UNION CARBIDE CORPORATION

Union Carbide Corporation
Defense and Space Systems Department

UCC/DSSD – 270

CROSS SECTION CALCULATIONS AND THE STUDY OF SPACE VEHICLE RADIATION SHIELDING(U)

by Clayton D. Zerby and Henry Brysk

Final Report

May 15, 1967

Work Performed Under Contract No. NAS8–18007

National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Huntsville, Alabama

Space Sciences and Engineering Laboratory
5 New Street
White Plains, New York 10601

PREFACE

This report covers work done on the calculation of the Bremsstrahlung cross section (both theoretical analysis and computer programming) under Contracts NASw-906, NASw-1235, and NAS8-18007. The theory given here is that which was finally applied to the development of the computer program. Additional theoretical results, representing alternative approaches as well as extensions of the analysis, will be found in a previous report (UCC/DSSD-206). A derivation is also given of the angular distribution of the scattered electrons; this is not included in the computer program. The structure and operation of the program are described. A comparison is given of the available computer results with experiment.

The Appendix consists of the latest listing of the program, adjusted for the Marshall Flight Space Center operating system. The program has been extensively checked out, both analytically and numerically. At the time of writing of this report, however, the authors feel that there is still reason for further checks (on the basis of a comparison of test runs). Results obtained with the program at 50 kev are in fair agreement with experiment, but no higher energy runs are yet available.

TABLE OF CONTENTS

BREMSSTRAHLUNG THEORY

AND

DISCUSSION OF THE COMPUTER PROGRAM

I. INTRODUCTION

It is well known that the Bethe-Heitler expression for the electron Bremsstrahlung cross section[1], which was derived using the Born approximation, yields inaccurate results in comparison with experiment.[2] In general at energies below approximately 10 Mev the photon energy spectrum is underestimated over its complete range with the most serious discrepancy occuring at the higher photon energies. As a result, it is estimated that at 1 Mev incident electron energy the average energy release in a radiative collision with Aluminum is underestimated by over 30% while for Gold it is underestimated by as much as 55%.[2]

The reason for the discrepancy can be attributed almost entirely to the use of plane waves for the electron wavefunction in the matrix element as prescribed by the Born approximation. The theoretical estimate can be improved in accuracy by using Coulomb wavefunctions in the matrix element. In this way the influence of the nuclear electrostatic field can be taken into account in the most accurate way. Jaeger and Hulme[3,4,5] were the first to use the more accurate wavefunctions in the closely allied process of electron-positron pair production by photons. Their results were in good agreement with experiment.[3,6] Hence, very accurate results for the theoretical estimate of the Bremsstrahlung cross section can also be expected with the corresponding approach. The difficulty with the improved method, and the reason it has not been used extensively in the past, is that it does not yield a simple analytic formula, but requires extensive numerical procedures to obtain results.

In the present study, the Bremsstrahlung cross section problem is formulated with Dirac wavefunctions for the screened and unscreened nuclear electrostatic field. The screened potential is obtained from

self-consistent-field calculations. This requires that the wavefunctions be expanded in a series of angular momentum states. Hence, the formulation is suitable for calculation only at relatively low incident energies.

The cross section for unpolarized incident electrons is obtained for various polarization states of the photon, differential in the polar angle and energy of the photon. With a simple change in input for the incident particle's charge and/or mass, the program is equally applicable to the calculation of Bremsstrahlung from positrons and positive and negative muons.

The derivation starts in Chapter II with the presentation of the eigenfunctions of the energy and angular momentum operator of the Dirac equation. The eigenfunctions are used in the matrix element for the production of Bremsstrahlung as described in Chapter III. The final derivation of the cross section is given in Chapter IV. In Chapter V, Racah algebra is used to perform the magnetic quantum number sums explicitly, leading to some computational as well as formal simplifications.

For the computation of matrix elements, the procedure, as described in Chapter VI, is to terminate the numerical integration at some moderate radius, and to continue the integration out to infinity analytically by means of an asymptotic evaluation. This involves integrating by parts at the cut-off radius, to obtain an expansion in inverse powers of the radius in terms of the values of the wavefunctions and their derivatives there.

Chapter VII clears up an ambiguity in the phase of the Coulomb wavefunctions found in the literature (this was needed to establish the sign of each partial wave in the starting conditions for integration). In Chapter VIII, the analytic limit of the Coulomb wavefunctions as the kinetic energy goes to zero is derived. These are the scattered-electron wavefunctions in the end-point case (electron giving up its entire kinetic energy to the photon).

Initial values and equations for the radial integrals are covered in Chapter IX. Phase shift determination by wave-matching at the cut-off radius is described in Chapter X.

Chapter XI explains the procedure used for the computation of normalization factors by matching to the W.K.B. approximation solution.

Chapter XII discusses the self-consistent-field calculations which are utilized to supply the program with the screened atomic potential.

A derivation of the angular distribution of the scattered electrons is given in Chapter XIII. This distribution has not been incorporated into the computer program.

