BNL 50199 (T-549)

220
5-5-70

1354

MOGUS – A CODE FOR EVALUATING THE MOTT SCATTERING CROSS SECTION AND THE GOUDSMIT-SAUNDERSON ANGULAR MULTIPLE-SCATTERING DISTRIBUTION FOR USE IN ELECTRON TRANSPORT CALCULATIONS

R.M. FELDER

July 1969

BROOKHAVEN NATIONAL LABORATORY
ASSOCIATED UNIVERSITIES, INC.
under contract with the
UNITED STATES ATOMIC ENERGY COMMISSION

BNL 50199 (T-549)
(Mathematics and Computers - TID-4500)

MOGUS - A CODE FOR EVALUATING THE MOTT SCATTERING CROSS SECTION AND THE GOUDSMIT-SAUNDERSON ANGULAR MULTIPLE-SCATTERING DISTRIBUTION FOR USE IN ELECTRON TRANSPORT CALCULATIONS

R.M. FELDER*

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

July 1969

*Present address: Department of Chemical Engineering, North Carolina State University, Raleigh, North Carolina 27607

BROOKHAVEN NATIONAL LABORATORY
UPTON, NEW YORK 11973

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.

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 employ ment or contract with the Commission, or his employment with such contractor

Printed in the United States of America
Available from
Clearinghouse for Federal Scientific and Technical Information
National Bureau of Standards, U S Department of Commerce
Springfield, Virginia 22151
Price Printed Copy $3 00, Microfiche $0 65

March 1970 575 copies

ABSTRACT

The formulas required for the calculation of the Mott cross section for scattering of electrons by an unscreened nuclear charge and the Goudsmit-Saunderson angular multiple-scattering distribution for electrons are summarized, and a FORTRAN-IV code which performs these calculations is given. The occurrence of the calculated quantities in electron transport theory is outlined in an introductory section. Subsequent sections contain a detailed summary of the transport model and analytical expressions for the calculated quantities, an outline of the code, a guide to the use of the code, illustrative results, and the code itself.

Table of Contents

I. INTRODUCTION: DEFINITIONS AND ASSUMPTIONS

Consider an energetic electron which originates at energy E_o and slows down in an infinite homogeneous medium. A coordinate system is established such that the electron is produced at the origin and proceeds along a trajectory initially tangent to the z axis. The direction of flight at a point on the trajectory is characterized by ω and ϕ, the polar and azimuthal angles respectively (relative to the z axis) of the tangent to the trajectory, or alternately by $\mu = \cos\omega$ and ϕ.

A dimensionless energy τ is defined as the ratio of the electron's kinetic energy E to its rest mass energy, or

$$\tau = E\ (\mathrm{MeV})/0.510976\ . \tag{1}$$

Quantities needed to describe the slowing-down process are

$s\ (\tau_o,\tau)$ path length, or total distance traveled by the electron in slowing down from τ_o to τ.

$s_t(\tau_o)$ $s(\tau,0)$, total path length for slowing down from τ to (in effect) 0.

$t(\tau_o,\tau)$ fraction of the total path length $s_t(\tau_o)$ which an electron has yet to travel when its energy is τ. By definition

$$t(\tau_o,\tau) = [s_t(\tau_o)-s(\tau_o,\tau)]/s_t(\tau_o)\ . \tag{2}$$

$A(\mu,t)d\mu$ $[A(\mu,s)d\mu]$ probability that when the fractional residual path length of an electron equals t (or the path length traversed equals s) the polar direction cosine lies within $d\mu$ of μ. Azimuthal symmetry is assumed.

A is normalized by the relation

$$\int_{-1}^{1} A(\mu,t)d\mu = \int_{0}^{\pi} A(\omega,t)\sin\,\omega d\omega = 1 \quad , \tag{3}$$

and hence, since the initial direction cosine is $\mu = 1$,

$$A(\mu,t{=}1) = \delta(\mu-1) \quad . \tag{4}$$

In general, electron slowing-down problems cannot be solved exactly; approximations must be made regarding the precise functionalities of $s(\tau_0,\tau)$ and $A(\mu,t)$, and for all but the simplest models numerical computations of one kind or another are required. Descriptions of models that have been adopted for various transport calculations are given in review articles by Berger[1] and Zerby and Keller.[2] A common model involves the use of the continuous-slowing-down approximation for energy loss calculations, so that a unique relationship between energy and path length is prescribed, and the choice of a multiple scattering formula for $A(\mu,t)$, of which the most accurate currently known is that of Goudsmit and Saunderson.[3]

Lateral spatial deflections are generally ignored in the absence of a satisfactory way to handle them, and one of various available empirical or semi-empirical relations between range and energy is customarily invoked to correlate distance traveled in a specific direction with total distance traveled.

A code has been written which sets up the prerequisites for numerical transport calculations based on the model just described. For a given set of system parameters, the code generates a table of path length s and fractional residual path length t vs. dimensionless energy τ, and performs calculations which prepare for the subsequent generation of the Goudsmit-Saunderson distribution. Thereafter, if a range decrement r and initial energy E_o are specified, the program calculates the final energy E_f from an empirical relation due to Katz and Penfold,[4] interpolates in the previously generated t–E table to obtain the fractional residual path length $t(E_f)$, and generates the Goudsmit-Saunderson distribution $A(\mu, t)$ for specified values of the deflection angle $\omega = \cos^{-1}\mu$. Alternately, the fractional residual path length itself may be specified and $A(\mu, t)$ calculated directly.

In the course of generating the Goudsmit-Saunderson distribution, it is necessary to calculate the Mott cross section for scattering of electrons by an unscreened nuclear charge.[5] This calculation is performed by a routine which is called internally in the angular distribution calculation, and may also be called externally if values of this cross section are separately desired.

In Section II the analytical expressions which form the basis of these calculations are presented. Section III contains an outline of the code, indicating where and how the code uses the material in Section II. Section IV contains a user's guide to the code, including required statements in the calling program, core storage requirements, and descriptions of additional routines which the user must supply. Section V contains illustrative results generated by the code and estimated running times, and Section VI contains the code itself and sample calling programs.

II. OUTLINE OF THE MODEL

A. Path Length vs. Energy

At the point of origin of an energetic electron, the dimensionless energy τ and the path length s equal τ_o and 0, respectively. τ and s are related in the continuous-slowing-down approximation by

$$\tau(s) = \tau_o - \int_0^s |d\tau/ds'| ds' \quad , \tag{5}$$

and

$$s(\tau_o, \tau) = \int_\tau^{\tau_o} |d\tau'/ds|^{-1} d\tau' \quad . \tag{6}$$

An expression for the stopping power $(d\tau/ds)_c$ due to collisions with atomic electrons has been derived, based on the Møller free electron scattering cross section for large energy transfers and a Born approximation formula due to Bethe for small transfers[6,7]:

$$-(d\tau/ds)_c(\text{cm}^{-1}) = 0.301314 \frac{\rho Z}{W} \frac{(\tau+1)^2}{\tau(\tau+2)} [\ln \frac{0.130548\, \tau^2(\tau+2)}{I^2} + f^-(\tau)$$

$$- \delta(\tau, Z) - C_s(\tau, Z)] \quad , \tag{7}$$

$$f^-(\tau) = \frac{1 + \tau^2/8 - (2\tau+1)\ln 2}{(\tau+1)^2} \quad , \tag{8}$$

where ρ, Z, and W denote the density (g/cm^3), atomic number, and atomic weight, respectively, of the stopping medium, I (MeV) is the so-called mean ionization potential, δ is a correction for the mean energy loss reduction due to polarization of the stopping medium by the energetic electrons, and C_s accounts for effects which occur when the incident electron velocity approximately equals the electron orbital velocity in a given shell. The numerical constants of Eq. (7) represent the following combinations of physical and natural constants:

$$0.301314 \approx \frac{2\pi e^4 N_{av}}{(m_o c^2)^2} \quad ,$$

$$0.13058 \approx \frac{1}{2}(m_o c^2)^2 \quad ,$$

where e is the electronic charge, N_{av} is Avogadro's number, and $m_o c^2$ is the electron rest mass energy.

Caldwell[8] has determined values of I for several materials, and Sternheimer[9] has calculated values of δ based on these I values. Results of both sets of calculations are tabulated by Nelms.[10] Huson[7] has obtained an empirical expression for the shell correction C_s by fitting experimental data:

$$C_s(\tau, Z) = \frac{Z^{\frac{1}{2}}}{x+a}\left(1 - \frac{b}{x^2}\right) \quad , \tag{9}$$

$$x = \frac{(137)^2 \tau(\tau+2)}{(\tau+1)^2 Z} \quad , \tag{9a}$$

$$a = \frac{Z^2(1 + 2/Z)}{1 + Z^2/30} \quad , \tag{9b}$$

$$b = \frac{36(Z^3 + 200)}{(10+Z)(Z^3 + 3000)} \quad . \tag{9c}$$

At higher energies, bremsstrahlung production makes an increasingly important contribution to the mean energy loss. The total stopping power due to both radiative and collision losses is approximately given by the formula[11]

$$-(d\tau/ds) = -(d\tau/ds)_c[1 + (Z\tau/1400)], \quad Z\tau \ll 1400 \ . \tag{10}$$

The validity of Eq. (10) is discussed in Section III. F.

B. Range vs. Energy

Katz and Penfold[4] have proposed empirical range-energy relations based on a compilation of experimental thick absorber transmission data. These relations, in terms of r_t (cm) (defined in Section I), E (MeV), and ρ (g/cm^3) are as follows:

1. $0.01 \leq E \leq 3$ MeV

$$r_t(E) = 0.412E^n/\rho \tag{11}$$

$$n = 1.265 - 0.0954 \ln E \ ;$$

$$E(r_t) = \exp\left\{6.629979035639 - 5.241090146751[1.261848095652 - 0.3816 \ln(\rho r_t)]^{\frac{1}{2}}\right\} . \tag{12}$$

2. $1 \leqslant E \leqslant 20$ MeV

$$r_t(E) = \rho^{-1}(0.530E - 0.106) , \tag{13}$$

$$E(r_t) = 1.886792453\, \rho r_t + 0.2 . \tag{14}$$

C. Mott Scattering Cross Section

The ratio of the Mott scattering cross section $\sigma_m(\tau, \theta)$ for scattering of electrons by an unscreened nuclear charge to the Rutherford cross section $\sigma_r(\tau, \theta)$ occurs as a parameter of the Goudsmit-Saunderson distribution. The latter cross section is (in cm^2)

$$\sigma_r(\tau, \theta) = \frac{7.961759 \times 10^{-26}\, Z^2(\tau+1)^2}{\tau^2(\tau+2)^2(1-\cos\theta)^2} \tag{15}$$

where

$$7.961759 \times 10^{-26} \approx e^4/(m_o c^2)^2 .$$

By definition, $2\pi\sigma_r \sin\theta d\theta$ is the cross section for scattering within $d\theta$ of the polar scattering angle θ.

The Mott cross section is given by[5]

$$\sigma_m(\tau, \theta) = \frac{7.944723 \times 10^{-26}\, Z^2}{\tau^2(\tau+2)^2} | F_o+F_1|^2 \csc^2(\theta/2) + \frac{1.491145 \times 10^{-21}}{\tau(\tau+2)} |G_o+G_1|^2 \sec^2(\theta/2) , \tag{16}$$

where

$$1.491145 \times 10^{-21} \approx (\hbar/m_o c)^2 \ ,$$

$$7.944723 \times 10^{-26} \approx 1.491145 \times 10^{-21}/(137)^2 \ .$$

The quantities F_o, F_1, G_o, and G_1 in terms of

$$q = \frac{Z}{137} \frac{(\tau+1)}{[\tau(\tau+2)]^{\frac{1}{2}}} \tag{17}$$

are

$$F_o = (i/2)\exp\left\{iq \ln[\sin^2(\theta/2)]\right\}\Gamma(1-iq)/\Gamma(1+iq) \ , \tag{18}$$

$$G_o = -iqF_o \cot^2(\theta/2) \ , \tag{19}$$

$$F_1 = (i/2) \sum_{k=0}^{\infty} (-)^k [kD_k + (k+1)D_{k+1}]P_k(\cos\theta) \ , \tag{20}$$

$$G_1 = (i/2) \sum_{k=0}^{\infty} (-)^k [k^2 D_k - (k+1)^2 D_{k+1}]P_k(\cos\theta) \ , \tag{21}$$

where Γ is the gamma function and P_k the Legendre polynomial of order k.

The coefficients D_k are

$$D_k = \frac{e^{-i\pi k}}{k+iq} \frac{\Gamma(k-iq)}{\Gamma(k+iq)} - \frac{e^{-i\pi\rho_k}}{\rho_k+iq} \frac{\Gamma(\rho_k-iq)}{\Gamma(\rho_k+iq)} \ , \tag{22}$$

where

$$\rho_k = [k^2 - (Z/137)^2]^{\frac{1}{2}} \ . \tag{23}$$

The series for F_1 and G_1 converge quite slowly, so much so that direct summation is all but impossible, even on the most advanced computers. To facilitate convergence, Feshbach[12] has applied the Euler transformation[13] to Eqs. (20) and (21), and Sherman[14] has applied two consecutive transformations, the first due to Yennie, Ravenhall, and Wilson[15] and the second the Euler transformation. Calculations have shown, however, that the introduction of the second transformation does not improve convergence, and in fact often results in a considerable decrease in the convergence rate and hence an increase in computing time. The present code consequently omits the Euler transformation.

The Yennie, Ravenhall, and Wilson transformation is as follows. If a function $f(\theta)$ may be expressed as an infinite Legendre series,

$$f(\theta) = \sum_{k=0}^{\infty} a_k P_k(\cos\theta) \quad , \tag{24a}$$

then

$$(1-\cos\theta)^m f(\theta) = \sum_{k=0}^{\infty} a_k^{(m)} P_k(\cos\theta) \quad , \tag{24b}$$

where

$$a_k^{(i+1)} = a_k^{(i)} - \frac{k+1}{2k+3} a_{k+1}^{(i)} - \frac{k}{2k-1} a_{k-1}^{(i)} \quad . \tag{24c}$$

By definition, $a_k^{(0)} \equiv a_k$, and $a_{-1}^{(i)} \equiv 0$.

It has been found most efficient to apply this transformation to Eqs. (20) and (21) with $m = 3$ for $Z \leqq 50$, and $m = 4$ for $Z > 50$. The transformed equations are

$$F_1 = \tfrac{1}{2}i(1-\cos\theta)^{-m} \sum_{k=0}^{\infty} f_k^{(m)} P_k(\cos\theta)\ , \quad \theta > 0 \tag{25}$$

$$f_k = (-)^k[kD_k + (k+1)D_{k+1}]\ ; \tag{25a}$$

$$G_1 = \tfrac{1}{2}i(1-\cos\theta)^{-m} \sum_{k=0}^{\infty} g_k^{(m)} P_k(\cos\theta)\ , \quad \theta > 0\ , \tag{26}$$

$$g_k = (-)^k[k^2D_k - (k+1)^2D_{k+1}]\ . \tag{26a}$$

The coefficients $\{f_k^{(m)}\}$ and $\{g_k^{(m)}\}$ are generated from $\{f_k\}$ and $\{g_k\}$ respectively using (24c).

The calculational procedure is as follows. F_o and G_o are calculated from Eqs. (18) and (19). F_1 and G_1 are calculated from Eqs. (25) and (26), with D_k being given by (22), ρ_k by (23), and $m = 3$ ($Z \leqq 50$) or 4 ($Z > 50$). The cross section σ_m is calculated from (16).

