION R=RO*A**1/3*
C    *          (INITIALIZED AT 1.5*10**-13 CM.)           *
C    * IAGAIN = 0 READ A NEW SET OF DATA , INITIALIZE RO    *
C    *        = 1 REDO THE CALCULATION USING A NEW RO       *
C    * GRAPH =THE TYPE OF GRAPH DESIRED(SEE SUBROUTINE PLOT)*
C    *                                                     *
C   *                        ******                        *
C    *                                                     *
C    * THE FOLLOWING VALUES MAY BE OBTAINED FROM CAMERON S  *
C    * PAPER IN THE CANADIAN JOURNAL OF PHYSICS 36, P.1040  *
C    *                                                     *
C    * AN(I) AND AZ(I) = CONSTANTS USED IN THE CALCULATION  *
C    *                   OF THE NUCLEAR TEMPERATURE        *
C    * DELT = SUM OF THE PAIRING ENERGIES FOR THE TARGET    *
C    * DELR = SUM OF THE PAIRING ENERGIES FOR THE RESIDUAL  *
```

```
C    *                                                              *
C    ****************************************************************
C
       IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 K,KR,MNEUT,MTARG
       SQRT(X)=DSQRT(X)
       EXP(X)=DEXP(X)
       IFIX(X)=IDINT(X)
       ABS(X)=DABS(X)
       FLOAT(I)=DFLOAT(I)
       DIMENSION NAT(100),E(1000),COMPD(1000),Y(5),AN(5),
      *          AZ(5),REACT(3),PK(10),PA(10)
       COMMON FJ(150),FN(150)
       COMMON/CONTRL/ INPUT,IPRINT
       COMMON/PRINTO/NAT,REACT,MN,MX,MNN,MY,KTYPE
       DATA PA/.68,.82,.91,.94,.97,.975,.98,.98,.98,.98/
       DATA PK/.42,.58,.68,.725,.77,.785,.80,.80,.80,.80/
       DATA PI,HBAR,MNEUT,C,EMNEUT,ESU/3.1415927,6.5817E-22,
      *1.008665,2.99793E+10,939.505,4.80286E-10/
       EQUIVALENCE(SN,SPORA)
C
  999 READ(INPUT,3,END=69)IAGAIN,R1
    3 FORMAT(I2,E10.4)
       IF(IAGAIN.EQ.1)GO TO 14
       RO=1.5E-13
       READ(INPUT,2)A,Z,MTARG,Q,ELAB,EMAX,STEP,KTYPE
       READ(INPUT,2)DELT,DELR
       READ(INPUT,2)(AN(I),I=1,5)
       READ(INPUT,2)(AZ(I),I=1,5),SPORA,GRAPH
    2 FORMAT(7E10.4,I10)
       LTOP=IFIX(1.+(EMAX-ELAB)/STEP)
       ELAB1=ELAB
       DO 40 M=1,5
   40 Y(M)=AN(M)+AZ(M)
       MX=IFIX(Z)
       MN=IFIX(A)
       IF(KTYPE.EQ.3) GO TO 11
       NU=IFIX(Z/10.)
       FURD=Z-FLOAT(NU)*10.
       GO TO(12,13,11),KTYPE
   12 MNN=MN
       MY=MX-1
       P1=((PK(NU+1)-PK(NU))*FURD/10.)+PK(NU)
       GOTO 14
   13 MNN=MN-3
       MY=MX-2
       P1=((PA(NU+1)-PA(NU))*FURD/10.)+PA(NU)
       GO TO 14
   11 MNN=MN-1
       MY=MX
       SN=-Q
```

```
   14 IF(IAGAIN.EQ.1) ELAB=ELAB1
      IF(R1.NE.0.) RO=R1
      R=RO*A**.33333333
      IF(KTYPE.NE.3) XV=(Z*FLOAT(KTYPE)*ESU*ESU)/(R+
     *FLOAT(KTYPE-1)*1.2E-13)*6.24181E+5
      WRITE(IPRINT,30)MN,NAT(MX),REACT(KTYPE),MNN,NAT(MY),
     *                ELAB,EMAX
      WRITE(IPRINT,37)RO,R
      WRITE(IPRINT,80)REACT(KTYPE)
      ELAB=ELAB*MTARG/(MTARG+MNEUT)
      DO 5 J=1,LTOP
C
C     ******* CALCULATION OF THE NUCLEAR TEMPERATURE *******
C
      U=ELAB+DELT
      IF(KTYPE.NE.3) U=ELAB+Q+SPORA+DELR
      SIG=1.E-10
      FACTOR=0.0
      COMPD(J)=0.0
      IF((ELAB+Q ).LE.0.0) GO TO 9
      G=AN(5)+AZ(5)
      T=(3./PI/PI/G)*(3./2.+SQRT(9./4.+2.*PI*PI*G*U/3.))
    7 T1=T
      G2=0.0
      T2=0.0
      DO6 M=1,5
      G2=G2+Y(M)*T**(M-1)
    6 T2=T2+T**(M-1)
      G=G2/T2
      T=(3./PI/PI/G)*(3./2.+SQRT(9./4.+2.*PI*PI*G*U/3.))
      IF(ABS(2.*(T1-T)/(T1+T)).GT..01)GOTO 7
C
C     ****** CALCULATION OF THE CROSS SECTIONFOR THE ******
C     ********** FORMATION OF A COMPOUNDNUCLEUS **********
C
      K=SQRT(2.*EMNEUT*ELAB)/C/HBAR
      CAPK=SQRT((9.*PI/8.)**.666666/ /RO/RO+K*K)
      KR=K*R
      CALL BESL(32,KR)
      DO 8 M=1,30
      FPRIME=K*FLOAT(M)*FJ(M)-K*K*R*FJ(M+1)
      GPRIME=-K*FLOAT(M)*FN(M)+K*K*R*FN(M+1)
      VL=1./((FN(M)*FN(M)+FJ(M)*FJ(M))*K*K*R*R)
      SL=K*R*VL
      DELTAL=(-FN(M)*GPRIME+FJ(M)*FPRIME)*R*R*K*VL
      PIG=(2.*FLOAT(M)-1.)*4.*CAPK*R*SL/(DELTAL**2+(CAPK*R+
     * SL)**2)
      IF(ABS(PIG/SIG).LT..01) GO TO 10
    8 SIG=SIG+PIG
   10 SIG=(SIG+PIG)*PI*1.E+2//K/K
      GO TO (18,18,17),KTYPE
```

```
   17 B=(ELAB-SN)/T
      FACTOR= 1.-(1.+B)*EXP(-B)
      GO TO 19
   18 FACTOR=EXP((Q+DELR-DELT-P1*XV)/T)
      IF(KTYPE.EQ.2)FACTOR=2.*FACTOR
   19 COMPD(J)=SIG*FACTOR
    9 E(J)=ELAB1+STEP*FLOAT(J-1)
      ECM=E(J)*MTARG/(MTARG+MNEUT)
      ESTAR=E(J)+SN
      WRITE(IPRINT,60)E(J),ECM,ESTAR,T,SIG,FACTOR,COMPD(J)
    5 ELAB=(E(J)+STEP)*MTARG/(MTARG+MNEUT)
C
C     ******* PLOT GRAPH OF CROSS SECTION VS. ENERGY *******
C
      CALL PLOT(E,COMPD,1,LTOP,IFIX(GRAPH),IPRINT)
      CALL OVFLO(0)
      WRITE(IPRINT,70)
      GO TO 999
   69 CALL EXIT
   30 FORMAT(15X,39HTHE STATISTICAL MODEL CROSS-SECTION FOR,
     *I4,1H-,A2,A6,I4,1H-,A3,4HFROM,F6.2,4H TO ,F6.2,4H MEV,
     * /,15X,41HUSING CAMERONS LEVEL DENSITY FORMULATION  ,
     *  ///)
   37 FORMAT(27X,4HRO =,E12.5,4H CM.,27X,3HR =,E12.5,4H CM.,
     *  /)
   80 FORMAT(6X,5HE-LAB,8X,6HE-C.M.,6X,8HE-EXCIT.,4X,
     * 10HNUC. TEMP.,6X,6HSIG(N),10X,6HFACTOR,9X,3HSIG,A6,/,
     * 6X,5H(MEV),9X,5H(MEV),8X,5H(MEV),8X,5H(MEV),9X,
     * 5H(MB.),27X,5H(MB.),///)