The structure of the computer program is outlined in Chapter XIV.

Operational instructions for running the program are given in Chapter XV.

Chapter XVI describes results obtained with the program and a comparison of them with experiment.

The Appendix consists of the latest listing of the computer program.

II. EIGENFUNCTIONS OF THE ENERGY AND ANGULAR MOMENTUM OPERATORS

The continuum eigenfunctions of the positive energy operator of the Dirac equation for an electron in a central electrostatic field have the form(7)

$$\phi_{\kappa,\mu} = \begin{Bmatrix} -ir^{-1} G_\kappa \Omega_{\kappa,\mu} \\ r^{-1} F_\kappa \Omega_{-\kappa,\mu} \end{Bmatrix} \tag{1}$$

where $\phi_{\kappa,\mu}$ is a bispinor, $\Omega_{\kappa,\mu}$ is a spinor and G_κ and F_κ are radial functions. The eigenfunctions $\phi_{\kappa,\mu}$ are also eigenfunctions of the operators, $\underset{\sim}{J}$, $\underset{\sim}{L}$, $\underset{\sim}{S}$, and J_z which are the total, orbital and spin angular momentum operators and the z component of the total angular momentum operator, respectively. The eigenvalues belonging to these operators are j, ℓ, $\frac{1}{2}$, and μ, respectively. The non-zero parameter κ, which can take on all positive and negative integral values, is related to j and ℓ by(7)

$$\left.\begin{aligned} \kappa &= -(j+\tfrac{1}{2}) = -(\ell+1), \text{ for } \kappa < 0, \\ \kappa &= j+\tfrac{1}{2} = \ell, \text{ for } \kappa > 0. \end{aligned}\right\} \tag{2}$$

The parameter κ is sufficient to designate both j and ℓ simultaneously since $j = |\kappa| - \frac{1}{2}$ and $\ell = j+\frac{1}{2}\kappa/|\kappa|$. To indicate the dependence of j and ℓ on κ, the eigenvalue j will be replaced with j_κ and ℓ will be replaced with ℓ_κ.

The eigenfunctions $\phi_{\kappa,\mu}$ are normalized on the energy scale so that*

---

*The notation ϕ^+ indicates the complex conjugate transpose of ϕ while ϕ^* indicates only the complex conjugate of ϕ.

$$\int \phi^{+}_{\kappa,\mu}(E)\,\phi_{\kappa',\mu'}(E')\,d^3x = \delta_{\kappa,\kappa'}\,\delta_{\mu,\mu'}\,\delta(E-E'). \tag{3}$$

The angular functions $\Omega_{\kappa,\mu}$ that appear in (1) are given by

$$\Omega_{\kappa,\mu} = \sum_n C(\ell_\kappa \tfrac{1}{2} j_\kappa ; \mu-n, n)\, Y_{\ell,\mu-n}(\hat{r})\, \eta_{\frac{1}{2},n} \tag{4}$$

where $\eta_{\frac{1}{2},n}$ is the spin-$\frac{1}{2}$ spinor with z-axis projection quantum number n, $Y_{\ell,\mu-n}(\hat{r})$ is a spherical harmonic, and $C(\ell_\kappa \tfrac{1}{2} j_\kappa ; \mu-n, n)$ is a Clebsch-Gordan coefficient.[8] These angular functions are orthonormal on the unit sphere so that

$$\int \Omega^{+}_{\kappa,\mu}\,\Omega_{\kappa',\mu'}\,d\Omega = \delta_{\kappa,\kappa'}\,\delta_{\mu,\mu'} \tag{5}$$

where $d\Omega$ is a differential element of solid angle.

The electron function $\phi_{\kappa,\mu}$ satisfies the equation

$$E\,\phi_{\kappa,\mu} = (-i\underset{\sim}{\alpha}\cdot\underset{\sim}{\nabla}+\beta+V)\,\phi_{\kappa,\mu}. \tag{6}$$

Consequently the radial functions satisfy

$$\left.\begin{aligned} F' - \frac{\kappa}{r}F + (E-V-1)\,G &= 0 \\ G' + \frac{\kappa}{r}G - (E-V+1)\,F &= 0 \end{aligned}\right\} \tag{7}$$

and have the asymptotic forms

$$\left.\begin{aligned} G &\sim \frac{\sqrt{E+1}}{\sqrt{\pi k}}\ \sin\left(kr - \frac{\ell_\kappa \pi}{2} + \delta_\kappa\right) \\ F &\sim \frac{\sqrt{E-1}}{\sqrt{\pi k}}\ \cos\left(kr - \frac{\ell_\kappa \pi}{2} + \delta_\kappa\right) \end{aligned}\right\} \tag{8}$$

where $k = (E^2-1)^{\frac{1}{2}}$ and δ_κ is the phase shift.