The preceding calculational method clearly fails when $\theta = 0$, since both σ_m and σ_r are infinite at this angle; in addition, convergence is poor for small but nonzero values of θ, regardless of the transformation used. Fortunately, an analytical expression exists for σ_m/σ_r at small angles, namely[1]

$$\sigma_m/\sigma_r \approx 1 + (\pi\beta^2 q/\sqrt{2})\cos\gamma\,(1-\cos\theta)^{\frac{1}{2}}\ , \tag{27}$$

where

$$\beta^2 = \tau(\tau+2)/(\tau+1)^2\ , \tag{27a}$$

$$\cos\gamma = \mathrm{Re}\left\{ \frac{\Gamma(\frac{1}{2}-iq)}{\Gamma(\frac{1}{2}+iq)} \; \frac{\Gamma(1+iq)}{\Gamma(1-iq)} \right\} , \tag{27b}$$

and q is given by Eq. (17). An immediate result is

$$\sigma_m/\sigma_r = 1, \quad \theta = 0 \; . \tag{28}$$

Eq. (16) for σ_m appears to blow up at $\theta = \pi$ as a result of the $\sec^2(\theta/2)$ term; however, from Eq. (19) $G_o(\pi) = 0$, and since $P_k(-1) = (-)^k$, from Eq. (21) $G_1(\pi) = 0$ as well. Eq. (16) thus becomes for $\theta = \pi$

$$\sigma_m(\pi) = \frac{7.944723 \times 10^{-26} Z^2}{\tau^2(\tau+2)^2} \, |F_o^* + F_1^*|^2 \; , \tag{29}$$

where

$$F_o^* = \tfrac{1}{2} i\Gamma(1-iq)/\Gamma(1+iq) \; , \tag{29a}$$

$$F_1^* = i \sum_{k=1}^{\infty} kD_k \; . \tag{29b}$$

D. The Goudsmit-Saunderson Angular Multiple-Scattering Distribution

The evaluation of the Goudsmit-Saunderson distribution using recursion relations developed by Spencer[16] is discussed by Berger.[1] The present section is an expanded (and to an extent corrected) version of Berger's discussion.

The distribution is given by

$$A[\mu, t(s)] = \sum_{\ell=0}^{\infty} (\ell + \tfrac{1}{2}) \exp\left\{ -\int_0^s G_\ell(s')ds' \right\} P_\ell(\mu) \; , \tag{30}$$

where μ is the cosine of the angle of deflection at path length s relative to the flight direction at s = 0, $A(\mu, t)$ is the probability defined in Section I, t(s) is given by Eq. (2), $P_\ell(\mu)$ is the Legendre polynomial of order ℓ, and

$$G_\ell(s') = 2\pi N \int_0^\pi \sigma[\tau(s'), \theta][1-P_\ell(\cos\theta)] \sin\theta \, d\theta \quad . \tag{31}$$

In Eq. (31), N is the numer of atoms/cm^3, $\sigma(\tau, \theta)$ is the single scattering cross section, and $\tau(s)$ is given by Eq. (5).

Berger derives an expression for σ by applying a screening correction obtained from Moliere's multiple scattering theory[17] to the Rutherford cross section σ_r, and multiplying the result by σ_m/σ_r, where σ_m is the Mott cross section. The factor σ_m/σ_r is in turn expressed as the sum of the small angle analytical formula [Eq. (27) + the screening term] for this ratio and a function $h(\theta)$, which is the true ratio σ_m/σ_r minus the small angle formula; this permits the integration of (31) to be carried out with relatively high precision in the small angle region, where the cross section has both its highest values and greatest degree of variation. Finally, the factor Z^2 in the Rutherford formula is replaced with Z(Z+1) to take into account inelastic scattering. The resulting expression multiplied by $2\pi N$ [expressed in terms of the density ρ (g/cm^3) and atomic weight (W)] is

$$2\pi N\sigma(\tau, \theta) = [0.301314\, \rho Z(Z+1)/W][(\tau+1)/\tau(\tau+2)]^2(1-\cos\theta + 2\eta)^{-2}$$
$$\times \left\{1 + (\pi q/\sqrt{2}) \cos\gamma \, (1-\cos\theta + 2\eta)^{\frac{1}{2}} + h(\tau, \theta)\right\}, \tag{32}$$

where q is given by Eq. (17),

$$\eta = 1.7 \times 10^{-5} Z^{2/3}[\tau(\tau+2)]^{-1}(1.13 + 3.76q^{2}) ,^{*} \tag{33}$$

and

$$h(\tau, \theta) = (\sigma_{m}/\sigma_{r}) - 1 - (\pi q/\sqrt{2}) \cos \gamma \, (1 - \cos \theta + 2\eta)^{\frac{1}{2}} . \tag{34}$$

The ratio σ_{m}/σ_{r} is given by the expressions of Section II. C. Berger notes in his paper a further small corrective term to $\sigma(\tau, \theta)$ suggested by Spencer, but he has since found this term to be suspect,[18] and it is accordingly omitted here.

The function $h(\tau, \theta)$ defined by Eq. (34) has been found to be well represented by a polynomial of the form[16]

$$h(\tau,\theta) = \sum_{j=1}^{J} h_{j}(\tau)(1-\cos \theta + 2\eta)^{j/2} \tag{35}$$

where J is approximately 6. This expression is substituted into Eq. (32), and the resulting expression for $2\pi N\sigma$ is substituted into Eq. (31) to yield

$$G_{\ell}[\tau(s)] = 0.301314\frac{\rho Z(Z+1)}{W}\left[\frac{(\tau+1)}{\tau(\tau+2)}\right]^{2}\left\{p(-2, \ell) + [h_{1}(\tau) + (\pi q/\sqrt{2}) \cos \gamma] \times p(-3/2, \ell) + \sum_{j=2}^{J} h_{j}(\tau)p[\tfrac{1}{2}(j-4), \ell]\right\} \tag{36}$$

where

$$p(m, \ell) = \int_{-1}^{1} (1-\mu' + 2\eta)^{m} \, [1-P_{\ell}(\mu')]d\mu' \ . \tag{37}$$

*Berger's version of this formula,[1] which omits the factor $[\tau(\tau+2)]^{-1}$, is incorrect.

Spencer has developed recursion relations for the terms p(m, ℓ), which are as follows[17]:

$$p(m,0) = 0 \quad , \tag{38}$$

$$p(-2,1) = \ln(1+\eta^{-1})-(1+\eta)^{-1} \quad , \tag{39}$$

$$\ell p(-2,\ell+1) = (2\ell+1)(1+2\eta)p(-2,\ell)-(\ell+1)p(-2,\ell-1)-(2\ell+1)(1+\eta)^{-1}, \quad \ell\geqslant 1 \tag{40}$$

$$p(-3/2,1) = 2(2\bar{\eta})^{3/2}(1+\bar{\eta})^{-1} \quad , \tag{41}$$

$$p(-3/2,\ell+1) = \bar{\eta}p(-3/2,\ell) + p(-3/2,1) \quad , \qquad \ell\geqslant 1 \tag{42}$$

$$p(m+1,\ell) = (1+2\eta)p(m,\ell) + p(m,1) - \frac{\ell+1}{2\ell+1}p(m,\ell+1) - \frac{\ell}{2\ell+1}p(m,\ell-1) \tag{43}$$

where

$$\bar{\eta} = 1-2\eta[-1+(1+\eta^{-1})^{1/2}] \quad . \tag{44}$$

As an example, consider the generation of the p(m, ℓ) terms needed to evaluate Eq. (36) for J = 5. The required values of m for any ℓ are -2, -1, 0, -3/2, -1/2, and +1/2. The following schematic table indicates the order in which the p's are evaluated for values of ℓ from 1 to 4 inclusive: a (1) under (m = -2, ℓ = 1) indicates that p(-2,1) is calculated first, etc.

m →	0	-1	-2	ℓ	-3/2	$-\frac{1}{2}$	$\frac{1}{2}$	← m
	(6)	(3)	(1)	1	(1)	(3)	(6)	
	(9)	(5)	(2)	2	(2)	(5)	(9)	
	(12)	(8)	(4)	3	(4)	(8)	(12)	
	(15)	(11)	(7)	4	(7)	(11)	(15)	
		(14)	(10)	5	(10)	(14)		
			(13)	6	(13)			

According to this table, the first p(m, ℓ)'s to be calculated are p(-2, 1) [from Eq. (39)] and p(-3/2, 1) [Eq. (41)]; next, p(-2, 2) [Eq. (40)] and p(-3/2, 2) [Eq. (42)]; then p(-1, 1) and p(-1/2, 1) [Eq. (43)], and so on. The row corresponding to a given value of ℓ contains the six p's needed to evaluate Eq. (36) for that ℓ.

If $G_1(\tau)$ is evaluated from Eq. (36) as indicated above, and the result is expressed first in terms of s [using Eq. (5) et seq.] and then in terms of t [as defined by Eq. (2)], it is found[16] that the resulting function is well approximated by

$$G_1(t) = c_1[t(t+c_2)]^{-1} \quad . \tag{45}$$

The constants c_1 and c_2 are best determined by solving Eq. (45) using the exact values of $G_1(t)$ at $t = 1$ $(\tau = \tau_0)$ and at a second value of t near $t = 1$. It is further found[16] that to a good approximation

$$G_\ell(t) = G_1(t)G_\ell(1)/G_1(1) \quad . \tag{46}$$

From Eqs. (2), (45), and (46),

$$\int_0^s G_\ell(s')ds' = s_t \int_t^1 G_\ell(t')dt' = \frac{s_t c_1}{c_2} \frac{G_\ell(1)}{G_1(1)} \ln\left[\frac{t+c_2}{t(1+c_2)}\right]^{*} . \tag{47}$$

It is thus seen that $G_\ell(t)$, given implicitly by Eq. (36), need only be calculated for two values of t when $\ell = 1$ [to determine c_1, c_2, and $G_1(1)$], and for one value, $t = 1$, when $\ell > 1$ [since only $G_\ell(1)$ occurs in Eq. (47)].

All relations needed for the evaluation of the distribution, Eq. (30), have now been established. The full calculational procedure is summarized in Section III.

E. Convergence of the Goudsmit-Saunderson Series

A limitation of the Goudsmit-Saunderson distribution is that the series of Eq. (30) requires a prohibitively large number of terms for convergence when the fractional residual path length t exceeds a critical value; in fact, it appears that the convergence of the series may be asymptotic, which is to say the most accurate result is obtained by summing a finite number of terms, beyond which accuracy decreases. The onset of this phenomenon is quite dramatic: in one case studied, the series converged in 75 terms for $t = 0.95$, while convergence was far from achieved after 500 terms for $t = 0.97$.

*This formula is also given incorrectly in ref. 1.

It has been found that if the Yennie-Ravenhall-Wilson transformation, (24) with m = 3, is applied to the Goudsmit-Saunderson series for $\mu < 1$, convergence is usually achieved quite rapidly in instances where direct summation fails. The transformed series is

$$A(\mu,t) = (1-\mu)^{-3} \sum_{\ell=0}^{\infty} a_\ell^{(3)} P_\ell(\mu) \quad , \qquad \mu < 1 \tag{48}$$

where

$$a_\ell = (\ell + \tfrac{1}{2})\exp[-\int_0^s G_\ell(s')ds'] \quad ,$$

G_ℓ is given by (31), and $\{a_\ell^{(3)}\}$ are generated from $\{a_\ell\}$ using (24c).

If A (μ=1) is plotted against A (μ=cos 5°) for a variety of atomic numbers and penetrations and a moderately broad range of initial energies, the resulting points very nearly fall on a single curve. (See Figure 1.) This suggests an approximate method of obtaining $A(\mu=1,t)$ when (30) does not converge for μ=1 but (30) or (48) converges for μ=cos 5°: namely, evaluate A(cos 5°, t) and interpolate on the appropriate curve of Figure 1 to obtain A(1, t). It is then in turn possible to calculate $A(\mu,t)$ if cos 5° < μ < 1 and (48) diverges for μ but converges for cos 5°. It has been observed[3b] that at small angles the multiple scattering distribution behaves like a modified Gaussian,

$$A(\mu,t) \approx \ A(1,t)e^{-K(1-\mu^2)} \quad .$$

If K is chosen so that the given function fits the known values of A(1,t) and A(cos 5°,t) (the former having been obtained from the latter by interpolation from Figure 1), the resulting expression is

$$A(\mu,t) \approx A(1,t)\exp\left[\frac{1-\mu^2}{1-\cos^2 5^\circ}\ \ln\frac{A(1,t)}{A(\cos 5^\circ,t)}\right],\quad \mu > \cos 5^\circ. \tag{49}$$

The approximation just presented is not likely to be required in most calculations, since (30) or (48) converge almost everywhere, failing only when both the energy and the penetration are quite low. For completeness, it should be noted that if the penetration is *exceptionally* small and the energy is low, even approximation (49) fails, for one of two reasons: first, Eq. (48) may not converge for $\mu = \cos 5^\circ$, and second, the [A(μ=1), A(μ=cos 5°)] pair may lie beyond the curve of Figure 1. The implication in either case is that the distribution is very nearly a delta function, and hence angular deflections may be neglected with little attendant error. It is emphasized, however, that the occurrence of such a situation in a transport calculation is most unlikely.

III. OUTLINE OF THE CODE

A. Path Length vs. Energy: TTAB, DSCAL, DMUS, DENCOR, GAUS

The five named routines perform the calculations outlined in Section II.A. DENCOR generates the polarization correction δ of Eq. (7) by applying an interpolation formula to the appropriate curve of ref. 10. DMUS calculates $(d\tau/ds)$ from (9), and DSCAL calculates $|d\tau/ds|^{-1}$, the integrand of (6). TTAB carries out the integration of (6) using GAUS, a numerical quadrature subroutine which must be written or otherwise obtained by the user (see Section IV.B.). The quantities generated by TTAB include the total path length s_t for a given initial energy E_o, a table of path length $s(E_o,E)$ vs. energy E, and the fractional residual path length t corresponding to each tabulated s. Both energy E and dimensionless energy τ are stored as alternative independent variables.

The table of s vs. E is generated by TTAB as follows. At the initial energy E_o the path length s by definition equals 0. An energy E_1 is calculated as E_o-h, where h is an energy decrement chosen by the user, and 12-point Gaussian quadrature is used to calculate $s(E_o,E_2)$ from Eq. (6). A second energy $E_2 = E_1-h$ is generated, integration over the range (E_2,E_1) is carried out, and the result is added to $s(E_o,E_1)$ to obtain $s(E_o,E_2)$. A final energy E=0 is ultimately reached, and $s_t = s(E_o, 0)$ is generated. Once the energy falls below 0.1 MeV, the slope dE/ds begins to change with increasing rapidity (see the tables of ref. 10), and so 20-point quadrature is used in this range. The values of s so calculated are estimated to be no more than 0.1% in error except at the very lowest energies, say less than 0.05 MeV, where the error may rise to 1 to 2%. Naturally, the smaller the energy decrement h the less is the error.

B. Range-Energy-Path Length: RANGEN

Subroutine RANGEN performs one of four calculations, according to the value of a parameter INDEX. Let $r_t(E)$ be the range and $t(E_o, E)$ the fractional residual path length as defined in Section I, expressed in terms of E (MeV) rather than τ, and define r(E) as $r_t(E_o) - r_t(E)$, so that r(E) is the distance the electron has traveled in the initial direction of its flight when its energy is E, its initial energy having been E_o. The energy E_o is assumed known in the ensuing discussion.

If INDEX = 1, the total range $r_t(E_o)$ is calculated from the appropriate Katz-Penfold formula [(11) or (13)]. This calculation must be performed before RANGEN can be called with INDEX greater than 1. If INDEX = 2, a value of r is read, the residual range $r_t(E)$ corresponding to this r is calculated as $r_t(E_o) - r$, the energy $E(r_t)$ is calculated from Eq. (12) or (14), and the fractional residual path length t(E) is obtained by interpolating in the previously generated t–E table. If INDEX = 3, the last of these calculations is omitted, and only $E[r_t(r)]$ is determined. If INDEX = 4, a value of t is read, E(t) is obtained by interpolation in the t–E table, $r_t(E)$ is calculated from Eq. (11) or (13), and r(E) is obtained as $r_t(E_o)-r_t(E)$.

C. Mott and Rutherford Cross Sections: RMOTT, RM1, CGAM

RMOTT calculates the Mott cross section σ_m and the Rutherford cross section σ_r using the formulas of Section II.C. If $\theta \leqq 10°$, the ratio σ_m/σ_r is calculated from (27), σ_r is calculated from (15), and σ_m is obtained as the product of these quantities. The term cos γ of (27) is generated by CGAM in the first of these calculations.

If $\theta > 10°$, the full calculational procedure of (16) et seq. is carried out to obtain σ_m. The generation of the coefficients $\{D_k\}$ [Eq. (22)] as they are needed is carried out in RM1. The series of (25) and (26) are summed until twice in succession the addition of another term changes the value of F (or G) by less than five parts in 10^6.