   60 FORMAT(4(5X,F8.2),5X,1PE11.5,5X,E11.5,5X,E11.5)
   70 FORMAT(1H1)
      END
      BLOCK DATA
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION NAT(100),REACT(3)
      COMMON/CONTRL/INPUT,IPRINT
      COMMON/PRINTO/NAT,REACT,MN,MX,MNN,MY,KTYPE
      DATA INPUT,IPRINT/5,6/
      DATA NAT/2H H,2HHE,2HLI,2HBE,2H B,2H C,2H N,2H O,2H F,
     1         2HNE,2HNA,2HMG,2HAL,2HSI,2H P,2H S,2HCL,2HAR,
     2          2H K,2HCA,2HSC,2HTI,2H V,2HCR,2HMN,2HFE,2HCO,
     3          2HNI,2HCU,2HZN,2HGA,2HGE,2HAS,2HSE,2HBR,2HKR,
     4          2HRB,2HSR,2H Y,2HZR,2HNB,2HMO,2HTC,2HRU,2HRH,
     5          2HPD,2HAG,2HCD,2HIN,2HSN,2HSB,2HTE,2H I,2HXE,
     6          2HCS,2HBA,2HLA,2HCE,2HPR,2HND,2HPM,2HSM,2HEU,
     7          2HGD,2HTB,2HDY,2HHO,2HER,2HTM,2HYB,2HLU,2HHF,
     8          2HTA,2H W,2HRE,2HOS,2HIR,2HPT,2HAU,2HHG,2HTL,
     9          2HPB,2HBI,2HPO,2HAT,2HRN,2HFR,2HRA,2HAC,2HTH,
     A          2HPA,2H U,2HNP,2HPU,2HAM,2HCM,2HBK,2HCF,2HES,
     B          2HFM/
      DATA REACT/6H (N,P),6H (N,A),6H(N,2N)/
```

```
      END
      SUBROUTINE BESL(L,XX)
C
C     ***************************************************************
C     *                                                             *
C     * THIS SUBROUTINE CALCULATES SPHERICAL BESSEL FUNCTIONS       *
C     * OF THE FIRST AND SECOND KINDS.  BESSEL FUNCTIONS OF         *
C     * THE SECOND KIND ARE COMMONLY CALLED NEUMANN FUNCTIONS       *
C     *                                                             *
C     *                          **********                         *
C     *                                                             *
C     *  FJ(I) IS THE BESSEL FUNCTION OF ORDER I-1                  *
C     *  FN(I) IS THE NEUMANN FUNCTION OF ORDER I-1                 *
C     *                                                             *
C     ***************************************************************
C
      IMPLICIT REAL*8(A-H,O-Z)
      COS(X)=DCOS(X)
      SIN(X)=DSIN(X)
      FLOAT(I)=DFLOAT(I)
      DIMENSION FJ(150),FN(150)
      COMMON  FJ, FN
 3101 CALL OVFLO(-1)
      X=XX
      FN(1)=-COS(X)/X
      FN(2)=(FN(1)-SIN(X))/X
      IF(X-.01) 3102,3102,3104
 3102 FJ0=1.0-X**2/6.
      IF(L-1) 3103,3103,3105
 3103 FJ(1)=FJ0
      FJ(2)=X*(0.33333333-X**2/30.)
      GO TO 3110
 3104 FJ0=SIN(X)/X
 3105 L2=L+5
      FJ(L2+1)=1.E-10
      FJ(L2)=1.0E-10*(2.0*FLOAT(L2)+1.0)/X
      L3=L2-1
      DO 3106 LL=1,L3
      L1=L2-LL
      FL1=L1
      FJ(L1)=(2.*FL1+1.)*FJ(L1+1)/X-FJ(L1+2)
      IF(FJ(L1)-1.E+30) 3106,3106,3111
 3106 CONTINUE
      ZZ=FJ0/FJ(1)
      DO 3107 L1=1,2
 3107 FJ(L1)=ZZ*FJ(L1)
      IF(L-1) 3110,3110,3108
 3108 L2=L2-4
      DO 3109 L1=3,L2
      FJ(L1)=ZZ*FJ(L1)
 3109 FN(L1)=(2.*FLOAT(L1)-3.)*FN(L1-1)/X-FN(L1-2)
```

```
 3110 CALLOVFLO(1)
      RETURN
 3111 DO 3112 L4=L1,L2
 3112 FJ(L4)=1.E-10*FJ(L4)
      GO TO 3106
      END
      SUBROUTINE OVFLO(N)
C
C    ***************************************************************
C    *                                                             *
C    *  SUBROUTINE FOR CHECKING THE OVERFLOW AND UNDERFLOW  *
C    *  INDICATORS                                                 *
C    *                                                             *
C    ***************************************************************
C
      COMMON/CONTRL/ INPUT,IPRINT
 5100 CALL OVERFL(KOOOFX)
      GO TO (5101,5106,5110),KOOOFX
 5101 N1=IABS(N)
     WRITE(IPRINT,5107)
 5120 IF(N)5102,5103,5103
 5110 N1=IABS(N)
      WRITE(IPRINT,5111)
      GO TO 5120
 5102 WRITE(IPRINT,5108)N1
      GO TO 5106
 5103 WRITE(IPRINT,5109)N1
 5106 RETURN
 5107 FORMAT(97X,21HOVERFLOW INDICATOR ON)
 5111 FORMAT(96X,22HUNDERFLOW INDICATORON)
 5108 FORMAT(103X,14HBEFOREROUTINE,13)
 5109 FORMAT(103X,14HDURING ROUTINE,13)
      END
      SUBROUTINE PLOT(XPOINT,YPOINT,KRDBGN,KRDEND,KONTRL,
     *               ITAPE)
C
C    **********************PLOT SUBROUTINE**********************
C    *                                                         *
C    * FORTRAN IV SINGLE PAGE PRINTER PLOT PROGRAM             *
C    ***********************************************************
C    *                                                         *
C    * SUBROUTINE PLOT ARGUEMENT LIST OF DEFINITIONS           *
C    * XPOINT IS THE X-ARRAY OF COORDINATES TO BE PLOTTED      *
C    * YPOINT IS THE Y-ARRAY OF COORDINATES TO BE PLOTTED      *
C    * XPOINT AND YPOINT MAY HAVE MAXIMUMDIMENSIONS OF 1000   *
C    *                                                         *
C    *                    ********************                 *
C    *                                                         *
C    * KRDBGN IS THE SUBSCRIPT OF FIRST POINT TO BE PLOTTED    *
C    *KRDEND IS THE SUBSCRIPT OF FINALPOINT TOBE PLOTTED       *
C  *         *                                       *         *
```

```
C    *                    ********************                          *
C    *                                                                  *
C    * KONTROL IS AN OPTION TO SELECT THE TYPE OFPLOT                   *
C    * KONTRL = 1 CONNECTS CONSECUTIVE POINTS WITH A LINE               *
C   * KONTROL = 0 PLOTTED COORDINATESAS INDIVIDUAL POINTS               *
C    *      *                                                    *      *
C    *                   ********************                           *
C   *                                                                   *
C    * ITAPE IS THE DATA SET REFERENCENUMBER                            *
C    *                                                                  *
C    ********************************************************************
C
C
      IMPLICIT REAL*8(A-H,O-Z)
      IFIX(X)=IDINT(X)
      ABS(X)=DABS(X)
     ALOG10(X)=DLOG10(X)
      DIMENSION NAT(100),REACT(3)
      DIMENSION NAME(113),MATRIX(6,51),LIST(31),XPOINT(1000)
     *,YPOINT(1000)
      DIMENSIONM1(20),M2(8),N1(16),N2(4),N3(2),N4(16),N5(4)
     *,N6(14)
      COMMON/PRINTO/NAT,REACT,MN,MX,MNN,MY,KTYPE
C
C                              **********
C
      DATA M1/1,2,4,8,16,32,64,128,256,512,1024,2048,40968
     *      8192,16384,32768,65536,131072,262144,5242887
C
C                              **********
C
      DATA M2/1,10,100,1000,10000,100000,1000000,10C0000/
      DATAN1/4H    ,4H   *,4H  *,4H  **,4H*  ,4H* *,
     *       4H ** ,4H ***,4H*   ,4H*  *,4H* * ,4H* **,
     *       4H**  ,4H** *,4H*** ,4H****/
C
C                              **********
C
      DATA N2/4H1   ,4H1*  ,4H*  ,4H** /
C
C                              **********
C
      DATAN3/4H1  ,4H*  /
C
C                             **********
C
      DATA N4/4H----,4H---*,4H--*-,4H--**,4H-*--,4H-*-*,
     *       4H-**-,4H-***,4H*---,4H*--*,4H*-*-,4H*-**,
     *      4H**--,4H**-*,4H***-,4H****/
C
C                              **********
```

```
C
      DATA N5/4H1-   ,4H1*   ,4H*-  ,4H** /
      DATA N6/4H0    ,4H1    ,4H2   ,4H3   ,4H4    ,4H5   ,
     *       4H6   ,4H7    ,4H8   ,4H9   ,4H.   ,4H+    ,
     *       4H-    ,4H     /
C
C                              **********
C
C
C    *******SEARCHFOR MINIMUM AND MAXIMUM COORDINATES*******
C
       CALL OVFLO(-2)
      XFIRST = XPOINT(KRDBGN)
       XFINAL= XPOINT(KRDBGN)
       YFIRST = YPOINT(KRDBGN)
       YFINAL= YPOINT(KRDBGN)
       KRD = KRDBGN
     1 KRD= KRD + 1
       IF( XPOINT(KRD) - XFIRST ) 2,3,3
     2 XFIRST = XPOINT(KRD)
     3 IF( XPOINT(KRD) -XFINAL ) 5,5,4
     4 XFINAL = XPOINT(KRD)
     5 IF( YPOINT(KRD) - YFIRST ) 6,7,7
    6 YFIRST = YPOINT(KRD)
     7 IF(YPOINT(KRD) - YFINAL ) 9,9,8
     8 YFINAL= YPOINT(KRD)
     9 IF( KRD - KRDEND) 1,10,10
    10 WRITE( ITAPE,11 )
    11 FORMAT(1H1)
       WRITE(ITAPE,100) MN,NAT(MX),REACT(KTYPE),MNN,NAT(MY)
   100 FORMAT(53X,I4,1H-,A2,A6,I4,1H-,A2,/,
      *       46X,33HCROSS-SECTION(MB)VS.ENERGY(MEV),//)
       IF(( XFINAL - XFIRST )/100.0 ) 15,15,12
    12 IF( XFIRST + XFINAL )13,14,14
    13 IF( ( XFIRST/(XFIRST -XFINAL)) -100000.0 ) 18,18,15
   14 IF(( XFINAL/(XFINAL-XFIRST)) - 100000.0)  18,18,15
    15 WRITE(ITAPE,16)
   16 FORMAT(                    PLOT ERROR,ZERO XPOINT HORIZ
      *     , ONTAL COORDINATE RANGE )
      XFIRST = XFIRST - ABS(0.005 *XFIRST )
       XFINAL = XFINAL + ABS( 0.005 * XFINAL )
       IF(( XFINAL - XFIRST)/100.0 ) 17,17,18
    17 XFIRST= -0.5
       XFINAL = 0.5
    18 IF(( YFINAL - YFIRST)/100.0 ) 22,22,19
    19 IF( YFIRST + YFINAL ) 20,21,21
    20 IF(( YFIRST/(YFIRST - YFINAL))- 100000.0 ) 25,25,22
   21 IF(( YFINAL/(YFINAL - YFIRST)) -100000.0 ) 25,25,22
    22 WRITE(ITAPE,23)
   23 FORMAT(                    PLOT ERROR,ZERO YPOINT VERTIC
      *     , AL RANGE  )
       YFIRST = YFIRST -ABS(0.005 * YFIRST )
```

```
      YFINAL = YFINAL+ ABS( 0.005 *YFINAL )
      IF(( YFINAL - YFIRST)/100.0 ) 24,24,25
   24 YFIRST = -0.5
      YFINAL = 0.5
   25 XSCALE = 100.0/(XFINAL -XFIRST )
      YSCALE = 50.0/( YFINAL- YFIRST)
      DO 26  KOLUMN = 1,6
      DO 26 LINE = 1,51
   26 MATRIX(KOLUMN,LINE) = 0
      IF(KONTRL) 27,27,30
C
C    ************KONTROL= 0, GENERATE POINTPLOT************
C
   27 DO 29 KRD= KRDBGN,KRDEND
      LASTX = 1.5 + (XSCALE * (XPOINT(KRD) - XFIRST ))
      LINE = 51.5 - ( YSCALE * (YPOINT(KRD) -YFIRST ))
      KOLUMN = ( LASTX + 19 )/20
      LOCATN = (20 * KOLUMN ) - LASTX + 1
      KOMPAR = MATRIX( KOLUMN,LINE )/ M1( LOCATN )
      IF( KOMPAR - ( 2 *( KOMPAR/2))) 28,28,29
   28 MATRIX(KOLUMN,LINE) = MATRIX(KOLUMN,LINE) + M1(LOCATN)
   29 CONTINUE
      GO TO 42
C
C    ************KONTROL = 1, GENERATE LINE PLOT************
C
   30 LASTX = 1.5 + ( XSCALE * ( XPOINT(KRDBGN) - XFIRST ))
      LINE = 51.5 - (YSCALE *( YPOINT(KRDBGN)- YFIRST ))
      KOLUMN = (LASTX + 19)/ 20
      LOCATN = ( 20 * KOLUMN ) - LASTX + 1
      MATRIX(KOLUMN,LINE) = M1(LOCATN)
      DO 41 KRD = KRDBGN,KRDEND
      MOVEX = IFIX( 1.5 + ( XSCALE* (XPOINT(KRD) -XFIRST)
     *)) -LASTX
      MOVEY = IFIX( 51.5 - ( YSCALE * (YPOINT(KRD) - YFIRST
     *))) - LINE
      JUMPX = IABS( MOVEX )
      JUMPY = IABS( MOVEY )
      IF(JUMPX ) 31,31,33
   31 IF( JUMPY ) 41,41,32
   32 LAGX = 0
      LAGY = 0
      MULT= JUMPY