If RMOTT is called for $\theta = 180°$, the value 179° is substituted and a message to this effect is printed out. The difference between $\sigma_m(179°)$ and $\sigma_m(180°)$ is insufficient to warrant the inclusion of Eq. (29) in the program.

D. Goudsmit-Saunderson Distribution: GOUD, HFIT, CGAM, G1G, GLG

To generate $A(\mu,t)$ for a given initial energy, penetration, and angular deflection, GOUD must be called three times, with a parameter INDEX successively taking on values of 1, 2, and 3. When GOUD is called with INDEX = 1 for a given initial energy τ_0, it in turn calls HFIT, which calculates $h(\theta)$ from Eq. (34) at IK equally spaced angles from 0 to π, and fits $h(\theta)$ exactly at these points with a polynomial of the form of Eq. (35). IK equals 7 in the program as currently written; it may be changed to any integer from 3 to 10 inclusive by redefining it in subroutine DEFINE. CGAM calculates cos γ from Eq. (27b) in the course of the evaluation of $h(\theta)$. GOUD then calls G1G, which uses the calculated coefficients $\{h_j\}$ to evaluate $G_1(t)$ from Eqs. (36)-(44) at two values of t: $t = 1$ ($\tau = \tau_0$) and the value once removed from $t = 1$ in the table generated as indicated in Section III.A. The resulting values of G_1 are used to determine c_1 and c_2 from Eq. (45), and control is then returned to the calling program.

The first value of the deflection angle cosine μ for which the distribution is calculated for a given fractional residual path length t must be 1, a calculation performed when GOUD is called with INDEX = 2. There are several reasons for this procedure. It is useful to distinguish the first calculation for a given t from subsequent calculations, since the values of certain quantities which do not depend on μ must be calculated only once, and thereafter may be read as stored variables. The value $\mu = 1$ is a convenient choice for isolation, since the Legendre polynomials $\{P_\ell(1)\}$ of (30) are all equal to 1, and hence the recursive calculations used to generate the polynomials for other values of μ are in this case superfluous. In addition, at small penetrations ($t \approx 1$) convergence problems occur, and the subsequent method of calculation depends on the behavior of the Goudsmit-Saunderson series for $\mu = 1$ (see Section II.E).

When GOUD is called with INDEX = 2 (for $\mu = 1$) or INDEX = 3 (for $\mu \neq 1$), terms of the series (30) are evaluated from (47) and summed. The terms $\{G_\ell(1)\}$ of (47) are calculated from (36)-(44) by subroutine GLG. The summation is terminated when twice in succession either the addition of a term changes the value of $A(\mu,t)$ by less than one part in 10^5 or the absolute value of the term added is less than 10^{-7}. The factors

$$(\ell + \tfrac{1}{2})\exp[-\int_0^s G_\ell(s')ds']$$

are stored, so that they need only be calculated once for a given t, regardless of the number of values of μ for which (30) is evaluated.

If one of the given convergence criteria is not satisfied after 150 terms have been summed and $\mu \leqq \cos 5°$, the transformed series for $A(\mu,t)$, Eq. (48), is summed, while if (30) fails to converge for $\mu > \cos 5°$, the approximate method described in the paragraph following (48a) is used. The returned value of one of the parameters of GOUD named LT indicates the procedure followed. If $3 \leqq \mathrm{LT} \leqq 150$, the series (30) converged, with LT being the number of terms required for convergence. If LT > 150, then (30) did not converge but (48) did, in LT−150 terms. If LT = 1, (30) failed to converge and the approximation following (48a) was used. Finally, if LT = 0, even the approximation failed, and the returned value of $A(\mu,t)$ is meaningless.

E. Miscellaneous Routines: COMPLX, CONJGX, DEFINE, GAMCAR, GAUS, LEQ

COMPLX generates a complex number from two real arguments

CONJGX calculates the conjugate of a complex number

DEFINE performs preliminary bookkeeping chores

GAMCAR calculates the gamma function of a complex argument

GAUS performs numerical quadrature

LEQ solves simultaneous linear algebraic equations

F. Restrictions and Limitations of the Code

The code may be used to evaluate the Goudsmit-Saunderson distribution for a single-element stopping medium having one of the following atomic numbers: Z = 3, 4, 6, 12, 13, 26, 29, 47, 50, 79, 82, 92. To perform the calculation for an atomic number not included in this list, a new function DENCOR must be written to generate the polarization correction $\delta(E)$ of Eq. (7). The details of the calculation of δ are given in refs. 7 and 9. To calculate the distribution for a stopping medium which is a compound or mixture of elements, TTAB must be rewritten so that a weighted average of single-element stopping powers is calculated and integrated to obtain the path length vs. energy table. This calculation is also outlined in ref. 7.

Three factors place an upper bound on the initial energy E_0. First, Function DENCOR generates $\delta(E)$ by interpolation on the curves of ref. 10, which apply to $E \leqq 10$ MeV. For initial energies $E_0 > 10$ MeV, DENCOR must be rewritten as indicated in the previous paragraph. Second, the range-energy relations of Katz and Penfold [Eqs. (11)-(14)] apply only to $E \leqq 20$ MeV, so that for $E_0 > 20$ MeV another range-energy relation must be found and written into the code. The third limitation results from the use of the approximate factor $1 + (Z\tau/1400)$ of Eq. (10) to estimate bremsstrahlung losses. An approximate upper limit on E_0 may be obtained by limiting the bremsstrahlung loss as calculated by the given expression to at most 10% of the collisional loss, which implies $ZE_0 < 72$, where E_0 is in MeV. For initial energies too high to satisfy the criterion, a more precise expression for the radiative energy loss should be used in subroutine DMUS (see, e.g., ref. 2).

It has been noted that the Goudsmit-Saunderson series fails to converge at low initial energies and small penetrations; in general, the lower the energy, the broader the range of penetrations in which this failure occurs. To insure that the series or the transformed series [Eq. (48)] converges down to penetrations small enough for angular deflections to be negligible, a lower bound should be imposed on the initial energy. $E_0 = 0.05$ MeV has been found adequate for this purpose.

Most of the program limitations apart from those inherent in the theoretical model are imposed by the dimensioning of certain arrays. The maximum number of intervals permitted in the generated t vs. E table is 150; in practice, no more than 100 are ever likely to be required for precision. If for some reason it is desired to increase this number, the dimensions of arrays E (or EG), TAU (or TAUG), and T (or TG) must be increased to the new maximum plus two in every routine in which Common BLOCKG occurs, the dimension of S must be similarly increased in Subroutine TTAB, and the number 150.0 in the second and third executable statements of TTAB should be changed to the new maximum.

Subroutine RMOTT, which generates the Mott cross section, provides for the evaluation of up to 90 terms of the series of Eqs. (25) and (26); if convergence to within five parts in 10^6 is not achieved in this many terms, a message to this effect is printed and the program is terminated. If it should ever prove necessary to increase the maximum number of terms, and this is most unlikely, the dimensions of complex arrays UF and UG in RMOTT must be changed to the desired maximum, and the terminal value of the DO 200 statement must be similarly altered.

The maximum number of terms provided by Subroutine GOUD for the evaluation of (30) and (48) for the Goudsmit-Saunderson distribution is 150. It is not recommended that this limit be increased, but if it is desired to do so, the dimension of GX (or GXG) in Common BLOCKG and the dimensions of GL and YX in Subroutine GOUD should be increased to 10 more than the new maximum, and the terminal indices of the DO 115, DO 150, and DO 490 statements in GOUD should be increased to the new maximum.

The parameter restrictions of the code are summarized below.

Atomic number: $Z = 3, 4, 6, 12, 13, 26, 29, 47, 50, 79, 82, 92$;

Initial energy: $0.05 \leqq E_o \leqq 10$ MeV;

Combination of Z and E_o: $ZE_o < 72$.

IV. GUIDE TO THE USE OF THE CODE

A. Calling Program for Energy Loss and Angular Distribution Calculations

The user must supply FORTRAN statements and define variables in the order given. Ø refers to the letter, 0 to the number.

(1) Statement.

CØMMØN/BLØCKG/RHØG, ZG, AG, RHØZAG, ALPHG, ALPHSG, AIMG, NANGG, ANGG(37), ANGRG(37), CØSG(37), NSTEPG, EG(152), TAUG(152), TG(152), RTG, STG, NTG, N1G, N2G, F1G, F2G, F3G, F4G, F5G, F6G, F7G, IKG, C1G, C1XG, BMG(10,10), PG(10), GXG(160), JXG, JHG, JPG, JH1G, JH2G, IØPTG, ASCØRG, BSCØRG, DSCØRG, NCRVG

The user should take care not to define in his program any of the variable names in this common block except as subsequently indicated. The easiest way to do this is to avoid ending variable names with the letter G.

(2) Define or read in as data: NCRVG

The value of this variable, which specifies the stopping element, must be an integer from 1 to 11 inclusive. The possible values of NCRVG and the corresponding elements and atomic number are as follows: 1(Li,3), 2(Be,4), 3(Graphite, 6), 4(Mg,12), 5(Al,13), 6(Fe,26), 7(Cu,29), 8(Ag,47), 9(Sn,50), 10(Pb,82), 11(U,92).

(3) Statement: CALL DEFINE

DEFINE defines a number of constants for subsequent energy loss and angular distribution calculations including RHOG, ZG, and AG, the density (g/cm^3), atomic number and atomic weight respectively of the stopping medium. In addition, the routine generates an array ANGG, which contains angles 0°, 5°, 10°,...,175°,179°, another array ANGRG, which contains the same angles in radians, and a third array CØSG, which contains the cosines of the angles. The latter two arrays are needed in subsequent calculations, but any or all of them are also available to the user for other purposes, e.g., printed tabulation of data.

(4) Define:

EI: initial electron energy, MeV. See Section III. F. for bounds on this parameter.

EMIN: energy below which the electron is essentially at the end of its range. A suggested value is 0.01.

NSTEPG: number of intervals between EI and EMIN. A suggested value is 50, the maximum is 150.

(5) Statement: CALL TTAB (EI,EMIN)

TTAB generates a table of TG vs. TAUG (and EG), with EG varying from EI to EMIN in NSTEPG equal decrements, as well as STG, where

TG = t, defined by Eq. (2), dimensionless

EG = electron energy, MeV

TAUG = τ, defined by Eq. (1), dimensionless

STG = s_t, the total path length corresponding to initial energy EI, cm.

The path length s for each tabulated value EG is not stored in common, but if desired may be calculated directly from STG (s_t) and TG (t) using Eq. (2), or printed out directly in TTAB. If only path length vs. energy data are required, the program terminates at this point.

(6) Statement: CALL GOUD (1,DUMMY,DUMMY,DUMMY,DUMMY)

This call of GOUD sets up the Goudsmit-Saunderson distribution calculation for initial energy EI, as discussed in Section III.D. The last four arguments of the calling statement are dummy variables, as the observant reader may have guessed.

(7) Statement: CALL RANGEN (1,RTG,EI,1.0)

RANGEN here calculates RTG, the extrapolated range in cm corresponding to initial energy EI, using (11) or (13).

If the distribution is to be calculated for a given distance traveled in the initial direction of the electron's flight, a conversion must be made to obtain t, the fractional residual path length at the end of this distance. If t is known to begin with, the next two steps are omitted.

(8) Define: R, the straight-line penetration, or the distance in cm traveled in the initial direction of the electron's flight.

(9) Statement: CALL RANGEN (2,R,EN,TN)

RANGEN first calculates the energy EN possessed by the electron after it has traversed the straight-line distance R, and then interpolates in the t-E table generated in Step (5) to obtain the fractional residual path length TN = t(EN). The details of the calculation are described in Section III. B.

If Steps (8) and (9) are executed, the next two steps are omitted.

(10) Define: TN, the fractional residual path length at the point for which the distribution is to be calculated.

(11) Statement (optional): CALL RANGEN (4,R,EF,TN)

For the given fractional residual path length TN, RANGEN calculates EF, the energy at the end of the path, and then R, the penetration in the initial flight direction. The details of the calculation are given in Section III. B.

(12) Statement: CALL GOUD (2,TN,1.0,AGS,LT)

The Goudsmit-Saunderson distribution calculation for the fractional residual path length TN is set up, and the value of the distribution at $\mu = 1$ ($\omega = 0$) is determined. Quantities calculated include:

AGS = A(μ = 1, t = TN), Eq. (30)

LT = number of terms of the series (30) needed for convergence to within one part in 10^5. If LT = 1, the value of AGS is an approximation, good to within approximately 10%. If LT = 0, the value of AGS cannot be determined, but to a good approximation the distribution may be taken to be a delta function.

In addition the computer stores in common for each term of the series from L =1 to L = LT

GXG(L) = expression (51) .

These terms are functions of t but not of μ. In calculations for a given t, each of them is determined once and stored, and thereafter the stored values are used directly. The individual terms GXG(L), which as the Legendre expansion coefficients of the multiple scattering law are sometimes separately needed in the course of a calculation, are thus available to the user.

(13) Statement: IF(LT-1) n_1, n_2, n_3

Statement n_1 should begin a segment of the program based on the fact of the nonconvergence of the Goudsmit-Saunderson series for $\mu = 1$ and the failure of the approximation formulas. It may be assumed that the distribution is nearly a delta function peaking at $\mu = 1$. Statement n_2 should begin a segment which either uses or rejects the returned value of AGS, which has been generated by interpolation on the curves of Figure 1. If it is desired to use the value, n_2 should equal n_3, the statement number of the next step, to which control proceeds if the Goudsmit-Saunderson series converges.

(14) Define: AMU = μ, a direction cosine for which $A(\mu,t)$ is to be calculated.

(15) Statement: CALL GOUD (3,TN,AMU,AGS,LT)

This statement causes to be calculated AGS = A (μ = AMU, t = TN). The meaning of LT is as given following Step (12), with the addition that if LT > 150 the transformed series (48) was used to calculate AGS, and converged in LT-150 terms.

(16) Statement: IF (LT-1) n_1,n_2,n_3

n_1, n_2, and n_3 are as defined following Step (13). Normally n_2 should equal n_3, the continuation of the program.

Once Steps (1)-(3) have been executed, each calculation that involves a new initial energy must begin with Step (4); when a new t (or r) is introduced at the same initial energy, the calculation recommences with Step (10) [or Step (8)], and for different values of μ at a given t only Steps (14)-(16) are required.

B. User-Written Subroutines for Energy Loss and Angular Distribution Calculations

The user must supply the following subroutines in addition to those given in Section VI.

1. SUBROUTINE GAUS (FUNC,A,B,NORD,INT)

GAUS carries out the integration

$$\int_A^B \text{FUNC}(x)dx$$

by dividing the interval A≤x≤B into INT equal subintervals, and performing

NORD-point quadrature on each subinterval. A SHARE subroutine of this type which uses Gaussian quadrature is available to institutions participating in this system; otherwise, the routine may be written using standard tables.[19] The code calls for NORD = 12 and 20 and INT = 1, so that a user-written routine need only provide for these values. If the user's version of GAUS does not provide for subdivision of the initial range (INT = 1 only), the defining statement should still contain five arguments, the fifth being a dummy integer. Any routine that calls GAUS should declare the name FUNC in an EXTERNAL statement.

2. SUBROUTINE LEQ (A,B,N,M,IA,IB,DET)

LEQ, which is part of many computer libraries, solves the matrix equation AX = B and evaluates the determinant of the matrix A. The subroutine parameters are defined as follows:

A is a two-dimensional array with dimensions (IA,IX), where both IA and IX are greater than or equal to N;

B is a one- or two-dimensional array with dimensions (IB,IY), where IB⩾N and IY⩾M. If B is one-dimensional, IY may be omitted;

N is the number of equations and unknowns;

M is the number of vector solutions desired (=1 in the code);

IA,IB are the first dimensions of arrays A and B respectively;

DET contains the determinant of A.

The elements of the solution matrix X are stored in B(I,K), where I = 1,2,...,N and K = 1,2,...,M. The original A and B matrices are destroyed by the routine.

C. Calling Program for the Mott Cross Section

The values of the Mott to Rutherford cross section ratio σ_m/σ_r needed for the Goudsmit-Saunderson distribution calculation are generated within the code. If it is desired to obtain σ_m in a separate calculation, the following procedure must be followed:

(1) Define:

Z = atomic number of the scattering medium

E = initial electron energy, MeV

(2) Define:

THETA = center-of-mass scattering angle, radians

AMU = $\cos(\theta)$

(3) Statement: CALL RMOTT (Z,E,AMU,THETA,R,SIGM,SIGR,1)

The calculated quantities are

SIGM = $\sigma_m(E,\theta)$, cm^2

SIGR = $\sigma_r(E,\theta)$, cm^2

R = σ_m/σ_r

The details of the calculation are described in Section III.C.