      GO TO 38
   33 IF(JUMPY ) 34,34,35
   34 LAGX = 0
      LAGY = 0
      MULT = JUMPX
      GO TO 38
   35 LAGX = (MOVEX * JUMPY)/( 2 * JUMPX )
      LAGY = (JUMPX * MOVEY)/( 2 *JUMPY )
```

```
      IF( JUMPX -JUMPY )36,37,37
   36 MULT = JUMPY
      GOTO 38
   37 MULT = JUMPX
   38 DO 40 J = 1,MULT
      NEWX = LASTX +((( J * MOVEX)+ LAGX)/MULT )
      NEWY = LINE +((( J * MOVEY )+ LAGY )/MULT)
      KOLUMN = ( NEWX + 19)/ 20
      LOCATN = ( 20 * KOLUMN ) - NEWX +1
      KOMPAR = MATRIX( KOLUMN,NEWY)/M1( LOCATN )
      IF( KOMPAR - ( 2 *(KOMPAR/2))) 39,39,40
   39 MATRIX( KOLUMN,NEWY) = MATRIX( KOLUMN,NEWY) +
     * M1( LOCATN )
   40 CONTINUE
     LASTX = NEWX
      LINE = NEWY
   41 CONTINUE
C
C    ***************PRINT THE PLOT STORAGE ARRAY***************
C
   42 LOCK = 0
      FIRST = YFIRST
      FINAL = YFINAL
      GO TO 53
   43 DO 50 LINE = 1,51
      IF( LOCK) 47,47,44
   44 LOCK = LOCK - 1
      DO 45 KOLUMN = 1,5
      INDEX5 =MATRIX(KOLUMN,LINE)/16
      INDEX4 = INDEX5/16
      INDEX3 = INDEX4/4
      INDEX2 = INDEX3/16
      INDEX1 = INDEX2/16
     INDEX6 = MATRIX(KOLUMN,LINE) -16 * INDEX5
      INDEX5 = INDEX5 - 16 * INDEX4
      INDEX4 = INDEX4 - 4 * INDEX3
     INDEX3 = INDEX3 - 16 * INDEX2
      INDEX2 = INDEX2 - 16 * INDEX1
    LIST( 6 * KOLUMN )= N1( INDEX6 + 1 )
      LIST( 6 * KOLUMN - 1 ) = N1( INDEX5 + 1 )
      LIST( 6 * KOLUMN - 2 ) = N2( INDEX4 + 1 )
      LIST( 6 * KOLUMN -3 ) = N1( INDEX3 + 1 )
    LIST( 6 * KOLUMN -4 ) = N1( INDEX2 + 1 )
   45 LIST( 6 * KOLUMN - 5 ) = N2( INDEX1 + 1 )
      INDEX1 = 1+ (MATRIX(6,LINE)/524288 )
      LIST(31) = N3(INDEX1 )
      WRITE(ITAPE,46) (LIST(I),I=1,31 )
   46 FORMAT(16X,10(A2,2A4),A1)
      GOTO 50
   47 LOCK= 4
      DO 48 KOLUMN = 1,5
```

```
      INDEX5 = MATRIX(KOLUMN,LINE)/16
      INDEX4 = INDEX5/16
      INDEX3 = INDEX4/4
      INDEX2 =INDEX3/16
      INDEX1 = INDEX2/16
      INDEX6 = MATRIX(KOLUMN,LINE) -16 *INDEX5
      INDEX5 = INDEX5 - 16 * INDEX4
      INDEX4 = INDEX4 - 4 * INDEX3
     INDEX3 = INDEX3 - 16 * INDEX2
      INDEX2 =INDEX2 - 16 * INDEX1
      LIST( 6* KOLUMN) = N4(INDEX6 + 1 )
      LIST( 6 * KOLUMN - 1 ) = N4( INDEX5 + 1 )
     LIST( 6 * KOLUMN - 2 ) = N5( INDEX4 + 1 )
      LIST( 6 * KOLUMN - 3 ) = N4( INDEX3 + 1 )
     LIST( 6 * KOLUMN -4 ) = N4( INDEX2 + 1 )
  48 LIST( 6 * KOLUMN - 5 ) = N5( INDEX1 + 1 )
      INDEX1 = 1 +( MATRIX(6,LINE)/524288)
      LIST(31) = N3( INDEX1 )
     INDEX1 = 101 - ( 2 * (LINE-1))
      INDEX2 = INDEX1 + 9
     WRITE(ITAPE,49) (NAME(I),I=INDEX1,INDEX2),
     *                (NAME(I),I=111,113),(LIST(I),I=1,31)
  49 FORMAT(1X,10A1,1HE,3A1,1X,10(A2,2A4),A1 )
  50 CONTINUE
      FIRST = XFIRST
      FINAL = XFINAL
      GO TO 53
  51 WRITE(ITAPE,52) (NAME(I),I=1,110),((NAME(I),I=111,113)
     *                                  ,J=1,11)
  52 FORMAT(1HO,11X,110A1/9X,11(6X,1HE,3A1))
      CALL OVFLO( 2)
      RETURN
C
C    *********ALPHANUMERIC CODE SCALENUMBER ARRAYS**********
C
  53 STEP= ( FINAL -FIRST)/10.0
     KOUNT = ALOG10(1.01 * STEP ) -2.0
      IF(( 1.01 * STEP ) - 100.0 ) 54,55,55
  54 KOUNT= KOUNT - 1
  55 ROUND = 5.0 * ( 10.0 ** KOUNT )
     IF( FIRST + FINAL ) 56,57,57
  56 FINAL = - FIRST
  57 FINAL =  FINAL+ ROUND
      KOEFF = ALOG10(FINAL)
      IF( FINAL -1.0 ) 58,59,59
  58 KOEFF = KOEFF - 1
      NAME(111) = N6(13)
      INDEX1 = - KOEFF/10
      INDEX2 = - KOEFF -10 * INDEX1
      GOTO 60
  59 NAME(111) = N6(12)
```

```
      INDEX1 = KOEFF/10
      INDEX2 = KOEFF -10 * INDEX1
   60 NAME(112)= N6( INDEX1 + 1 )
      NAME(113) = N6( INDEX2 + 1 )
      DO 61 I = 1,110
   61 NAME(I) = N6(14)
      KOUNT = KOEFF - KOUNT
      IF( KOUNT - 8 ) 63,63,62
   62 KOUNT = 8
   63  KOEFF = KOUNT - KOEFF- 1
      ADD = 0.0
      DO 71 I = 1,11
     LEFT = ( I*10) - 9 + (( 9 - KOUNT)/2 )
      IF( FIRST +ADD ) 64,65,65
   64 NAME(LEFT - 1 ) = N6(13)
      VALUE= ( ROUND -ADD ) - FIRST
     GO TO 66
   65 VALUE = ( ROUND + ADD ) +FIRST
   66 INTEGR = VALUE * ( 10.0 **KOEFF )
   67 INDEX1 = 1 + ( INTEGR/M2(KOUNT))
      IF( INDEX1 - 11 ) 69,68,72
   68 INTEGR = INTEGR - 1
      GO TO 67
   69 NAME(LEFT) = N6(INDEX1)
      NAME( LEFT + 1 ) = N6(11)
    LEFT = LEFT + 2
     DO 70 J = 2,KOUNT
     INDEX1 = KOUNT - J + 1
      INDEX1 = 1 + (INTEGR/M2(INDEX1))-(10*(INTEGR/
     *     M2(INDEX1 + 1 )))
      NAME( LEFT ) = N6( INDEX1)
   70 LEFT = LEFT + 1
   71 ADD = ADD + STEP
   72 IF( LOCK)43,43,51
      END
```

APPENDIX B

SAMPO

Usage

Program SAMPO is self contained. The user needs only to supply data and indicate which options and calculations are desired. All parameters used are initialized; they can be reset at any time, and they will remain at the new values until further changed. The usage of the code is controlled by program control cards the order of which should follow the desired operation to be performed. For uniformity all program control cards have the same format. This implies that numbers that would ordinarily be integers must be put in decimal form; they will be converted into integers by the program.