If cross sections are to be calculated for additional angles at the same Z and E, the next two steps are successively executed for each calculation. If Z or E changes, the calculation must begin with Step (1).

(4) Define: THETA,AMU [as in Step (2)]

(5) Statement: CALL RMOTT (Z,E,AMU,THETA,R,SIGM,SIGR,2)

If RMOTT is called for $\theta = 0$, it returns the correct value of R (=1), and does nothing about SIGM and SIGR, both of which are infinite. It should be clear that the numbers returned for these variables are in this case meaningless, and should not be used.

D. Storage Requirements

The code for the generation of the Goudsmit-Saunderson distribution, including the calling program GOUDS given in Section VI, occupies about 25,000 (octal) or 11,000 (decimal) locations.

V. ILLUSTRATIVE RESULTS

Typical results generated by the code are displayed in Tables 1-8, along with the times in seconds required to generate them. The latter figures represent central processor time on a CDC 6600 computer; a rough estimate of the equivalent calculating time on an IBM 7094 may be obtained by multiplying the given figures by 5.

The results of path length vs. energy calculations for two atomic numbers and a single initial energy are shown in Table 1. The time required for calculations of this sort is approximately $0.0012n_i$ seconds, where n_i is the number of intervals in the table.

Tables 2-4 list values of the Mott to Rutherford cross section ratio for several energies, angles, and atomic numbers. The time required for the calculation of a single value is approximately 0.05 seconds.

Tables 5-8 give representative values of the Goudsmit-Saunderson distribution $A(\omega,t)$ for two atomic numbers and two initial energies. The final energies E_f corresponding to the tabulated fractional residual path lengths t are shown, as are the values of LT for each calculation. To repeat from the discussion of Section III.D., if $3 \leqq LT \leqq 150$, the Goudsmit-Saunderson series (30) converged to within one part in 10^5 in LT terms; if $LT > 150$, the transformed series (48) converged in LT-150 terms; if $LT = 1$, the approximation given in the paragraph following (48a) was used, and if $LT = 0$, the value of A could not be determined.

A number of characteristic features of the distribution may be observed in these tables. At small values of t, corresponding to large penetrations, the distribution approaches isotropy: in the limit as t approaches 0, $A(\omega,t) \equiv 0.5$. As t increases, the distribution becomes increasingly biased in the initial direction of the electron's flight ($\omega = 0$), and the number of terms required to evaluate the series expression (30) for $A(\omega,t)$ also increases. As t approaches 1, (30) fails to converge and the transformed series (48) is summed instead: this occurs first at large angles and low initial energies, and eventually over the entire range of angles $\omega>0$ and at all initial energies for t sufficiently close to 1.

The computing times given in Tables 5-8 do not include the times required to generate the required t-E tables (≈0.1 seconds per table), and to call GOUD with INDEX = 1 (≈0.2 seconds).

Figure 1 displays a collection of pairs $[A(\omega=0^\circ), A(\omega=5^\circ)]$. The points shown represent data for Z = 13, 50, and 82, and a variety of fractional residual path lengths t. The fact that the curves shown fit the data as well as they do despite the range of energies and atomic numbers forms the basis for the approximation given after (48a).

VI. FORTRAN IV CODE

Listings follow of Programs GCAL and SIGMOT, which are sample calling programs for Goudsmit-Saunderson distribution and Mott cross section calculations respectively, and subroutines GOUD, DEFINE, TTAB, DENCOR, DSCAL, DMUS, HFIT, RANGEN, CGAM, G1G, GLG, RMOTT, RM1, GAMCAR, CMPLX, and CONJGX. The functions of each of these routines are described in Section III.

GCAL is set up so that a single set of parameters is read, the distribution is calculated, and the program terminated. If it is desired to run the program for several sets of data, the CALL EXIT statement (GCAL 051) should be changed to GO TO 25 (for a new stopping element), GO TO 30 (for the same stopping element and a new initial energy), or GO TO 40 (for the same stopping element and initial energy and a new penetration). SIGMOT is set up to read an atomic number Z, calculate the Mott to Rutherford cross section ratio for several energies and angles, read another Z, etc.

SAMPLE CALLING PROGRAM FOR GOUDSMIT-SAUNDERSON CALCULATION

```
      PROGRAM GCAL(INPUT,OUTPUT)                                        GCAL 001
      COMMON/BLOCKG/RHOG,ZG,AG,RHOZAG,ALPHG,ALPHSG,AIMG,NANGG,ANGG(37),AGCAL 002
     1NGRG(37),COSG(37),NSTEPG,EG(152),TAUG(152),TG(152),RTG,STG,NTG,N1GGCAL 003
     2,N2G,F1G,F2G,F3G,F4G,F5G,F6G,F7G,IKG,C1G,C1XG,BMG(10,10),PG(10),GXGCAL 004
     3G(160),JXG,JHG,JPG,JH1G,JH2G,IOPTG,ASCORG,BSCORG,DSCORG,NCRVG     GCAL 005
    4 FORMAT(1H 10X3HE =F6.3,6X3HS =E10.3,6X3HT =E10.3)                 GCAL 006
   25 READ 1,NCRVG                                                      GCAL 007
    1 FORMAT(I10)                                                       GCAL 008
      CALL DEFINE                                                       GCAL 009
    2 FORMAT(2F10.0,I10)                                                GCAL 010
   30 READ 2,EI,EMIN,NSTEPG                                             GCAL 011
      CALL TTAB(EI,EMIN)                                                GCAL 012
      CALL RANGEN(1,RTG,EI,1.0)                                         GCAL 013
      PRINT 3,ZG,EI,STG,RTG                                             GCAL 014
    3 FORMAT(1H120X*ENERGY VS. FRACTIONAL RESIDUAL PATH LENGTH*//5X3HZ =GCAL 015
     1F5.1,6X4HEI =F5.2,6X*TOTAL PATH LENGTH ST =*E10.3,6X*RANGE RT =*E1GCAL 016
     20.3/)                                                             GCAL 017
      DO 35J=1,NTG                                                      GCAL 018
      S=STG*(1.0-TG(J))                                                 GCAL 019
   35 PRINT 4,EG(J),S,TG(J)                                             GCAL 020
      PRINT 5,ZG,EI                                                     GCAL 021
    5 FORMAT(1H110X*GOUDSMIT-SAUNDERSON DISTRIBUTION  Z =*F5.1,5X4HEI =FGCAL 022
     15.2/)                                                             GCAL 023
      CALL GOUD(1,DUM,DUM,DUM,DUM)                                      GCAL 024
   40 READ 8,TN                                                         GCAL 025
    8 FORMAT(F10.0)                                                     GCAL 026
      PRINT 17                                                          GCAL 027
   17 FORMAT(1H08X1HT9X2HEF8X5HOMEGA7X3HAGS7X5HTERMS)                   GCAL 028
      CALL RANGEN(4,R,EF,TN)                                            GCAL 029
      CALL GOUD(2,TN,1.0,AGS,LT)                                        GCAL 030
      PRINT 6,TN,EF,ANGG(1),AGS,LT                                      GCAL 031
    6 FORMAT(1H04XF7.4,3XF7.4,4XF7.1,3XE11.4,4XI3)                      GCAL 032
      DO 90J=2,3                                                        GCAL 033
      CALL GOUD(3,TN,COSG(J),AGS,LT)                                    GCAL 034
      IF(LT-150) 62,62,63                                               GCAL 035
   62 PRINT 7,ANGG(J),AGS,LT                                            GCAL 036
    7 FORMAT(1H 25XF7.1,3XE11.4,4XI3)                                   GCAL 037
      GO TO 90                                                          GCAL 038
   63 LT=LT-150                                                         GCAL 039
      PRINT 20,ANGG(J),AGS,LT                                           GCAL 040
   20 FORMAT(1H 25XF7.1,3XE11.4,4XI3,*(YRW)*)                           GCAL 041
   90 CONTINUE                                                          GCAL 042
      DO 100J=4,NANGG,3                                                 GCAL 043
      CALL GOUD(3,TN,COSG(J),AGS,LT)                                    GCAL 044
   93 IF(LT-150) 94,94,95                                               GCAL 045
   94 PRINT 7,ANGG(J),AGS,LT                                            GCAL 046
      GO TO 100                                                         GCAL 047
   95 LT=LT-150                                                         GCAL 048
      PRINT 20,ANGG(J),AGS,LT                                           GCAL 049
  100 CONTINUE                                                          GCAL 050
      CALL EXIT                                                         GCAL 051
      END                                                               GCAL 052
```