The program control cards are written in FORMAT (A10, 6E10.0,A4). The first word, which identifies the function of the control card, is always started in card column 1 and is known as the codeword (CODEW). Up to six parameters (WHAT(I), I = 1, 6) follow. The meaning of these parameters is determined by CODEW. These parameters are placed in card columns 11-20, 21-30, 31-40, 41-50, 51-60, and 61-70, respectively, and must contain a decimal point. If nothing is placed in a parameter field the computer assigns a value of 0.0 for the respective WHAT(I). The last alphanumeric word, NAME

(i.e. the A4 format), must appear in card columns 71-74. It may be used to identify the particular set of experimental data with which it is associated. It is not used by the program except for output purposes. The codewords, their functions and associated parameters are listed in the following section entitled Codewords.

Data can be read from cards in any desired format, and eventually a magnetic tape reading subroutine will be inserted. For rapid, essentially automatic analysis of relatively simple spectra, a peak-search algorithm scans the raw data to locate statistically significant events (peaks). The program then performs peak-shape, energy, and intensity calibrations, fits the peaks found in the spectrum (FITDO), and provides complete statistical and calibration error estimates in summarizing the results.

The automatic peak-fitting routine (FITDO) will select intervals such that up to five peaks may be fitted simultaneously. The user controlled fitting routine (FITS) will accept multiplets containing up to six components in a single fitting interval.

The qualitative limitations of the code arise, as might be expected, from the form and statistical quality of the data. The user quickly becomes familiar with the situations the program will and will not handle.

Typical Analysis

At this point it is desirable to illustrate how a user might proceed in carrying out the analysis of a typical spectrum using the SAMPO code.

First Run

The objective of the first run is to determine peak shape parameters and locations of the peaks that will be sensed by the automatic peak-finding routine. The fitting intervals that will later be selected by the program will be obtained so that the user can avert possible problems that may arise when the actual fitting is performed in the second run. The program data cards appropriate to this run are as given in Figure 6.

Col. 1	Col. 11	Col. 21	Col. 31	Col. 41	Col. 51	Col. 61
CODEWORD	WHAT(1)	WHAT(2)	WHAT(3)	WHAT(4)	WHAT(5)	WHAT(6)
OPTIONS	2.					
DATAIN	2.					
(12x,6(F8.0))						
5/69	RUN TITLE(Col. 7-64)			NUMBER OF CHANNELS(Col. 76)		1024.
SHAPEDO	110.	90.	120.		1.	
SHAPEDO	722.	708.	741.			
PEAKFIND						
FITTEST						
CALDATA	2.					
STOP						

Fig. 6--Data cards for the first run

The user has now obtained standard peak shapes for the spectrum, as well as an indication of how the program will try to analyze the data. With this information the user can decide exactly how he believes the data should be analyzed in the second run.

Second Run

The second run consists, basically, of the automatic fitting of the photopeaks found in the spectrum by the peak-search algorithm. For the present considerations it will be assumed that the user has obtained energy calibration data (from SHAPEDO or previously done FITDO centroids), as well as efficiency calibration data. This being the case, the analysis would proceed as in Figure 7.

After the analysis the data are now fitted, and may be inspected for regions where the analysis is not satisfactory. The punched card output allows the user to carry out any further manipulations with regard to energy or efficiency calibration at a later time, without having to repeat the time consiming fitting process. This is a rather valuable safeguard against possible calibration errors.

Third Run

The user may now wish to "clean-up" one or more of the unsatisfactory fits. Also, the user may want to change the energy calibration, examine a polynomial fit to the energy or efficiency calibration, normalize the relative intensities

Col. 1	Col. 11	Col. 21	Col. 31	Col. 41	Col. 51	Col. 61
CODEWORD	WHAT(1)	WHAT(2)	WHAT(3)	WHAT(4)	WHAT(5)	WHAT(6)
OPTIONS	2.	2.	2.			
DATAIN	2.					
(12x,6(F8.0))						
5/69	RUN TITLE(Col. 7-64)		NUMBER OF CHANNELS(Col. 76)			1024.
SHAPEIN	100.	3.5	4.7	1.8	1.	
SHAPEIN	500.	3.8	4.9	2.1		
SHAPEIN	650.	3.9	4.95	2.2		
SHAPEIN	800.	4.0	5.0	2.3		
SHAPEIN	1000.	4.2	5.1	2.4		
ENIN	109.952	88.35	0.05		1.	
ENIN	352.374	175.42	0.10			
ENIN	512.402	242.67	0.15			
EFIN	100.	.0324	5.		1.	
EFIN	500.	.0116	5.			
EFIN	1000.	.0032	5.			
PEAKFIND						
PEAKADD	84.	233.	524.	528.		
PEAKDROP	526.					
PEAKLIST						
FITDO						
CALDATA	2.	2.	2.			
RESULT				1.		
STOP						

Fig. 7--Data cards for the second run

to a specific gamma ray, or use any one of the other special options provided in the code.

Col. 1	Col. 11	Col. 21	Col. 31	Col. 41	Col. 51	Col. 61
CODEWORD	WHAT(1)	WHAT(2)	WHAT(3)	WHAT(4)	WHAT(5)	WHAT(6)
OPTIONS	2.	2.	2.			
DATAIN	2.					
(12x,6(F8.0))						
5/69	RUN TITLE(Col. 7-64)		NUMBER OF CHANNELS(Col. 76)		1024.	
SHAPEIN	100.	3.5	4.7	1.8	1.	
SHAPEIN	500.	3.8	4.9	2.1		
SHAPEIN	650.	3.9	4.95	2.2		
SHAPEIN	800.	4.0	5.0	2.4		
SHAPEIN	1000.	4.2	5.1	2.5		
ENIN	109.952	88.35	0.05		1.	
ENIN	352.374	175.42	0.10			
ENIN	512.402	242.67	0.15			
EFIN	100.	.0324	5.		1.	
EFIN	500.	.0116	5.			
EFIN	100.	.0032	5.			
FITS	2.			1.		
	3 507 545	519 525 529				
	6 802 865	820 828 837 844 847 854				
CALDATA	2.	2.	2.			
RESULT			1.			
STOP						

Fig. 8--Data cards for the third run

The data cards used in the "clean-up" run might appear as those in Figure 8. At the completion of the third run, the analysis should be complete.

Note that the analysis proceeds in a completely logical fashion. Although the program will recognize any legal data input card, it obviously cannot perform a SHAPEDO function until the spectral data has been read in. Similarly, the FITDO function must be preceded by the PEAKFIND function so that the program knows the approximate locations of the peaks it is being asked to fit. In general, the program will ignore any unreasonable request, and it will proceed to the next codeword data card.