SAMPLE CALLING PROGRAM FOR MOTT CROSS SECTION CALCULATION

```
      PROGRAM SIGMOT(INPUT,OUTPUT)                                      SIGM 001
      DIMENSION ANG(20),ANGR(20),COSA(20),E(10),BETAS(10),Q(10),SLAMB(10SIGM 002
     1),R(10,20)                                                        SIGM 003
      DATA NE,(E(I),I=1,4)/4,10.0,1.0,0.1,0.01/                         SIGM 004
      DATA NANG,(ANG(J),J=2,13)/13,5.0,10.0,15.0,30.0,60.0,90.0,120.0,15SIGM 005
     10.0,165.0,170.0,175.0,179.0/                                      SIGM 006
    2 FORMAT(F10.0)                                                     SIGM 007
    3 FORMAT(1H120X42HRATIO OF MOTT TO RUTHERFORD CROSS SECTIONS//4H Z =SIGM 008
     1F5.1//)                                                           SIGM 009
    4 FORMAT(1H 5X1HEF7.2,9(4XF5.2))                                    SIGM 010
    5 FORMAT(1H 5HTHETA/)                                               SIGM 011
    8 FORMAT(1H010XF6.3,* SECONDS CP TIME FOR PRECEDING CALCULATION.*/) SIGM 012
    9 FORMAT(1H F5,1,10(2XF7.4))                                        SIGM 013
      FACT=3.1415927/180.0                                              SIGM 014
      ANG(1)=0.0                                                        SIGM 015
      ANGR(1)=0.0                                                       SIGM 016
      COSA(1)=1.0                                                       SIGM 017
      DO 21J=2,NANG                                                     SIGM 018
      ANGR(J)=ANG(J)*FACT                                               SIGM 019
   21 COSA(J)=COS(ANGR(J))                                              SIGM 020
   30 READ 2,Z                                                          SIGM 021
      IF(Z.EQ.0.0) CALL EXIT                                            SIGM 022
      PRINT 3,Z                                                         SIGM 023
      PRINT 4,(E(I),I=1,NE)                                             SIGM 024
      PRINT 5                                                           SIGM 025
      DO 60I=1,NE                                                       SIGM 026
      J=1                                                               SIGM 027
      CALL RMOTT(Z,E(I),COSA(J),ANGR(J),R(I,J),SIGM,SIGR,1)             SIGM 028
      DO 60J=2,NANG                                                     SIGM 029
   60 CALL RMOTT(Z,E(I),COSA(J),ANGR(J),R(I,J),SIGM,SIGR,2)             SIGM 030
      DO 65J=1,NANG                                                     SIGM 031
      GO TO 30                                                          SIGM 033
   65 PRINT 9,ANG(J),(R(I,J),I=1,NE)                                    SIGM 032
      END                                                               SIGM 034
      SUBROUTINE GOUD(INDEX,TX,COSX,AGS,LT)
C     INDEX=1 FOR NEW EI, =2 FOR NEW TX,OLD EI, =3 FOR AGS(COSW,TX,EI)
      COMMON/BLOCKG/RHO,Z,A,RHOZA,ALPHA,ALPHAS,AIMEAN,NANG,ANG(37),ANGR(
     137),COSA(37),NSTEP,E(152),TAU(152),T(152),RT,ST,NT,N1,N2,F1,F2,F3,
     2F4,F5,F6,F7,IK,C1,C1X,BM(10,10),P(10),GX(160),JX,JH,JP,JH1,JH2,IOP
     3T,ASCOR,BSCOR,DSCOR,NCRV
      DIMENSION G1(2),B(10),BI(10),PAI(5,7),PBI(5,7),PA(5,7),PB(5,7),GL(
     1160),AF(7,3),YX(150),ATOL(15),AT5L(6,2)
      DATA NT5,MR,MRP/6,3,4/
      DATA(ATOL(J),J=1,6)/-.693147,2.30259,2.99573,3.91202,4.31749,4.605
     117/
C     ATOL CONTAINS LOGS OF .5,10,20,50,75,100
      DATA (AT5L(J),J=1,6)/-.693147,2.27213,2.91777,3.71357,4.00733,4.15
     1888/
C     AT5L(COL 1) CONTAINS LOGS OF .5,9.7,18.5,41,55,64
      DATA (AT5L(J),J=7,12)/-.693147,2.25129,2.86220,3.59731,3.87120,4.0
     12535/
C     AT5L(COL 2) CONTAINS LOGS OF .5,9.5,17.5,36.5,48,56
      GO TO (20,85,120),INDEX
   20 LMAXT=0
      NC5=1
      IF(E(1).LT.0.25) NC5=2
      CALL HFIT(1,BETAI,BETASI,ETAI,QI,FP)
      F3X=1.0+1.0/ETAI
      F4X=1.0/(1.0+ETAI)
      F5X=1.0+ETAI+ETAI
      ETABI=1.0-(ETAI+ETAI)*(SQRT(F3X)-1.0)
      BI(1)=C1X/(BETASI*TAU(1)*(TAU(1)+2.0))
      BI(2)=BI(1)*(P(1)+FP)
      DO 40J=2,IK
   40 BI(J+1)=BI(1)*P(J)
      G1(1)=G1G(ETAI,ETABI,PAI,PBI,BI,F3X,F4X,F5X)
      I=N2
      CALL HFIT(I,BETA,BETAS,ETA,Q,FQ)
      F3Y=1.0+1.0/ETA
      F4Y=1.0/(1.0+ETA)
      F5Y=1.0+ETA+ETA
      ETAB=1.0-(ETA+ETA)*(SQRT(F3Y)-1.0)
      B(1)=C1X/(BETAS*TAU(I)*(TAU(I)+2.0))
      B(2)=B(1)*(P(1)+FQ)
      DO 55J=2,IK
   55 B(J+1)=B(1)*P(J)
      G1(2)=G1G(ETA,ETAB,PA,PB,B,F3Y,F4Y,F5Y)
      AJH=JH
      PAI(1,JH1)=((AJH+AJH+1.0)*(F5X*PAI(1,JH)-F4X)-(AJH+1.0)*PAI(1,JX))
     1/AJH
      PBI(1,JH1)=ETABI*PBI(1,JH)+PBI(1,1)
      DO 75J=2,JH
      LF=JH1-J+1
      ALF=LF
      JM1=J-1
      PAI(J,LF)=F5X*PAI(JM1,LF)+PAI(JM1,1)-((ALF+1.0)*PAI(JM1,LF+1)+ALF*
     1PAI(JM1,LF-1))/(ALF+ALF+1.0)
      PBI(J,LF)=F5X*PBI(JM1,LF)+PBI(JM1,1)-((ALF+1.0)*PBI(JM1,LF+1)+ALF*
     1PBI(JM1,LF-1))/(ALF+ALF+1.0)
   75 CONTINUE
      GL(2)=0.0
      DO 77J=1,JX
      JJ=J+J-1                                                          GOUD 058
   77 GL(2)=GL(2)+BI(JJ)*PAI(J,2)+BI(JJ+1)*PBI(J,2)                     GOUD 059
      GO TO (79,81),IOPT                                                GOUD 060
   79 GL(2)=GL(2)+BI(JP)*PAI(JH,2)                                      GOUD 061
      GO TO 83                                                          GOUD 062
   81 GL(2)=GL(2)+BI(JP)*PAI(JH,2)+BI(JP+1)*PBI(JH,2)                   GOUD 063
   83 FACT=1.0/(G1(1)-G1(2)*T(N2))                                      GOUD 064
      P(1)=G1(1)*G1(2)*T(N2)*(T(N2)-1.0)*FACT                           GOUD 065
      P(2)=(G1(2)*T(N2)**2-G1(1))*FACT                                  GOUD 066
      RETURN                                                            GOUD 067
   85 FACT=(TX+P(2))/(TX*(1.0+P(2)))                                    GOUD 068
      COSMAX=-1.1                                                       GOUD 069
      LMAX=0                                                            GOUD 070
      NYX=0                                                             GOUD 071
      AG5=0.0                                                           GOUD 072
      NCONV=0                                                           GOUD 073
      NCONX=0                                                           GOUD 074
      IF(FACT) 90,90,95                                                 GOUD 075
   90 IND1=1                                                            GOUD 076
      GX(1)=0.0                                                         GOUD 077
      GX(2)=0.0                                                         GOUD 078
      AGS=0.5                                                           GOUD 079
      LT=3                                                              GOUD 080
      RETURN                                                            GOUD 081
   95 IND1=2                                                            GOUD 082
      FACT=-ST*P(1)*ALOG(FACT)/(G1(1)*P(2))                             GOUD 083
      GX(1)=1.5*EXP(FACT*G1(1))                                         GOUD 084
      GX(2)=2.5*EXP(FACT*GL(2))                                         GOUD 085
   97 AGP=0.5+GX(1)+GX(2)                                               GOUD 086
      NTR=1                                                             GOUD 087
      DO 115L=3,150                                                     GOUD 088
      AL=L                                                              GOUD 089
      IF(L-LMAXT) 100,100,98                                            GOUD 090
   98 LMAXT=L                                                           GOUD 091
      GL(L)=GLG(L,ETAI,ETABI,PAI,PBI,BI,F4X,F5X)                        GOUD 092
  100 IF(L-LMAX) 105,105,104                                            GOUD 093
  104 GX(L)=(AL+0.5)*EXP(FACT*GL(L))                                    GOUD 094
      IF(GX(L).GT.1.0E6) GO TO 1150                                     GOUD 095
      LMAX=L                                                            GOUD 096
  105 AGS=AGP+GX(L)                                                     GOUD 097
      IF(GX(L)/AGS.LE.1.0E-5) 109,110                                   GOUD 098
  109 IF(NTR.EQ.2) GO TO 117                                            GOUD 099
      NTR=2                                                             GOUD 100
      AGL=AGS                                                           GOUD 101
      AGL1=AGP                                                          GOUD 102
      GO TO 115                                                         GOUD 103
  110 NTR=1                                                             GOUD 104
  115 AGP=AGS                                                           GOUD 105
 1150 NCON=2                                                            GOUD 106
      NCONX=1                                                           GOUD 107
      COSW=COSA(3)                                                      GOUD 108
      GO TO 400                                                         GOUD 109
 1151 IF(AGS.LT.0.5) GO TO 116                                          GOUD 110
      AG10=AGS                                                          GOUD 111
      COSW=COSA(2)                                                      GOUD 112
      NCON=3                                                            GOUD 113
      GO TO 400                                                         GOUD 114
 1152 IF(AG10.LT.10.0.AND.AG10/AGS.LT.0.5) GO TO 116                    GOUD 115
      AG5L=ALOG(AGS)                                                    GOUD 116
      DO 1153J=2,NT5                                                    GOUD 117
      IF(AG5L.LE.AT5L(J,NC5)) 1155,1153                                 GOUD 118
 1153 CONTINUE                                                          GOUD 119
      GO TO 116                                                         GOUD 120
 1155 AG5=AGS                                                           GOUD 121
      JM1=J-1                                                           GOUD 122
      AGOL=ATOL(JM1)+(AG5L-AT5L(JM1,NC5))*(ATOL(J)-ATOL(JM1))/(AT5L(J,NCGOUD 123
     15)-AT5L(JM1,NC5))                                                 GOUD 124
      AG0=EXP(AGOL)                                                     GOUD 125
      AGS=AG0                                                           GOUD 126
      ECON=(AGOL-AG5L)/(1.0-COSA(2)**2)                                 GOUD 127
      LT=1                                                              GOUD 128
      RETURN                                                            GOUD 129
  116 NCONV=1                                                           GOUD 130
 1161 LT=0                                                              GOUD 131
      RETURN                                                            GOUD 132
  117 LT=L                                                              GOUD 133
      ADNM=AGS+AGL1-AGL-AGL                                             GOUD 134
      IF(ABS(ADNM).LE.1.0E-12) GO TO 118                                GOUD 135
      AGS=(AGS*AGL1-AGL**2)/(AGS+AGL1-AGL-AGL)                          GOUD 136
  118 AG0=AGS                                                           GOUD 137
      RETURN                                                            GOUD 138
  120 NCON=1                                                            GOUD 139
      COSW=COSX                                                         GOUD 140
      GO TO (122,121),IND1                                              GOUD 141
  121 IF(COSW.EQ.1.0) 97,123                                            GOUD 142
  122 AGS=0.5                                                           GOUD 143
      LT=3                                                              GOUD 144
      RETURN                                                            GOUD 145
  123 IF(COSW.LE.COSA(2)) GO TO 125                                     GOUD 146
      IF(NCONV.EQ.1) GO TO 1161                                         GOUD 147
      IF(NCONX.EQ.1) GO TO 493                                          GOUD 148
  125 IF(COSW.LE.COSMAX) GO TO 400                                      GOUD 149
      PL2=COSW                                                          GOUD 150
      PL1=0.5*(3.0*PL2**2-1.0)                                          GOUD 151
      AGP=0.5+PL2*GX(1)+PL1*GX(2)                                       GOUD 152
      NTR=1                                                             GOUD 153
      DO 150L=3,100                                                     GOUD 154
      AL=L                                                              GOUD 155
      PL=((AL+AL-1.0)*COSW*PL1-(AL-1.0)*PL2)/AL                         GOUD 156
      IF(L-LMAXT) 132,132,130                                           GOUD 157
  130 LMAXT=L                                                           GOUD 158
      GL(L)=GLG(L,ETAI,ETABI,PAI,PBI,BI,F4X,F5X)                        GOUD 159
  132 IF(L-LMAX) 135,135,134                                            GOUD 160
  134 GX(L)=(AL+0.5)*EXP(FACT*GL(L))                                    GOUD 161
      IF(GX(L).GT.1.0E6) GO TO 491                                      GOUD 162
      LMAX=L                                                            GOUD 163
  135 DAGS=PL*GX(L)                                                     GOUD 164
      AGS=AGP+DAGS                                                      GOUD 165
      IF(ABS(PL)-1.0E-6) 145,145,138                                    GOUD 166
  138 IF(ABS(DAGS/AGS).LE.1.0E-5.OR.ABS(DAGS).LE.1.0E-7) 139,140        GOUD 167
  139 IF(NTR.EQ.2) GO TO 170                                            GOUD 168
      NTR=2                                                             GOUD 169
      AGL=AGS                                                           GOUD 170
      AGL1=AGP                                                          GOUD 171
      GO TO 145                                                         GOUD 172
  140 NTR=1                                                             GOUD 173
  145 PL2=PL1                                                           GOUD 174
      PL1=PL                                                            GOUD 175
      AGP=AGS                                                           GOUD 176
  150 CONTINUE                                                          GOUD 177
      COSMAX=COSW                                                       GOUD 178
      GO TO 400                                                         GOUD 179
  170 LT=L
  171 ADNM=AGS+AGL1-AGL-AGL
      IF(ABS(ADNM).LE.1.0E-12) GO TO 175
      AGS=(AGS*AGL1-AGL**2)/ADNM
  175 GO TO(180,1151,1152,176),NCON
  176 COSW=COSX
      AG5=AGS
      ECON=ALOG(AG0/AG5)/(1.0-COSA(2)**2)
      GO TO 493
  180 RETURN
  400 AMCON=1.0/(1.0-COSW)**MR
      AMTST=1.0E-7/AMCON
      IF(NYX.GE.1) GO TO 441
      AF(1,1)=0.5
      AF(1,2)=GX(1)
      AF(2,1)=0.5-0.333333333333*GX(1)
      AF(1,3)=GX(2)
      AF(2,2)=AF(1,2)-0.4*AF(1,3)-AF(1,1)
      K=2
      AF(3,1)=AF(2,1)-0.333333333333*AF(2,2)
      DO 440J=2,MR
      JM1=J-1
      K=K+1
      AK=K
      DO 430I=1,JM1
      AF(I,1)=AF(I,2)
      AF(I,2)=AF(I,3)
      IF(I.EQ.1) 427,428
  427 AF(1,3)=GX(K)
      GO TO 430
  428 IM1=I-1
      AK=AK-1.0
      AF(I,3)=AF(IM1,2)-(AK+1.0)*AF(IM1,3)/(AK+AK+3.0)-AK*AF(IM1,1)/(AK+
     1AK-1.0)
  430 CONTINUE
      AF(J,3)=AF(JM1,2)-.4285714285714*AF(JM1,3)-.666666666667*AF(JM1,1)
      AF(J+1,2)=AF(J,2)-0.4*AF(J,3)-AF(J,1)
      IF(J.EQ.MR) GO TO 440
      AF(J+2,1)=AF(J+1,1)-0.333333333333*AF(J+1,2)
  440 CONTINUE
      YXI=AF(MRP,1)
      YX(1)=AF(MRP,2)
  441 AGP=YXI+YX(1)*COSW
      PL2=1.0
      PL1=COSW
      NTR=1
      DO 490L=2,150
      AL=L
      PL=((AL+AL-1.0)*COSW*PL1-(AL-1.0)*PL2)/AL
      IF(L-NYX) 465,465,445
  445 K=K+1
      AK=K
      DO 460I=1,MR
      AF(I,1)=AF(I,2)
      AF(I,2)=AF(I,3)
      IF(I.EQ.1) 452,454
  452 IF(K-LMAXT) 4522,4522,4521
 4521 LMAXT=K
      GL(K)=GLG(K,ETAI,ETABI,PAI,PBI,BI,F4X,F5X)
 4522 IF(K-LMAX) 4525,4525,4524
 4524 GX(K)=(AK+0.5)*EXP(FACT*GL(K))
```

```
     IF(GX(K).GT.1.0E6) GO TO 491                                      GOUD 241
     LMAX=K                                                            GOUD 242
4525 AF(1,3)=GX(K)                                                     GOUD 243
     GO TO 460                                                         GOUD 244
 454 IM1=I-1                                                           GOUD 245
     AK=AK-1.0                                                         GOUD 246
     AF(I,3)=AF(IM1,2)-(AK+1.0)*AF(IM1,3)/(AK+AK+3.0)-AK*AF(IM1,1)/(AK+GOUD 247
    1AK-1.0)                                                           GOUD 248
 460 CONTINUE                                                          GOUD 249
     YX(L)=AF(MR,;)-(AL+1.0)*AF(MR,3)/(AL+AL+3.0)-AL*AF(MR,1)/(AL+AL-1.GOUD 250
    10)                                                                GOUD 251
     NYX=L                                                             GOUD 252
 465 DAGS=YX(L)*PL                                                     GOUD 253
     AGS=AGP+DAGS                                                      GOUD 254
     IF(ABS(PL)-1.0E-6) 475,475,468                                    GOUD 255
 468 IF(ABS(DAGS/AGS).LE.1.0E-5.OR.ABS(DAGS).LE.AMTST) 469,470        GOUD 256
 469 IF(NTR.EQ.2) GO TO 495                                            GOUD 257
     NTR=2                                                             GOUD 258
     AGL=AGS                                                           GOUD 259
     AGL1=AGP                                                          GOUD 260
     GO TO 475                                                         GOUD 261
 470 NTR=1                                                             GOUD 262
 475 PL2=PL1                                                           GOUD 263
     PL1=PL                                                            GOUD 264
     AGP=AGS                                                           GOUD 265
 490 CONTINUE                                                          GOUD 266
 491 IF(NCON.GT.1.OR.NCONV.GT.0.OR.COSW.LE.COSA(2)) GO TO 116         GOUD 267
     IF(AG5.GT.0.0) GO TO 493                                          GOUD 268
     NCON=4                                                            GOUD 269
     COSW=COSA(2)                                                      GOUD 270
     GO TO 400                                                         GOUD 271
 493 AGS=AG0*EXP(-ECON*(1.0-COSW**2))                                  GOUD 272
     LT=1                                                              GOUD 273
     RETURN                                                            GOUD 274
 495 LT=L+150                                                          GOUD 275
     AGS=AGS*AMCON                                                     GOUD 276
     AGL=AGL*AMCON                                                     GOUD 277
     AGL1=AGL1*AMCON                                                   GOUD 278
     GO TO 171                                                         GOUD 279
     END                                                               GOUD 280


     SUBROUTINE DEFINE                                                 DEFI 001
     COMMON/BLOCKG/RHO,Z,A,RHOZA,ALPHA,ALPHAS,AIMEAN,NANG,ANG(37),ANGR(DEFI 002
    137),COSA(37),NSTEP,E(152),TAU(152),T(152),RT,ST,NT,N1,N2,F1,F2,F3,DEFI 003
    2F4,F5,F6,F7,IK,C1,C1X,BM(10,10),P(10),GX(160),JX,JH,JP,JH1,JH2,IOPDEFI 004
    3T,ASCOR,BSCOR,DSCOR,NCRV                                          DEFI 005
     DIMENSION ZT(12),AT(12),RHOT(12),AIMT(12)                         DEFI 006
     DATA(ZT(J),J:1,11)/3.0,4.0,6.0,12.0,13.0,26.0,29.0,47.0,50.0,82.0,DEFI 007
    192.0/                                                             DEFI 008
     DATA(RHOT(J),J=1,11)/0.53,1.816,2.26,1.74,2.7,7.86,8.92,10.5,7.31,DEFI 009
    111.337,18.485/                                                    DEFI 010
     DATA(AT(J),J=1,11)/6.94,9.02,12.01,24.32,26.97,55.85,63.57,107.88,DEFI 011
    1118.7,207.21,238.07/                                              DEFI 012
     DATA(AIMT(J),J=1,11)/3.9E-5,6.41E-5,7.81E-5,1.56E-4,1.63E-4,3.37E-DEFI 013
    14,3.77E-4,6.60E-4,7.09E-4,1.18E-3,1.325E-3/                       DEFI 014
     DATA SQRT2,PI/1.414214,3.1415927/                                 DEFI 015
     DATA NANG,NANG1,ANG(1),ANGR(1),COSA(1)/37,36,0.0,0.0,1.0/         DEFI 016
     DATA IKL,NML,N1,N2/0,0,1,3/                                       DEFI 017
     N1 MUST BE 1, N2 MAY BE BETWEEN (AND INCLUDING) 2 AND 10          DEFI 018
     Z=ZT(NCRV)                                                        DEFI 019
```

```
      A=AT(NCRV)                                                        DEFI 020
      RHO=RHOT(NCRV)                                                    DEFI 021
      AIMEAN=AIMT(NCRV)                                                 DEFI 022
      RHOZA=0.301314*RHO*Z/A                                            DEFI 023
      C1=0.130548/AIMEAN**2                                             DEFI 024
      C1X=RHOZA*(Z+1.0)                                                 DEFI 025
      FACT=PI/180.0                                                     DEFI 026
      DO 21J=2,NANG1                                                    DEFI 027
      ANG(J)=ANG(J-1)+5.0                                               DEFI 028
      ANGR(J)=ANG(J)*FACT                                               DEFI 029
      COSA(J)=COS(ANGR(J))                                              DEFI 030
      IF(ABS(COSA(J))-1.0E-5) 20,20,21                                  DEFI 031
   20 COSA(J)=0.0                                                       DEFI 032
   21 CONTINUE                                                          DEFI 033
      ANG(NANG)=179.0                                                   DEFI 034
      ANGR(NANG)=ANG(NANG)*FACT                                         DEFI 035
      COSA(NANG)=COS(ANGR(NANG))                                        DEFI 036
      IK=7                                                              DEFI 037
C     IK MUST NEVER BE LESS THAN 3 OR MORE THAN 9                       DEFI 038
      JX=IK/2                                                           DEFI 039
      JH=JX+1                                                           DEFI 040
      JP=JX+JH                                                          DEFI 041
      JH1=JH+1                                                          DEFI 042
      JH2=JH+2                                                          DEFI 043
      IF(JX-(IK+1)/2) 31,32,32                                          DEFI 044
   31 IOPT=2                                                            DEFI 045
      GO TO 33                                                          DEFI 046
   32 IOPT=1                                                            DEFI 047
   33 ALPHA=Z/137.0                                                     DEFI 048
      ALPHAS=ALPHA**2                                                   DEFI 049
      Z2=Z*Z                                                            DEFI 050
      Z3=Z*Z2                                                           DEFI 051
      ASCOR=SQRT(Z)                                                     DEFI 052
      BSCOR=Z2*(1.0+2.0/Z)/(1.0+Z2/30.0)                                DEFI 053
      DSCOR=36.0*(Z3+200.0)/((Z3+3000.0)*(10.0+Z))                      DEFI 054
      F6=Z**(2.0/3.0)                                                   DEFI 055
      F1=1.7E-5*F6                                                      DEFI 056
      F2=3.76*ALPHAS                                                    DEFI 057
      F3=PI*ALPHA/SQRT2                                                 DEFI 058
      F4=0.160*F6                                                       DEFI 059
      F7=3.33*ALPHA                                                     DEFI 060
      RETURN                                                            DEFI 061
      END                                                               DEFI 062


      SUBROUTINE TTAB(EI,EMIN)                                          TTAB 001
      COMMON/BLOCKG/RHO,Z,A,RHOZA,ALPHA,ALPHAS,AIMEAN,NANG,ANG(37),ANGR(TTAB 002
     137),COSA(37),NSTEP,E(152),TAU(152),T(152),RT,ST,NT,N1,N2,F1,F2,F3,TTAB 003
     2F4,F5,F6,F7,IK,C1,C1X,BM(10,10),P(10),GX(160),JX,JH,JP,JH1,JH2,IOPTTAB 004
     3T,ASCOR,BSCOR,DSCOR,NCRV                                          TTAB 005
      DIMENSION S(152)                                                  TTAB 006
      EXTERNAL DSCAL                                                    TTAB 007
      DATA FX/1.95703907/                                               TTAB 008
      IF(NSTEP-150) 11,11,10                                            TTAB 009
   10 NSTEP=150                                                         TTAB 010
      PRINT 1                                                           TTAB 011
    1 FORMAT(1H0*TTAB HAS RESET NSTEP TO 150*)                          TTAB 012
   11 HSTEP=(EI-EMIN)/FLOAT(NSTEP)                                      TTAB 013
      NST1=NSTEP+1                                                      TTAB 014
      NT=NSTEP+2                                                        TTAB 015
      NORD=12                                                           TTAB 016
      IND=1
      E(1)=EI
      TAU(1)=E(1)*FX
      DO 20J=2,NST1
      E(J)=E(J-1)-HSTEP
      IF(E(J).LE.0.1.AND.IND.EQ.1) 15,16
   15 NC=J
      IND=2
   16 TAU(J)=E(J)*FX
   20 CONTINUE
   30 E(NT)=0.0
      TAU(NT)=0.0
      S(1)=0.0
      DO 50J=2,NT
      IF(J.EQ.NC) NORD=20
      BMIN=TAU(J)
      BMAX=TAU(J-1)
   50 S(J)=S(J-1)+GAUS(DSCAL,BMIN,BMAX,NORD,1)
      ST=S(NT)
      T(1)=1.0
      FY=1.0/ST
      DO 80J=2,NST1
   80 T(J)=FY*(ST-S(J))
      T(NT)=0.0
      RETURN
      END


      FUNCTION DENCOR(TAUX,NCRV)
C     DENSITY CORRECTION FROM NELMS TABLES
C     NCRV=1(LITHIUM-Z=3),2(BE-4),3(C-6),4(MG-12),5(AL-13),6(FE-26),7(CU-29),
C     8(AG-47),9(SN-50 OR AU-79),10(PB-82),11(U-92)
      DIMENSION ET(15),DENLIM(11),DELC(15,11)
      DATA(ET(J),J: 1,15)/0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0,2.0,3.0,5.0
     1,7.0,8.5,10.(/
      DATA(DENLIM(J),J=1,11)/0.0,0.05,0.04,0.03,0.05,0.05,0.035,0.015,0.
     103,0.0,0.0/
      DATA (DELC(J),J=1,15) /.01,.10,.16,.22,.28,.348,.41,.47,.52,1.02,1
     1.45,2.03,2.51,2.83,3.1/
      DATA (DELC(J),J=16,30)/.08,.145,.221,.3,.375,.45,.52,.59,.65,1.2,1
     1.64,2.3,2.79,3.1,3.35/
      DATA (DELC(J),J=31,45)/.058,.1,.16,.22,.28,.345,.41,.47,.52,1.02,1
     1.43,2.03,2.5,2.8,3.05/
      DATA (DELC(J),J=46,60)/.04,.042,.044,.048,.058,.075,.1,.126,.152,.
     1452,.77,1.25,1.64,1.88,2.1/
      DATA(DELC(J),J=61,75)/.05,.051,.061,.09,.12,.158,.196,.236,.275,.6
     15,.99,1.52,1.92,2.17,2.35/
      DATA(DELC(J),J=76,90)/.05,.05,.055,.068,.088,.115,.145,.178,.210,.
     154,.83,1.31,1.68,1.93,2.15/
      DATA(DELC(J),J=91,105)/.035,.036,.038,.04,.05,.061,.08,.105,.13,.4
     13,.71,1.18,1.56,1.79,2.0/
      DATA(DELC(J),J=106,120)/6*.015,.02,.029,.04,.21,.42,.76,1.05,1.25,
     11.43/
      DATA(DELC(J),J=121,135)/3*.03,.033,.037,.04,.045,.052,.061,.15,.35
     1,.70,.95,1.15,1.25/
      DATA(DELC(J),J=136,150)/9*0.0,.11,.23,.48,.71,.87,1.01/
      DATA(DELC(J),J=151,165)/9*0.0,.16,.32,.60,.85,1.02,1.18/
   11 E=.510976*TAUX
      IF(E-ET(1)) 15,20
   15 DENCOR=DENLIM(NCRV)
      RETURN
   20 IF(E-10.0) 30,30,25
   25 PRINT 1
    1 FORMAT(1H010X*E IS TOO LARGE FOR DENCOR TO HANDLE*)
      STOP
   30 DO 50J=3,15,2
      IF(E-ET(J)) 55,55,50
   50 CONTINUE
   55 P=(E-ET(J-1))/(ET(J)-ET(J-1))
      DENCOR=0.5*P*(P-1.0)*DELC(J-2,NCRV)+(1.0-P*P)*DELC(J-1,NCRV)+0.5*P
     1*(P+1.0)*DELC(J,NCRV)
      RETURN
      END


      FUNCTION DSCAL(TAU)
      CALL DMUS(TAU,DEDS)
      DSCAL=1.0/ABS(DEDS)
      RETURN
      END


      SUBROUTINE DMUS(TAUX,DEDS)
C     CALCULATES DEDS=DE/DS FOR TAUX=E/.510976
      COMMON/BLOCKG/RHO,Z,A,RHOZA,ALPHA,ALPHAS,AIMEAN,NANG,ANG(37),ANGR(
     137),COSA(37),NSTEP,E(152),TAU(152),T(152),RT,ST,NT,N1,N2,F1,F2,F3,
     2F4,F5,F6,F7,IK,C1,C1X,BM(10,10),P(10),GX(160),JX,JH,JP,JH1,JH2,IOP
     3T,ASCOR,BSCOR,DSCOR,NCRV
      CA=1.0/(TAUX+1.0)**2
      CB=TAUX**2
      BETAS=CA*TAUX*(TAUX+2.0)
      FMINUS=CA+CA*CB/8.0-(TAUX+TAUX+1.0)*CA*0.693147
      X=BETAS*Z/ALPHAS
      SCOR=ASCOR*(1.0-DSCOR/X**2)/(X+BSCOR)
C     SCOR=SHELL CORRECTION
C     DENCOR=DENSITY CORRECTION
      DEDS=-RHOZA*(ALOG(C1*CB*(TAUX+2.0))+FMINUS-SCOR-SCOR-DENCOR(TAUX,N
     1CRV))/BETAS
      DEDS=DEDS*(1.0+Z*TAUX/1400.0)
      RETURN
      END


      SUBROUTINE HFIT(N,BETA,BETAS,ETA,Q,FP)
      COMMON/BLOCKG/RHO,Z,A,RHOZA,ALPHA,ALPHAS,AIMEAN,NANG,ANG(37),ANGR(
     137),COSA(37),NSTEP,E(152),TAU(152),T(152),RT,ST,NT,N1,N2,F1,F2,F3,
     2F4,F5,F6,F7,IK,C1,C1X,BM(10,10),P(10),GX(160),JX,JH,JP,JH1,JH2,IOP
     3T,ASCOR,BSCOR,DSCOR,NCRV
      BETAS=TAU(N)*(TAU(N)+2.0)/(TAU(N)+1.0)**2
      BETA=SQRT(BETAS)
      ETA=F1*(1.13+F2/BETAS)/(TAU(N)*(TAU(N)+2.0))
      Q=ALPHA/BETA
      FP=F3*BETA*CGAM(Q)
      IKM=IK-1
      IKS=NANG/IKM
      JN=1
      BM(1,1)=SQRT(1.0-COSA(JN)+ETA+ETA)
      CALL RMOTT(Z,E(N),COSA(JN),ANGR(JN),R,SIGM,SIGR,1)
      P(1)=R-1.0-FP*BM(1,1)
      DO 35J=2,IKM
      JN=JN+IKS
      BM(J,1)=SQRT(1.0-COSA(JN)+ETA+ETA)
      CALL RMOTT(Z,E(N),COSA(JN),ANGR(JN),R,SIGM,SIGR,2)                HFIT 020
   35 P(J)=R-1.0-FP*BM(J,1)                                             HFIT 021
      BM(IK,1)=SQRT(1.0-COSA(NANG)+ETA+ETA)                             HFIT 022
      CALL RMOTT(Z,E(N),COSA(NANG),ANGR(NANG),R,SIGM,SIGR,2)            HFIT 023
      P(IK)=R-1.0-FP*BM(IK,1)                                           HFIT 024
      DO 50J=1,IK                                                       HFIT 025
      DO 50I=2,IK                                                       HFIT 026
   50 BM(J,I)=BM(J,I-1)*BM(J,1)                                         HFIT 027
      CALL LEQ(BM,P,IK,1,10,10,DET)                                     HFIT 028
      RETURN                                                            HFIT 029
      END                                                               HFIT 030


      SUBROUTINE RANGEN(INDEX,RN,EN,TN)                                 RANG 001
C     INDEX=1 FOR TOTAL RANGE RN FOR INITIAL ENERGY EN. (TN=1.0)
C     INDEX=2,3,OR 4 -- RN IS THE PORTION OF THE RANGE TRAVERSED BETWEEN EI
C        AND FINAL ENERGY EN. INDEX=2 FOR EN(RN),TN(EN). INDEX=3 FOR JUST
C        EN(RN). INDEX=4 FOR EN(TN),RN(EN).
      COMMON/BLOCK( /RHO,Z,A,RHOZA,ALPHA,ALPHAS,AIMEAN,NANG,ANG(37),ANGR(RANG 002
     137),COSA(37),NSTEP,E(152),TAU(152),T(152),RT,ST,NT,N1,N2,F1,F2,F3,RANG 003
     2F4,F5,F6,F7,IK,C1,C1X,BM(10,10),P(10),GX(160),JX,JH,JP,JH1,JH2,IOPRANG 004
     3T,ASCOR,BSCOR,DSCOR,NCRV                                          RANG 005
      GO TO (190,10,10,150),INDEX                                       RANG 006
   10 RX=RHO*(RT-RN)                                                    RANG 007
      IF(RX-0.95) 12,13,13                                              RANG 008
   12 EN=EXP(6.629979035639-5.241090146751*SQRT(1.261848095652-0.3816*ALRANG 009
     1OG(RX)))                                                          RANG 010
      IF(INDEX.EQ.2) 20,100                                             RANG 011
   13 EN=1.886792453*RX+0.2                                             RANG 012
      IF(INDEX.EQ.2) 20,100                                             RANG 013
   20 DO 30J=2,NT                                                       RANG 014
      IF(EN-E(J)) 30,35,35                                              RANG 015
   30 CONTINUE                                                          RANG 016
   35 TN=T(J)+(T(J-1)-T(J))*(EN-E(J))/HSTEP                             RANG 017
  100 RETURN                                                            RANG 018
  150 DO 160J=2,NT                                                      RANG 019
      IF(TN-T(J)) 160,165,165                                           RANG 020
  160 CONTINUE                                                          RANG 021
  165 JM1=J-1                                                           RANG 022
      EN=E(J)+(E(JM1)-E(J))*(TN-T(J))/(T(JM1)-T(J))                     RANG 023
      IF(EN-2.0) 170,175,175                                            RANG 024
  170 RN=RT-.412*EN**(1.265-.0954*ALOG(EN))/RHO                         RANG 025
      RETURN                                                            RANG 026
  175 RN=RT-(0.53*EN-0.106)/RHO                                         RANG 027
      RETURN                                                            RANG 028
  190 IF(EN-2.0) 195,200,200                                            RANG 029
  195 RN=0.412*EN**(1.265-0.0954*ALOG(EN))/RHO                          RANG 030
      RETURN                                                            RANG 031
  200 RN=(0.53*EN-0.106)/RHO                                            RANG 032
      RETURN                                                            RANG 033
      END                                                               RANG 034


      FUNCTION CGAM(Q)                                                  CGAM 001
      COMPLEX C1,C2,COMPLX,CONJGX,GAMCAR                                CGAM 002
      C1=GAMCAR(COMPLX(0.5,Q))                                          CGAM 003
      C2=GAMCAR(COMPLX(1.0,Q))                                          CGAM 004
      CGAM=REAL(C2*CONJGX(C1)/(C1*CONJGX(C2)))                          CGAM 005
      RETURN                                                            CGAM 006
      END                                                               CGAM 007
      FUNCTION G1G(ETA,ETAB,PA,PB,B,FP,FQ,FR)                           G1G  001
      COMMON/BLOCKG/RHO,Z,A,RHOZA,ALPHA,ALPHAS,AIMEAN,NANG,ANG(37),ANGR(G1G  002
     137),COSA(37),NSTEP,E(152),TAU(152),T(152),RT,ST,NT,N1,N2,F1,F2,F3,G1G  003
     2F4,F5,F6,F7,IK,C1,C1X,BM(10,10),P(10),GX(160),JX,JH,JP,JH1,JH2,IOPG1G  004
     3T,ASCOR,BSCOR,DSCOR,NCRV                                          G1G  005
      DIMENSION PA(5,7),PB(5,7),B(10)                                   G1G  006
      PA(1,1)=ALOG(FP)-FQ                                               G1G  007
      PB(1,1)=2.0*(ETAB+ETAB)**1.5/(1.0+ETAB)                           G1G  008
      PA(1,2)=3.0*(FR*PA(1,1)-FQ)                                       G1G  009
      PB(1,2)=PB(1,1)*(1.0+ETAB)                                        G1G  010
      DO 50L=3,JH                                                       G1G  011
      AL=L-1                                                            G1G  012
      PA(1,L)=((AL+AL+1.0)*(FR*PA(1,L-1)-FQ)-(AL+1.0)*PA(1,L-2))/AL     G1G  013
   50 PB(1,L)=ETAB*PB(1,L-1)+PB(1,1)                                    G1G  014
      DO 60J=2,JX                                                       G1G  015
      JM1=J-1                                                           G1G  016
      PA(J,1)=(1.0+FR)*PA(JM1,1)-0.666667*PA(JM1,2)                     G1G  017
      PB(J,1)=(1.0+FR)*PB(JM1,1)-0.666667*PB(JM1,2)                     G1G  018
      LF=JH-J+1                                                         G1G  019
      DO 60L=2,LF                                                       G1G  020
      AL=L                                                              G1G  021
      DFACT=1.0/(AL+AL+1.0)                                             G1G  022
      PA(J,L)=FR*PA(JM1,L)+PA(JM1,1)-((AL+1.0)*PA(JM1,L+1)+AL*PA(JM1,L-1G1G  023
     1))*DFACT                                                          G1G  024
   60 PB(J,L)=FR*PB(JM1,L)+PB(JM1,1)-((AL+1.0)*PB(JM1,L+1)+AL*PB(JM1,L-1G1G  025
     1))*DFACT                                                          G1G  026