Options

Most codewords have associated with them a series of options (i.e. WHAT(I), I = 1, 6). It is up to the user to select the particular option which is applicable to the type of analysis he is doing. The correct option is transmitted to the computer by assigning a given value and card position to each.

Codewords

There are twenty-seven codewords within the SAMPO program. Their function, as has been shown in Figures 6-8, is to allow the user to specify exactly which procedure is to be carried out on the data. Below is a list of codewords with their associated options.

CALDATA.--This codeword prints the data used for shape, energy, and efficiency calibration. Its options are:

WHAT(1) Specifies shape calibration requested.

= 0. Print no data

= 1. Print arrays stored

= 2. Print arrays read in or generated

WHAT(2) Specifies energy calibration requested.

= 0. Print no data

= 1. Print arrays stored for linear interpolation

= 2. Print arrays read in or generated for linear interpolation

= 3. Print coefficients stored for polynomial curve

= 4. Print coefficients read in or generated

WHAT(3) Specifies efficiency calibration requested.

= 0. Print no data

= 1. Print arrays stored for linear interpolation

= 2. Print arrays read in or generated for linear interpolation

= 3. Print parameters of the functional fit stored

= 4. Print parameters read in or generated for functional fit

COMMENTS.--This codeword reads comments that are on the following data card. These comments are printed out with the fitting results and tables. The COMMENT storage area is initialized with blank spaces.

DATAGRAPH.--This codeword makes a printer graph of any portion of the spectral data. Its options are:

WHAT(1) Type of graph desired.

= 0. Semilogarithmic scale

≠ 0. Linear scale

WHAT(2) Which channels to plot.

= 0. Plot every channel

≠ 0. Plot every WHAT(2) channel

WHAT(3) Starting channel for the plot.

= 0. The first channel

≠ 0. Start with channel number WHAT(3)

WHAT(4) Stopping channel for the plot.

= 0. The last channel

≠ 0. Stop with channel number WHAT(4)

WHAT(5) Channels in each subdivision.

= 0. Every 100 channels

≠ 0. Every WHAT(5) number of channels

DATAIN.--This codeword reads in the indicative and spectral data. The card that follows the DATAIN card must have a run number in columns 1-6, alphanumeric indicative data in columns 7-66, numeric indicative data in columns 67-70 and 71-75, and the number of channels of data in columns 76-80. The options for this codeword are:

WHAT(1) Type of input.

= 0. Search and read in the spectrum name from an input tape (not implemented at present)

= 1. Card input, spectral data in FORMAT(10F7.0)

= 2. Card input, spectral data in FORMAT specified on the next data card

WHAT(2) Disposition of old spectrum

= 0. Erase previous spectrum

≠ 0. Add previous spectrum multiplied by WHAT(2) to the new spectrum

WHAT(3) Manipulation of new spectrum

= 0. Nothing

≠ 0. Multiply new spectrum by WHAT(3) and add it to the previous spectrum

EFDO.--This codeword assigns efficiency calibration values for linear interpolation to a peak fitted using FITDO or FITS. This codewords options are:

WHAT(1) Approximate energy of the point; accuracy tolerance of 3 KeV

WHAT(2) Number of disintegrations corresponding to the WHAT(1) energy peak

WHAT(3) Calibration error in per cent

WHAT(4) Change accuracy tolerance to WHAT(4)

WHAT(5) Disposition of old calibration table

= 0. Add values to the existing efficiency calibration table

= 1. Start a new efficiency calibration table with the value on this data card

EFFITADJ.--This codeword adjusts the coefficient C_1 in the functional form of the fit to the efficiency data (fit generated in codeword EFFITDO). The parameters of this codeword are:

WHAT(1) Energy of adjustment point.

WHAT(2) Efficiency at the WHAT(1) energy point.

WHAT(3) Nothing.

WHAT(4) Parameters to be adjusted.

= 1. Adjust parameters stored

= 2. Adjust parameters generated

EFFITDO.--This codeword performs a functional fit to the points read in under EFDO or EFIN. The resulting curve may be used for subsequent energy determination if specified on the OPTIONS card. Calibration error will be the same as in linear interpolation. The functional form used is

$$\text{Efficiency} = C_1 \cdot (E^{C_2} + C_3 \cdot e^{C_4 \cdot E}) \quad ,$$

where C_1, C_2, C_3, and C_4 are parameters to be determined and E is the energy in KeV. The one option for this codeword is:

WHAT(1) Initial guess for parameter C_1.

EFFITIN.--This codeword reads in parameters from a previous functional fit to the efficiency data. Its options are:

WHAT(1) Energy value for calibration error point in KeV.

WHAT(2) Value of the parameter.

WHAT(3) Calibration error in per cent

WHAT(4) The index of the parameter (i.e. 1, 2, 3, or 4)

WHAT(5) Disposition of the old table

= 0. Add calibration error pair to existing table

= 1. Start a new calibration error table with these values

EFIN.--This codeword reads in the efficiency calibration points to be used through linear interpolation. The options here are:

WHAT(1) Energy of the calibration point.

WHAT(2) Efficiency of the calibration point.

WHAT(3) Calibration error in per cent

WHAT(4) Nothing.

WHAT(5) Disposition of the old table.

= 0. Add these values to the existing table

= 1. Start a new table with these values

ENDO.--This codeword assigns an energy calibration value for linear interpolation to a peak fitted under FITDO or FITS. The options for ENDO are:

WHAT(1) Approximate channel location, accuracy tolerance of 3 channels or as specified by WHAT(4).

WHAT(2) Energy in KeV.

WHAT(3) Calibration error in KeV.

WHAT(4) Assignment of accuracy tolerance.

= 0. Use the previous accuracy tolerance, initialized at 3.

≠ 0. Change the accuracy tolerance to WHAT(4).

WHAT(5) Disposition of the old table.

= 0. Add these values to existing energy calibration table.

= 1. Start a new table with these values.

ENFITDO.--This codeword performs a polynomial fit to the points read in under ENDO or ENIN. The resulting curve may be used for subsequent energy determinations if it was specified on the OPTIONS card. Calibration error will be the same as in linear interpolation. The option for this codeword is:

WHAT(1) The number of terms in the polynomial fit (i.e. 1 + degree of the polynomial).

ENFITIN.--This codeword reads in polynomial coefficients and calibration error points. Its options are:

WHAT(1) Value of the polynomial coefficient.

WHAT(2) Energy value for the calibration error point.

WHAT(3) Calibration error in KeV.

WHAT(4) Disposition of the old coefficient table.

= 1. Start a new coefficient table.

≠ 1. The number of the coefficient (i.e. 1 +

degree) to be added to this table.

WHAT(5) Disposition of the old energy table.

= 0. Add energy calibration error point to the existing list.

= 1. Start a new energy calibration error list.

ENIN.--This codeword reads in energy calibration points to be used through linear interpolation (see OPTIONS). The options for ENIN are:

WHAT(1) Exact channel location.

WHAT(2) Energy of channel WHAT(1) in KeV.

WHAT(3) Calibration error in KeV.

WHAT(4) Nothing.

WHAT(5) Disposition of the old table.

= 0. Add values to existing energy calibration table.

= 1. Start new table with these values.