      PA(JH,1)=(1.0+FR)*PA(JX,1)-0.66666666666667*PA(JX,2)              G1G  027
      PB(JH,1)=(1.0+FR)*PB(JX,1)-0.66666666666667*PB(JX,2)              G1G  028
      G1G=0.0                                                           G1G  029
      DO 75J=1,JX                                                       G1G  030
      JJ=J+J-1                                                          G1G  031
   75 G1G=G1G+B(JJ)*PA(J,1)+B(JJ+1)*PB(J.1)                             G1G  032
      GO TO (80,85),IOPT                                                G1G  033
   80 G1G=G1G+B(JP)*PA(JH,1)                                            G1G  034
      RETURN                                                            G1G  035
   85 G1G=G1G+B(JP)*PA(JH,1)+B(JP+1)*PB(JH,1)                           G1G  036
      RETURN                                                            G1G  037
      END                                                               G1G  038


      FUNCTION GLG(L,ETA,ETAB,PA,PB,B,FP,FQ)                            GLG  001
      COMMON/BLOCKG/RHO,Z,A,RHOZA,ALPHA,ALPHAS,AIMEAN,NANG,ANG(37),ANGR(GLG  002
     137),COSA(37),NSTEP,E(152),TAU(152),T(152),RT,ST,NT,N1,N2,F1,F2,F3,GLG  003
     2F4,F5,F6,F7,IK,C1,C1X,BM(10,10),P(10),GX(160),JX,JH,JP,JH1,JH2,IOPGLG  004
     3T,ASCOR,BSCOR,DSCOR,NCRV                                          GLG  005
      DIMENSION PA(5,7),PB(5,7),B(10)                                   GLG  006
      AL2=L+JH-2                                                        GLG  007
      PA(1,JH2)=((AL2+AL2+1.0)*(FQ*PA(1,JH1)-FP)-(AL2+1.0)*PA(1,JH))/AL2GLG  008
      PB(1,JH2)=ETAB*PB(1,JH1)+PB(1,1)                                  GLG  009
      DO 50J=2,JH                                                       GLG  010
      LF=JH2-J+1                                                        GLG  011
      ALF=LF+L-3                                                        GLG  012
      JM1=J-1                                                           GLG  013
      PA(J,LF)=FQ*PA(JM1,LF)+PA(JM1,1)-((ALF+1.0)*PA(JM1,LF+1)+ALF*PA(JMGLG  014
     11,LF-1))/(ALF+ALF+1.0)                                            GLG  015
   50 PB(J,LF)=FQ*PB(JM1,LF)+PB(JM1,1)-((ALF+1.0)*PB(JM1,LF+1)+ALF*PB(JMGLG  016
     11,LF-1))/(ALF+ALF+1.0)                                            GLG  017
      GLG=0.0                                                           GLG  018
      DO 75J=1,JX                                                       GLG  019
      JJ=J+J-1                                                          GLG  020
   75 GLG=GLG+B(JJ)*PA(J,3)+B(JJ+1)*PB(J,3)                             GLG  021
      GO TO (80,85),IOPT                                                GLG  022
   80 GLG=GLG+B(JP)*PA(JH,3)                                            GLG  023
      GO TO 100                                                         GLG  024
   85 GLG=GLG+B(JP)*PA(JH,3) +B(JP+1)*PB(JH,3)                          GLG  025
  100 DO 110J=2,JH1                                                     GLG  026
      LF=JH1-J+1                                                        GLG  027
      DO 110I=1,LF                                                      GLG  028
      PA(I,J)=PA(I,J+1)                                                 GLG  029
  110 PB(I,J)=PB(I,J+1)                                                 GLG  030
      RETURN                                                            GLG  031
      END                                                               GLG  032


      SUBROUTINE RMOTT(Z,E,COSA,THETA,R,SIGM,SIGR,INDEX)                RMOT 001
C     Z=ATOMIC NUMBER, E=ENERGY(MEV),THETA=ANGLE(RADIANS),COSA=COS(THETA)
C     INDEX=1 FOR NEW Z OR E, OTHERWISE=2
C     SIGM=MOTT CROSS SECTION, SIGR=RUTHERFORD CROSS SECTION
C     R IS THE CALCULATED VALUE OF SIGMA(MOTT)/SIGMA(RUTHERFORD)
C     CODE USES ASYMPTOTIC FORMULA FOR THETA LE TEN DEGREES
      DATA PI,SQRT2,TOL/3.1415927,1.414213562,5.0E-6/                   RMOT 002
      COMPLEX C1,C2,C3,C4,C5,C6,F,G,DKF,DKG,F0,G0,UF0,UG0,AF(7,3),AG(7,3RMOT 003
     1),FCON,UF(100),UG(100)                                            RMOT 004
      COMPLEX COMPLX,CONJGX,CEXP,GAMCAR                                 RMOT 005
    1 FORMAT(1H010X*RMOTT HAS REPLACED 180 DEG. WITH 179 DEG. TO AVOID BRMOT 006
     1LOWUP.*/)                                                         RMOT 007
    3 FORMAT(1H0*SERIES IN RMOTT FOR THETA =*E11.4,* RADIANS DOES NOT CORMOT 008
     1NVERGE IN 90 TERMS.*)                                             RMOT 009
      GO TO (15,100),INDEX                                              RMOT 010
   15 ALPHA=Z/137.0                                                     RMOT 011
      ALPHAS=ALPHA**2                                                   RMOT 012
      IF(Z-50) 16,16,17                                                 RMOT 013
   16 MR=3                                                              RMOT 014
      MRP=4                                                             RMOT 015
      GO TO 19                                                          RMOT 016
   17 MR=4                                                              RMOT 017
      MRP=5                                                             RMOT 018
   19 TAU=E/0.510976                                                    RMOT 019
      BETAS=TAU*(TAU+2.0)/(TAU+1.0)**2                                  RMOT 020
      BETA=SQRT(BETAS)                                                  RMOT 021
      Q=ALPHA/BETA                                                      RMOT 022
      SLAMB=1.494E-21/(TAU*(TAU+2.0))                                   RMOT 023
      C1=COMPLX(1.( ,Q)                                                 RMOT 024
      C2=GAMCAR(C1)                                                     RMOT 025
      FCON=CONJGX(C2)/C2                                                RMOT 026
      RHOK=SQRT(1.0-ALPHAS)                                             RMOT 027
      C4=COMPLX(RHOK,Q)                                                 RMOT 028
      C5=GAMCAR(C4)                                                     RMOT 029
      C6=C1*C2                                                          RMOT 030
      DKF=CEXP(COMPLX(0.0,-PI))*CONJGX(C2)/C6 - CEXP(COMPLX(0.0,-PI*RHOKRMOT 031
     1))*CONJGX(C5)/(C4*C5)                                             RMOT 032
      DKG=DKF                                                           RMOT 033
      CGQ=PI*ALPHA*BETA*CGAM(Q)/SQRT2                                   RMOT 034
      AF(1,1)=DKF                                                       RMOT 035
      AG(1,1)=-DKF                                                      RMOT 036
      CALL RM1(1,ALPHAS,Q,DKF,DKG,C2,C4,C5,C6,AF(1,2),AG(1,2))          RMOT 037
      AF(1,2)=-AF(1,2)                                                  RMOT 038
      AG(1,2)=-AG(1,2)                                                  RMOT 039
      AF(2,1)=AF(1,1)-0.333333333333*AF(1,2)                            RMOT 040
      AG(2,1)=AG(1,1)-0.333333333333*AG(1,2)                            RMOT 041
      CALL RM1(2,ALPHAS,Q,DKF,DKG,C2,C4,C5,C6,AF(1,3),AG(1,3))          RMOT 042
      AF(2,2)=AF(1,2)-0.4*AF(1,3)-AF(1,1)                               RMOT 043
      AG(2,2)=AG(1,2)-0.4*AG(1,3)-AG(1,1)                               RMOT 044
      AF(3,1)=AF(2,1)-0.333333333333*AF(2,2)                            RMOT 045
      AG(3,1)=AG(2,1)-0.333333333333*AG(2,2)                            RMOT 046
      SN=1.0                                                            RMOT 047
      K=2                                                               RMOT 048
      DO 40J=2,MR                                                       RMOT 049
      JM1=J-1                                                           RMOT 050
      K=K+1                                                             RMOT 051
      AL=K                                                              RMOT 052
      DO 30I=1,JM1                                                      RMOT 053
      AF(I,1)=AF(I,2)                                                   RMOT 054
      AG(I,1)=AG(I,2)                                                   RMOT 055
      AF(I,2)=AF(I,3)                                                   RMOT 056
      AG(I,2)=AG(I,3)                                                   RMOT 057
      IF(I.EQ.1) 27,28                                                  RMOT 058
   27 CALL RM1(K,ALPHAS,Q,DKF,DKG,C2,C4,C5,C6,AF(1,3),AG(1,3))          RMOT 059
      SN=-SN                                                            RMOT 060
      AF(1,3)=AF(1,3)*SN                                                RMOT 061
      AG(1,3)=AG(1,3)*SN                                                RMOT 062
      GO TO 30                                                          RMOT 063
   28 IM1=I-1                                                           RMOT 064
      AL=AL-1.0                                                         RMOT 065
      AF(I,3)=AF(IM1,2)-(AL+1.0)*AF(IM1,3)/(AL+AL+3.0)-AL*AF(IM1,1)/(AL+RMOT 066
     1AL-1.0)                                                           RMOT 067
      AG(I,3)=AG(IM1,2)-(AL+1.0)*AG(IM1,3)/(AL+AL+3.0)-AL*AG(IM1,1)/(AL+RMOT 068
     1AL-1.0)                                                           RMOT 069
   30 CONTINUE                                                          RMOT 070
      AF(J,3)=AF(JM1,2)-.4285714285714*AF(JM1,3)-.666666666667*AF(JM1,1)RMOT 071
      AG(J,3)=AG(JM1,2)-.4285714285714*AG(JM1,3)-.666666666667*AG(JM1,1)RMOT 072
      AF(J+1,2)=AF(J,2)-0.4*AF(J,3)-AF(J,1)                             RMOT 073
      AG(J+1,2)=AG(J,2)-0.4*AG(J,3)-AG(J,1)                             RMOT 074
      IF(J.EQ.MR) GO TO 40                                              RMOT 075
      AF(J+2,1)=AF(J+1,1)-0.333333333333*AF(J+1,2)                      RMOT 076
      AG(J+2,1)=AG(J+1,1)-0.333333333333*AG(J+1,2)                      RMOT 077
   40 CONTINUE                                                          RMOT 078
      JMAX=1                                                            RMOT 079
      UF0=AF(MRP,1)                                                     RMOT 080
      UG0=AG(MRP,1)                                                     RMOT 081
      UF(1)=AF(MRP,2)                                                   RMOT 082
      UG(1)=AG(MRP,2)                                                   RMOT 083
  100 IF(COSA-1.0) 110,105,105                                          RMOT 084
  105 R=1.0                                                             RMOT 085
      RETURN                                                            RMOT 086
  110 SIGR=1.4943E-21*ALPHAS*(1.0-BETAS)/(BETAS**2*(1.0-COSA)**2)       RMOT 087
      IF(THETA-0.175) 115,115,120                                       RMOT 088
  115 R=1.0+CGQ*SQRT(1.0-COSA)                                          RMOT 089
      SIGM=R*SIGR                                                       RMOT 090
      RETURN                                                            RMOT 091
  120 IF(COSA+0.99999) 125,125,130                                      RMOT 092
  125 PRINT 1                                                           RMOT 093
      THETA=179.0*PI/180.0                                              RMOT 094
      COSA=COS(THETA)                                                   RMOT 095
  130 ANG=0.5*THETA                                                     RMOT 096
      SINS=SIN(ANG)**2                                                  RMOT 097
      SECS=1.0/(1.0-SINS)                                               RMOT 098
      F0=FCON*(1.0-COSA)**MR*CEXP(COMPLX(0.0,Q*ALOG(SINS)))             RMOT 099
      G0=COMPLX(0.0,-Q/(SINS*SECS))*F0                                  RMOT 100
      F=F0+UF0+UF(1)*COSA                                               RMOT 101
      G=G0+UG0+UG(1)*COSA                                               RMOT 102
      FLAST=CABS(F)                                                     RMOT 103
      GLAST=CABS(G)                                                     RMOT 104
      PK2=1.0                                                           RMOT 105
      PK1=COSA                                                          RMOT 106
      NTRY=1                                                            RMOT 107
      DO 200J=2,90                                                      RMOT 108
      AJ=J                                                              RMOT 109
      JM1=J-1                                                           RMOT 110
      PK=((AJ+AJ-1.0)*COSA*PK1-(AJ-1.0)*PK2)/AJ                         RMOT 111
      IF(J-JMAX) 160,160,140                                            RMOT 112
  140 JMAX=J                                                            RMOT 113
      K=K+1                                                             RMOT 114
      AL=K                                                              RMOT 115
      DO 155I=1,MR                                                      RMOT 116
      AF(I,1)=AF(I,2)                                                   RMOT 117
      AG(I,1)=AG(I,2)                                                   RMOT 118
      AF(I,2)=AF(I,3)                                                   RMOT 119
      AG(I,2)=AG(I,3)                                                   RMOT 120
      IF(I.EQ.1) 152,154                                                RMOT 121
  152 CALL RM1(K,ALPHAS,Q,DKF,DKG,C2,C4,C5,C6,AF(1,3),AG(1,3))          RMOT 122
      SN=-SN                                                            RMOT 123
      AF(1,3)=AF(1,3)*SN                                                RMOT 124
      AG(1,3)=AG(1,3)*SN                                                RMOT 125
      GO TO 155                                                         RMOT 126
  154 IM1=I-1                                                           RMOT 127
      AL=AL-1.0                                                         RMOT 128
      AF(I,3)=AF(IM1,2)-(AL+1.0)*AF(IM1,3)/(AL+AL+3.0)-AL*AF(IM1,1)/(AL+RMOT 129
     1AL-1.0)                                                           RMOT 130
      AG(I,3)=AG(IM1,2)-(AL+1.0)*AG(IM1,3)/(AL+AL+3.0)-AL*AG(IM1,1)/(AL+RMOT 131
     1AL-1.0)                                                           RMOT 132
  155 CONTINUE                                                          RMOT 133
      UF(J)=AF(MR,2)-(AJ+1.0)*AF(MR,3)/(AJ+AJ+3.0)-AJ*AF(MR,1)/(AJ+AJ-1.RMOT 134
     10)                                                                RMOT 135
      UG(J)=AG(MR,2)-(AJ+1.0)*AG(MR,3)/(AJ+AJ+3.0)-AJ*AG(MR,1)/(AJ+AJ-1.RMOT 136
     10)                                                                RMOT 137
  160 F=F+UF(J)*PK                                                      RMOT 138
      G=G+UG(J)*PK                                                      RMOT 139
      FABS=CABS(F)                                                      RMOT 140
      GABS=CABS(G)                                                      RMOT 141
      IF(ABS(PK)-1.0E-6) 176,176,165                                    RMOT 142
  165 IF(ABS(FABS-FLAST)/FABS-TOL) 170,170,175                          RMOT 143
  170 IF(ABS(GABS-GLAST)/GABS-TOL) 171,171,175                          RMOT 144
  171 IF(NTRY.EQ.2) GO TO 210                                           RMOT 145
      NTRY=2                                                            RMOT 146
      GO TO 176                                                         RMOT 147
  175 NTRY=1                                                            RMOT 148
  176 FLAST=FABS                                                        RMOT 149
      GLAST=GABS                                                        RMOT 150
      PK2=PK1                                                           RMOT 151
  200 PK1=PK                                                            RMOT 152
      PRINT 3,THETA                                                     RMOT 153
      STOP                                                              RMOT 154
  210 C3=(0.0,0.5)/(1.0-COSA)**MR                                       RMOT 155
      F=C3*F                                                            RMOT 156
      G=C3*G                                                            RMOT 157
      FABS=CABS(F)                                                      RMOT 158
      GABS=CABS(G)                                                      RMOT 159
      SIGM =SLAMB*(Q**2*(1.0-BETAS)*FABS**2/SINS + GABS**2*SECS)        RMOT 160
      R=SIGM/SIGR                                                       RMOT 161
      RETURN                                                            RMOT 162
      END                                                               RMOT 163
      SUBROUTINE RM1(K,ALPHAS,Q,DKF,DKG,C2,C4,C5,C6,AF,AG)
      COMPLEX DKF,DKG,DK1F,DK1G,C2,C4,C5,C6,AF,AG
      COMPLEX COMPLX,CONJGX,GAMCAR,CEXP
      AK1=K+1
      RHOK=SQRT(AK1**2-ALPHAS)
      C2=C6
      C4=COMPLX(RHOK,Q)
      C5=GAMCAR(C4)
      C6=COMPLX(AK1,Q)*C2
      DK1F=AK1*(CEXP(COMPLX(0.0,-3.1415927*AK1))*CONJGX(C2)/C6 - CEXP(CO
     1MPLX(0.0,-3.1415927*RHOK))*CONJGX(C5)/(C4*C5))
      DK1G=AK1*DK1F
      AF=DKF+DK1F
      AG=DKG-DK1G
      DKF=DK1F
      DKG=DK1G
      RETURN
      END


      COMPLEX FUNCTION GAMCAR(CMX)
      COMPLEX CMX,COMX,GAMX,E1,E11,TX,GAM1,CLOG,CEXP
      COMX=CMX-1.
      E1=COMX+.5
      TX=COMX+5.5
      E11=E1*CLOG(TX)
      GAM1=CEXP(E11-TX)*2.50662827465
      GAMX=GAM1*(1.+76.18009173/(COMX+1.)-86.50532033/(COMX+2.)
     G+24.01409822/(COMX+3.)-1.231739516/(COMX+4.)+.120858003E-2/(COMX+
     G5.)-.536382E-5/(COMX+6.))
      GAMCAR=GAMX
      RETURN
      END


      COMPLEX FUNCTION COMPLX(A,B)
      COMPLX=A+(0.0,1.0)*B
      RETURN
      END


      COMPLEX FUNCTION CONJGX(C)
      COMPLEX C
      XR=REAL(C)
      XIM=AIMAG(C)
      CONJGX=XR-(0.0,1.0)*XIM
      RETURN
      END
```

References

1. M. J. Berger, Methods in Computational Physics, Vol. I, p. 135, Eds., B. Adler, S. Fernbach and M. Rotenberg, Academic Press, New York, 1963.

2. C. D. Zerby and F. L. Keller, Nucl. Sci. Eng. 27, 190 (1968).

3. S. Goudsmit and J. L. Saunderson, (a) Phys. Rev. 57, 24 (1940); (b) Ibid., p. 36.

4. L. Katz and A. S. Penfold, Rev. Mod. Phys. 24, 28 (1952).

5. N. F. Mott, (a) Proc. Roy. Soc. (London) A124, 426 (1929); (b) Ibid., A135, 429 (1932).

6. F. Rohrlich and B. C. Carlson, Phys. Rev. 93, 38 (1954).

7. F. R. Huson, BNL 11368 - L42, 1967.

8. D. O. Caldwell, Phys. Rev. 100, 291 (1955).

9. R. M. Sternheimer, Ibid., 103, 511 (1956).

10. A. T. Nelms, Nat. Bur. Std. (U.S.), Circular 577 (Supplement), 1958.

11. H. J. Bhabha, Proc. Roy. Soc. (London) A164, 257 (1938).

12. H. Feshbach, Phys. Rev. 88, 295 (1952).

13. Handbook of Mathematical Functions, Eds., M. Abramowitz and I. A. Stegun, p. 16, Nat. Bur. Std., Washington, D.C., 1964.

14. N. Sherman, Phys. Rev. 103, 1601 (1956).

15. D. R. Yennie, D. G. Ravenhall, and R. N. Wilson, Ibid., 95, 500 (1954).

16. L. V. Spencer, (a) Ibid., 98, 1507 (1955); (b) Nat. Bur. Std. (U.S.), Monograph 1 (1959).

17. G. Moliere, Z. Naturforsch 39, 78 (1948).

18. M. J. Berger, private communication.

19. Handbook of Mathematical Functions, Eds., M. Abramowitz and I. A. Stegun, pp. 916-9, Nat. Bur. Std., Washington, D.C., 1964.

Table 1. Path Length $s(E_0, E)$, Fractional Residual Path Length $t(E)$, Total Path Length $s(E_0, 0)$, and Range $r_t(E_0)$ for Al (Z=13) and Pb (Z=82), E_0 = 0.5 MeV

	Z = 13		Z = 82	
E (MeV)	s (cm)	t	s (cm)	t
0.50	0.000	1.000	0.000	1.000
0.48	4.539×10^{-3}	9.454×10^{-1}	1.672×10^{-3}	9.486×10^{-1}
0.46	9.040 "	8.912 "	3.337 "	8.974 "
0.44	1.350×10^{-2}	8.376 "	4.992 "	8.465 "
0.42	1.791 "	7.845 "	6.637 "	7.959 "
0.40	2.228 "	7.320 "	8.271 "	7.456 "
0.38	2.658 "	6.801 "	9.890 "	6.958 "
0.36	3.083 "	6.291 "	1.149×10^{-2}	6.465 "
0.34	3.501 "	5.788 "	1.308 "	5.977 "
0.32	3.911 "	5.294 "	1.464 "	5.496 "
0.30	4.313 "	4.810 "	1.619 "	5.022 "
0.28	4.706 "	4.337 "	1.770 "	4.556 "
0.26	5.089 "	3.877 "	1.918 "	4.100 "
0.24	5.461 "	3.429 "	2.063 "	3.654 "
0.22	5.820 "	2.997 "	2.204 "	3.220 "
0.20	6.165 "	2.582 "	2.341 "	2.800 "
0.18	6.495 "	2.185 "	2.472 "	2.396 "
0.16	6.808 "	1.809 "	2.598 "	2.010 "
0.14	7.100 "	1.456 "	2.717 "	1.644 "
0.12	7.371 "	1.131 "	2.828 "	1.302 "
0.10	7.617 "	8.348×10^{-2}	2.931 "	9.867×10^{-2}
0.08	7.834 "	5.733 "	3.023 "	7.032 "
0.06	8.019 "	3.511 "	3.103 "	4.566 "
0.04	8.166 "	1.748 "	3.169 "	2.539 "
0.02	8.267 "	5.292×10^{-3}	3.218 "	1.043 "
	$s_t(E_0)$ = 0.08311 cm		$s_t(E_0)$ = 0.03251 cm	
	$r_t(E_0)$ = 0.06065 cm		$r_t(E_0)$ = 0.01444 cm	

0.12 seconds computing time

Table 2. Ratio of Mott to Rutherford Cross Sections: Z = 13

θ (degrees)	10.0 MeV	1.0 MeV	0.1 MeV	0.01 MeV
0	1.000	1.000	1.000	1.000
5	1.0126	1.0118	1.0064	1.0014
10	1.0251	1.0235	1.0128	1.0027
15	1.0192	1.0191	1.0151	1.0053
30	0.9969	1.0009	1.0166	1.0109
60	0.8375	0.8611	0.9771	1.0141
90	0.5752	0.6278	0.8975	1.0072
120	0.2939	0.3769	0.8074	0.9963
150	0.0813	0.1868	0.7381	0.9873
165	0.0225	0.1343	0.7188	0.9848
170	0.0114	0.1243	0.7151	0.9843
175	0.0047	0.1183	0.7129	0.9840
179	0.0025	0.1164	0.7122	0.9839

1.6 seconds computing time

Table 3. Ratio of Mott to Rutherford Cross Sections: Z = 50

θ (degrees)	10.0 MeV	1.0 MeV	0.1 MeV	0.01 MeV
0	1.0000	1.0000	1.0000	1.0000
5	1.0334	1.0303	1.0111	1.0013
10	1.0668	1.0606	1.0221	1.0026
15	1.1214	1.1123	1.0408	1.0057
30	1.2345	1.2226	1.1127	1.0006
60	1.2736	1.2764	1.2280	1.0162
90	1.0126	1.0548	1.2261	1.0969
120	0.5693	0.6639	1.1421	1.2207
150	0.1651	0.3048	1.0527	1.3450
165	0.0451	0.1980	1.0255	1.3840
170	0.0220	0.1774	1.0203	1.3917
175	0.0080	0.1649	1.0171	1.3963
179	0.0035	0.1609	1.0161	1.3978

2.2 seconds computing time

Table 4. Ratio of Mott to Rutherford Cross Sections: Z = 80

θ (degrees)	10.0 MeV	1.0 MeV	0.1 MeV	0.01 MeV
0	1.0000	1.0000	1.0000	1.0000
5	1.0366	1.0326	1.0107	1.0013
10	1.0732	1.0651	1.0214	1.0026
15	1.1250	1.1066	1.0186	1.0050
30	1.3583	1.3181	1.0486	1.0098
60	1.8737	1.8199	1.3394	0.9825
90	1.8930	1.8927	1.6466	0.9529
120	1.2465	1.3660	1.8073	1.3127
150	0.3940	0.6424	1.8703	1.9177
165	0.1087	0.3984	1.8832	2.1335
170	0.0523	0.3501	1.8855	2.1766
175	0.0180	0.3206	1.8869	2.2030
179	0.0069	0.3111	1.8873	2.2115

2.5 seconds computing time

Table 5. Goudsmit-Saunderson Distribution: $Z = 13$, $E_o = 1.0$ MeV

	t = 0.1 $E_f = 0.1928$		t = 0.5 $E_f = 0.5786$		t = 0.9 $E_f = 0.9167$		t = 0.99 $E_f = 0.9917$	
ω (degrees)	A	LT	A	LT	A	LT	A	LT
0	5.207×10^{-1}	4	1.073×10^{0}	6	7.656×10^{0}	17	1.242×10^{2}	73
5	5.206 "	4	1.070 "	6	7.415 "	17	6.929×10^{1}	62
10	5.204 "	4	1.061 "	6	6.742 "	14	1.697 "	67
15	5.200 "	4	1.046 "	6	5.763 "	17	3.447×10^{0}	80
30	5.179 "	4	9.691×10^{-1}	5	2.584 "	16	1.281×10^{-1}	94
60	5.103 "	4	7.194 "	6	2.185×10^{-1}	18	6.284×10^{-3}	191*
90	5.000 "	6	4.522 "	8	2.448×10^{-2}	22	1.096 "	186*
120	4.896 "	4	2.565 "	6	5.476×10^{-3}	21	2.896×10^{-4}	168*
150	4.821 "	4	1.508 "	6	1.852 "	21	9.367×10^{-5}	166*
165	4.800 "	4	1.268 "	7	1.301 "	21	6.128 "	168*
180	4.793 "	4	1.190 "	7	1.140 "	22	5.151 "	169*
Computing time, seconds	0.04		0.05		0.06		0.11	

*Transformed series converged in LT-150 terms.

Table 6. Goudsmit-Saunderson Distribution: $Z = 13$, $E_o = 0.1$ MeV

	t = 0.1 $E_f = 0.0261$		t = 0.5 $E_f = 0.0664$		t = 0.9 $E_f = 0.0939$		t = 0.99 $E_f = 0.0994$	
ω (degrees)	A	LT	A	LT	A	LT	A	LT
0	5.114×10^{-1}	4	9.077×10^{-1}	6	6.706×10^{0}	19	**	0**
5	5.114 "	4	9.057 "	6	6.499 "	18	6.169×10^{1}	248*
10	5.113 "	4	8.996 "	6	5.920 "	17	1.380 "	217*
15	5.110 "	4	8.896 "	6	5.090 "	18	3.650×10^{0}	193*
30	5.099 "	4	8.378 "	5	2.432 "	17	2.210×10^{-1}	190*
60	5.058 "	4	6.662 "	6	3.142×10^{-1}	20	1.292×10^{-2}	186*
90	5.000 "	6	4.728 "	8	5.449×10^{-2}	24	2.823×10^{-3}	180*
120	4.943 "	4	3.200 "	6	1.751 "	24	1.106 "	162*
150	4.901 "	4	2.303 "	6	9.267×10^{-3}	23	6.474×10^{-4}	161*
165	4.890 "	4	2.088 "	6	7.928 "	25	5.668 "	160*
180	4.886 "	4	2.017 "	7	7.529 "	27	5.424 "	163*
Computing time, seconds	0.04		0.05		0.06		0.11	

*Transformed series converged in LT-150 terms.

**Series failed to converge.

Table 7. Goudsmit-Saunderson Distribution: Z = 82, E_o = 1.0 MeV

	t = 0.1 E_f = 0.1928		t = 0.5 E_f = 0.5786		t = 0.9 E_f = 0.9167		t = 0.99 E_f = 0.9917	
ω (degrees)	A	LT	A	LT	A	LT	A	LT
0	5.000×10^{-1}	4	5.001×10^{-1}	4	9.917×10^{-1}	7	1.529×10^{1}	29
5	"	4	"	4	9.890 "	7	1.413 "	25
10	"	4	"	4	9.809 "	7	1.125 "	27
15	"	4	"	4	9.675 "	7	7.891×10^{0}	28
30	"	4	"	4	8.995 "	6	1.781 "	29
60	"	4	"	4	6.833 "	6	1.326×10^{-1}	35
90	"	6	5.000×10^{-1}	6	4.580 "	8	3.078×10^{-2}	38
120	"	4	"	4	2.952 "	6	1.039 "	39
150	"	4	4.999×10^{-1}	4	2.063 "	7	3.945×10^{-3}	40
165	"	4	"	4	1.856 "	7	2.712 "	41
180	"	4	"	4	1.789 "	7	2.329 "	43
Computing time, seconds	0.04		0.04		0.05		0.09	

Table 8. Goudsmit-Saunderson Distribution: Z = 82, E_o = 0.1 MeV

	t = 0.1 E_f = 0.0190		t = 0.5 E_f = 0.0632		t = 0.9 E_f = 0.0933		t = 0.99 E_f = 0.0993	
ω (degrees)	A	LT	A	LT	A	LT	A	LT
0	5.000×10^{-1}	4	5.000×10^{-1}	4	8.778×10^{-1}	8	2.460×10^{1}	1**
5	"	4	"	4	8.755 "	8	2.066 "	245*
10	"	4	"	4	8.687 "	8	1.115 "	200*
15	"	4	"	4	8.574 "	8	6.269×10^{0}	194*
30	"	4	"	4	8.011 "	8	1.392 "	166*
60	"	4	"	4	6.308 "	8	1.733×10^{-1}	165*
90	"	6	"	6	4.652 "	10	5.285×10^{-2}	166*
120	"	4	"	4	3.503 "	8	2.597 "	165*
150	"	4	"	4	2.877 "	9	1.749 "	165*
165	"	4	"	4	2.731 "	8	1.589 "	165*
180	"	4	"	4	2.683 "	9	1.540 "	172*
Computing time, seconds	0.04		0.04		0.06		0.12	

*Transformed series converged in LT-150 terms.

**A obtained by interpolation from Figure 1.

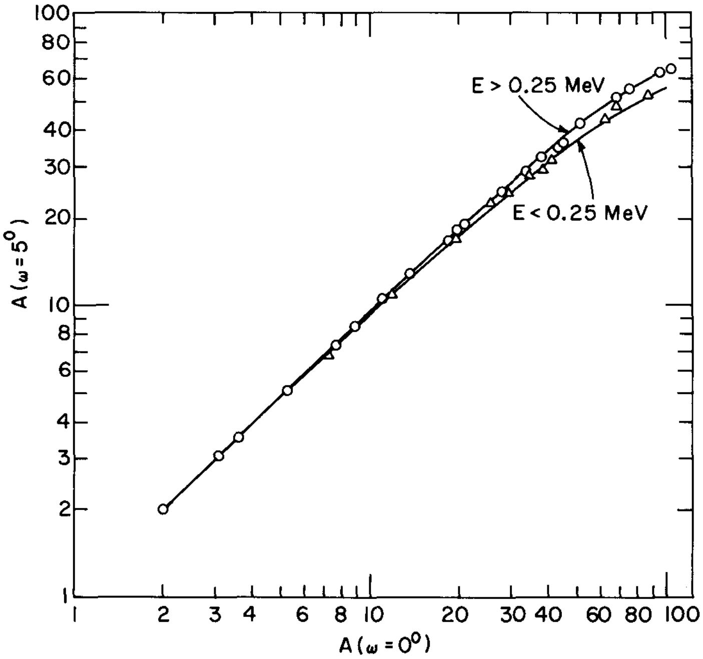

Figure 1

Goudsmit-Saunderson Distribution: A (ω=5°) vs. A (ω=0°) for Several Values of Z, E_0, and t