FITDO.--This codeword performs complete nonlinear fits of peaks found by the peak search algorithm. Selection of the fitting intervals and the peaks included in each fit is done automatically as in FITTEST. The results include the exact channel locations, peak areas, energies, intensities, and statistical and calibration errors for each peak. The energy and efficiency calibration is performed as specified on the OPTIONS card which must come somewhere before this codeword.

FITIN.--This codeword reads in fitting results from punched cards. The FORMAT is the same as that used in punching results. The FITIN card is followed by WHAT(1) number of cards, which have the energy values in columns 1-8, the exact channel locations in columns 15-23, the energy errors in columns 30-35, the peak areas in columns 36-46, the peak intensities in columns 53-63, the intensity errors in columns 69-74, and the run number in columns 75-80. The options for FITIN are:

WHAT(1) Number of peaks to be read in.

WHAT(2) Nothing.

WHAT(3) Nothing.

WHAT(4) Nothing.

WHAT(5) Disposition of the old table.

= 0. Add the results to the existing list of fitted peaks.

= 1. Start a new list of fitted peaks.

FITREPEAT.--This codeword repeats the cases specified by the last FITS card. Only channel numbers read in under FORMAT (16I5) were stored, not the special weighting or initial guesses. WHAT(2), WHAT(3), and WHAT(5) have the same meaning as under FITS.

FITS.--This codeword performs complete nonlinear fits of peaks specified on the cards following. The computation is the same as performed under FITDO, only complete control

over the selection of intervals and peaks is possible. Special weighting and/or initial guesses for peak amplitudes may be used. If special weightings (i.e. WHAT(2) = 2.) or initial guesses (i.e. WHAT(3) = 1.) are specified, they must follow each case in FORMAT (8E10.0). Up to 200 numbers may be specified in FORMAT (16I5) and stored to be reused under FITREPEAT. The FITS card is followed by cards specifying WHAT(1) number of cases in FORMAT (16I5). The number of peaks in the fit is in columns 1-5, the channel defining the lower limit of fitting interval is in columns 6-10, and the channel defining the higher limit of fitting interval is in columns 11-15. The centroid of the parabolic correction in the background polynomial is in columns 16-20. If columns 16-20 are blank, the centroid is assigned to the peak channel for single peaks and the centroid of the fitting interval for multiplets. The channel in the first peak in the fit is in columns 21-25, and the other peaks in the fit are in columns 26-80. The options for FITS are:

WHAT(1) Number of special cases specified by the following WHAT(1) cards.

WHAT(2) Weighting factor.

= 0. Normal weighting, that is 1/YDATA.

= 1. Special weighting, $1/YDATA^2$.

= 2. Special weighting as read in for each data point as specified in FORMAT (8E10.0).

WHAT(3) Initial guess.

= 0. Automatic initial guess.

= 1. Initial guess read in for each peak amplitude in FORMAT (3E10.0) after the card specifying the channels.

WHAT(4) Nothing.

WHAT(5) Disposition of the old table.

= 0. Add the results to existing list of fitted peaks.

= 1. Start a new list of fitted peaks with these results.

FITTEST.--This codeword determines and prints the channels limiting the fitting intervals and the peaks included for each fit. The fitting is not performed under this codeword.

OPTIONS.--This codeword specifies internal calibrations and options. The options here are:

WHAT(1) Specifies shape calibration, initialized at 2.

= 1. Use arrays stored in.

= 2. Use arrays generated or read in.

WHAT(2) Specifies energy calibration, initialized at 0.

= 0. No energy calibration.

= 1. Use arrays stored for linear interpolation.

= 2. Use arrays generated or read in for linear interpolation.

= 3. Use polynomial stored.

= 4. Use polynomial generated or read in.

WHAT(3) Specifies efficiency calibration, initialized at 0.

= 0. No efficiency calibration.

= 1. Use arrays stored for linear interpolation.

= 2. Use arrays generated or read in for linear interpolation.

= 3. Use function stored.

= 4. Use function generated or read in.

WHAT(4) Specifies when linear background instead of parabolic is used. Whenever the fitting interval measured in units of the CW parameter remains smaller than a specified parameter, initialized at 15., only a linear background is used.

= 0. Use existing value for this parameter.

≠ 0. Change the parameter to WHAT(4).

PEAKADD.--This codeword adds peaks to the list of peaks found. This option is used to correct the list of peaks found during the second run. The search routine (PEAKFIND) should be done before this codeword. The options for PEAKADD are:

WHAT(I), I=1,6 The channel numbers of the peaks to be added, in any order.

PEAKDROP.--This codeword drops peaks from the list of peaks found. See comments regarding PEAKADD. The options for PEAKDROP are:

WHAT(I), I=1,6 The channel numbers of the peaks to be dropped in any order.

PEAKFIND.--This codeword searches and lists statistically significant peaks. The options for this codeword are:

WHAT(1) Starting channel.

= 0. Starts with channel number 50.

≠ 0. Start with channel number WHAT(1).

WHAT(2) Stopping channel.

= 0. Stop with the last channel in the spectrum.

≠ 0. Stop with channel number WHAT(2).

WHAT(3) Significance limit for listing peaks.

= 0. Use existing value (initialized at 2).

≠ 0. Change the significance limit to WHAT(3).

WHAT(4) Significance limit for fitting peaks.

= 0. Use existing value (initialized at 4).

≠ 0. Change the significance limit to WHAT(4).

WHAT(5) CW parameter (FWHM = 2.355CW).

= 0. Use existing value obtained in the shape calibration.

≠ 0. Change the CW parameter to WHAT(5).

PEAKLIST.--This codeword prints the corrected list of peaks found.

RESET.--This codeword resets the arrays from peak search and peak fitting to zero. Its options are:

WHAT(1) Peak search array.

= 0. Nothing.

= 1. Reset the peak search array from PEAKFIND.

WHAT(2) Result array.

= 0. Nothing.

= 1. Reset the result array from peak fitting.

RESULT.--This codeword tabulates the fitting results accumulated and computes relative intensities in the analysis of the spectrum. The options for RESULT are:

WHAT(1) Approximate channel of the reference peak for relative intensity calculations.

= 0. Use the largest peak as reference.

≠ 0. Use the peak in channel WHAT(1) as reference for relative intensity calculations.

WHAT(2) Energy determination.

= 0. No new energy determination.

≠ 0. New energy determination using OPTION as specified by WHAT(2) (see OPTIONS).

WHAT(3) Intensity determination.

= 0. No new intensity determination.

≠ 0. New intensity determination using OPTION. as specified by WHAT(3) (see OPTION).

WHAT(4) Punched output.

= 0. No punched output requested.

= 1. Place results on punched cards.

WHAT(5) Counting time for the spectrum (used for absolute intensity measurements).

SHAPEDO.--This codeword performs shape calibration fits. This should be done using strong single lines in the spectrum under analysis or in a spectrum measured under identical conditions. The resulting shape parameters will be stored and interpolated to fit all peaks in the spectrum. These parameters need to be computed only once for each experimental set-up; the values will remain unchanged till specified, and can be read in rather than recomputed for later runs. In specifying fitting interval include enough channels to cover both tails, but do not over extend or include fluctuations due to other peaks. The options for SHAPEDO are:

WHAT(1) Center channel of the calibration peak.

WHAT(2) Lower limit of fitting interval.

WHAT(3) Higher limit of fitting interval.

WHAT(4) Additional calibrations.

= 0. Nothing.

= 1. Use this peak for energy calibration by adding ENDO card with energy and calibration error after SHAPEDO card.

= 2. Use this peak for efficiency calibration by adding EFDO card with the peak area and the calibration error.

= 3. Energy and efficiency calibration as above.

WHAT(5) Disposition of the old table.

= 0. Add resulting shape parameters to the present table of parameters.

= 1. Start new shape parameter table with the resulting parameters.

WHAT(6) Initial guess for the CW parameter.

= 0. Set the CW parameter to 1.8.

≠ 0. Set the CW parameter to WHAT(6).

SHAPEIN.--This codeword reads in and stores shape parameters which were computed in a previous run. The options for this codeword are:

WHAT(1) Channel of the peak center.

WHAT(2) CL parameter, specifies the channel number where the lower tail stops (CL = the number of channels from the peak centroid where the exponential tail is joined to the gaussian on the low energy side).

WHAT(3) CH parameter, specifies the channel number where the higher tail stops (CH = the number of channels from the peak centroid where the exponential tail is joined to the gaussian on the high energy side).

WHAT(4) CW parameter, specifies the peak width.

WHAT(5) Disposition of the old table.

= 0. Add these values to the present table.

= 1. Start a new table with these values.

STOP.--This codeword terminates computations and exits the program from the machine.

Comments

The following is a list of observations, which are helpful to the beginning user of program SAMPO:

1. The number of calibration fits required depends upon the total energy range of the spectrum. If enough strong, single lines are available one should generally use one for about each interval of 100 to 200 KeV in the spectrum.

2. The shape parameters need to be calculated only once for each experimental set-up.

3. It should be noted that whenever the tailing parameter becomes large compared with the width parameter, its effect becomes negligible and its value need not be accurate.

4. For each fit a region in the spectrum must be specified. This can be done either by the user (FITS) or by an automatic algorithm (FITDO). The procedures used in the fitting are identical in both cases.

5. The standard deviation of the peak height, expressed in per cent of the peak height, is taken to be the uncertainty in the peak area.

6. In selection of fitting intervals, found "automatically" (FITDO), represented by less than a given separation parameter, generally six times the full width half maximum, are fitted together.

7. Because the quality of the fit is tested using a chi-square convergence criterion, it is found that very weak lines in the vicinity of much stronger photopeaks are often overlooked by the fitting algorithm, and even if the user explicitly specifies the locations and intensities of the lines in question (FITS), a satisfactory fit to the data is sometimes obtained only after repeated "juggling" of the fitting parameters, if indeed it is obtained at all.

8. The parameter CW, used throughout the program, is simply an easier computational form of the full width half maximum.

9. In checking the fits output by the code if it is noticed that the computed background seems to be partially "sucked up" into the peak, generally this means that there is a small peak overlapped on the larger peak which was not found during the PEAKFIND routine.

10. Using measured spectral information, the authors generated test cases for analyzing single peaks of varying intensities on a Compton plateau and Compton edge of average 2000 counts per channel. The results are summarized in the following table. For the peak on the Compton plateau errors found are less than two per cent down to a peak-height -to-continuum greater than 1.0. The error estimates indicated in the table reflect the deviations of the calculated intensity from the known value for the photopeak on the Compton continuum.

TABLE II

COMPARISON OF TRUE DATA
TO SAMPO OUTPUT

Peak height / Compton plateau	SAMPO intensity / True intensity	Peak height / Compton edge	SAMPO intensity / True intensity
0.035	1.246 ± 0.267	0.03	2.256 ± 0.037
0.07	1.084 ± 0.139	0.06	1.588 ± 0.157
0.14	1.068 ± 0.072	0.12	1.329 ± 0.079
0.35	1.009 ± 0.031	0.3	1.120 ± 0.033
0.7	1.014 ± 0.016	0.6	1.078 ± 0.017
1.4	1.001 ± 0.009	1.2	1.038 ± 0.009
3.5	1.000 ± 0.006	3.0	1.023 ± 0.005
7.0	0.999 ± 0.005	6.0	1.015 ± 0.005
14.0	1.007 ± 0.004	12.0	1.021 ± 0.005
35.0	0.998 ± 0.005	30.0	1.011 ± 0.005

11. Check your results table closely. If the same peak was fitted twice (i.e., under FITDO and later under FITS) it is quite possible that both values calculated for this peak will be in the result table.

REFERENCES

1. S. K. Mangal and P. S. Gill, Nucl. Phys. 49, 510 (1963).
2. W. Lu, N. RanaKumar, and R. W. Fink, Phys. Rev. C 1, 350 (1970).
3. B. Minetti and A. Pasquarelli, Z. Physik 207, 132 (1967).
4. G. C. Bonazzola, P. Brovetto, E. Chiavassa, R. Spinoglio, and A. Pasquarelli, Nucl. Phys. 51, 337 (1964).
5. E. B. Paul and B. L. Clarke, Can. J. Phys. 31, 267 (1953).
6. W. Lu, N. RanaKumar, and R. W. Fink, Phys. Rev. C 1, 358 (1970).
7. S. G. Forbes, Phys. Rev. 88, 1309 (1952).
8. R. J. Prestwood and B. P. Bayhurst, Phys. Rev. 121, 1430 (1961).
9. J. M. Ferguson and W. E. Thompson, Phys. Rev. 118, 228 (1960).
10. A. Poularikas and R. W. Fink, Phys. Rev. 115, 989 (1952).
11. L. A. Rayburn, Phys. Rev. 122, 168 (1961).
12. J. E. Brolley, J. L. Fowler, and L. K. Schlacks, Phys. Rev. 88, 618 (1952).
13. Regional Nuclear Physics Laboratory, operated jointly by North Texas State University and Southern Methodist University.
14. S. A. Marsh, unpublished master's thesis, Department of Physics, North Texas State University, Denton, Texas, 1970.

15. J. T. Routti and S. G. Prussin, Nucl. Inst. and Methods 72, 125 (1969).

16. N. Bohr, Nature 137, 344 (1936).

17. N. Bohr, Science 86, 161 (1937).

18. J. M. Blatt and V. F. Weisskopf, Theoretical Nuclear Physics (John Wiley and Sons, New York, 1952).

19. E. Segre, Experimental Nuclear Physics (John Wiley and Sons, New York, 1953).

20. I. E. McCarthy, Introduction to Nuclear Theory (John Wiley and Sons, New York, 1968).

21. H. V. Buttlar, Nuclear Physics an Introduction (Academic Press, New York, 1968).

22. R. D. Evans, The Atomic Nucleus (McGraw-Hill, New York, 1955).

23. I. Kaplan, Nuclear Physics (Addison-Wesley, Reading Massachusetts, 1958).

24. P. Cuzzocrea, S. Notarrigo and E. Perillo, Nuovo Cimento 52B, 476 (1967).

25. A. G. W. Cameron, Can. J. Phys. 36, 1040 (1958).

26. J. Dostrovsky and Z. Fraenkel, Phys. Rev. 116, 683 (1959).

27. J. H. E. Mattauch and W. Thiele, Nucl. Phys. 67, 32 (1965).

28. S. Pearlstein, Nuclear Data 3, 327 (1967).

29. C. M. Lederer, J. M. Hollander, and I. Perlman, Table of Isotopes (John Wiley and Sons, New York, 1967).

30. J. A. Pinston, F. Schussler, and A. Moussa, Nucl. Phys. A133, 124 (1969).