In the case of a pure Coulomb field with $V = -\frac{Ze^2}{r}$, the radial functions have the form

$$\left.\begin{aligned} G_\kappa &= \sqrt{E+1}\ \ (H_\kappa + H_\kappa^*) \\ F_\kappa &= i\sqrt{E-1}\ \ (H_\kappa - H_\kappa^*) \end{aligned}\right\} \tag{9}$$

where

$$H_\kappa = A_\kappa e^{-i\frac{1}{2}\pi(\gamma+\frac{1}{2})}\ (2kr)^{-\frac{1}{2}}\ M_{-\frac{1}{2}-i\nu,\gamma}\ (2ikr) \tag{10}$$

with

$$A_K = \frac{|\Gamma(\gamma+i\nu)|\ \ e^{\frac{1}{2}\pi\nu}\ e^{i\eta(\kappa)}\ (\gamma+i\nu)}{2(\pi k)^{\frac{1}{2}}\ \Gamma(2\gamma+1)} \tag{11}$$

In the above $M_{-\frac{1}{2}-i\nu,\gamma}\ (2ikr)$ is a Whittaker function,[9]

$\gamma = |(\kappa^2 - Z^2e^2)^{\frac{1}{2}}|$, $\nu = EZe^2/k$, and $\exp[2i\,\eta(\kappa)] = (-\kappa+i\nu E^{-1})(\gamma+i\nu)^{-1}$.

In the subsequent development it is not advantageous to use the complex conjugate form of H_κ shown in (10). Instead, the following relation for the Whittaker function can be used:[9]

$$M_{K,\mu}(z) = e^{i\epsilon\pi(\frac{1}{2}+\mu)}\ M_{-K,\mu}(-z)$$

where $\epsilon=1$ if $\mathrm{Im}(z)>0$ and $\epsilon=-1$ if $\mathrm{Im}(z)\leq 0$, to obtain the alternative and more useful expression:

$$H_\kappa^* = A_\kappa^*\ e^{-i\frac{1}{2}\pi(\gamma+\frac{1}{2})}\ (2kr)^{-\frac{1}{2}}\ M_{\frac{1}{2}-i\nu,\gamma}\ (2ikr). \tag{12}$$

In the case of the Coulomb field, the asymptotic forms of (8) are replaced with

$$\left.\begin{aligned} G &\sim \frac{\sqrt{E+1}}{\sqrt{\pi k}}\ \sin\left(kr+\nu\ \ell n\,(2kr) - \frac{\ell_\kappa \pi}{2} + \delta_\kappa\right) \\ F &\sim \frac{\sqrt{E-1}}{\sqrt{\pi k}}\ \cos\left(kr+\nu\ \ell n\,(2kr) - \frac{\ell_\kappa \pi}{2} + \delta_\kappa\right) \end{aligned}\right\} \tag{13}$$

where

$$\delta_\kappa = \eta(\kappa) - \arg\Gamma(\gamma+i\nu) - \frac{\pi}{2}\,(\gamma-\ell_\kappa-1) \tag{14}$$

III. THE MATRIX ELEMENT

The matrix element for the production of Bremsstrahlung is[10]

$$M = -e\int \underline{A}_q \cdot (\Psi_f^+(E')\ \underset{\sim}{\alpha}\ \Psi_m(E))\ d^3x \tag{1}$$

where $\underset{\sim}{\alpha}$ is the Dirac matrix operator. Since we will not be interested in angular or polarization details about the electron leaving the interaction, the final state electron function $\Psi_f(E')$ can be normalized on the energy scale according to the form given in Chapter II, Eq. 1:

$$\Psi_f(E') \equiv \phi_{\kappa',\mu'} = \begin{Bmatrix} -ir^{-1} G_{\kappa'}\, \Omega_{\kappa',\mu'} \\ r^{-1} F_{\kappa'}\, \Omega_{-\kappa',\mu'} \end{Bmatrix} . \tag{2}$$

The initial state electron function $\Psi_m(E)$ is normalized on the momentum scale and has the asymptotic form

$$\Psi_m(E) \sim u(m) e^{i\underset{\sim}{k}\cdot\underset{\sim}{r}} + Gr^{-1} e^{ikr} \tag{3}$$

where u(m) and G are bispinors. The z component of the spin associated with the plane wave in (3) is designated by m. It can be shown that if the axis of quantization is taken along the direction of propagation of the incident electron, then[7] $\Psi_m(E)$ is given by

$$\Psi_m(E) = \sum_{\kappa} 2\pi(2Ek)^{-\frac{1}{2}} C_{\kappa,m}\, \phi_{\kappa,m} \tag{4}$$

with

$$C_{\kappa,m} = i^{\ell_\kappa+1} (2\ell_\kappa+1)^{\frac{1}{2}} C(\ell_\kappa \tfrac{1}{2} j_\kappa; om)\ e^{i\delta_\kappa} \tag{5}$$

and $\phi_{\kappa,m}$ as defined in (1) of Chapter II.

The electromagnetic wave in (1) can be represented as a linear combination of waves which are circularly polarized perpendicular to the direction of propagation. If the wave was propagating along the z axis it would have the form

$$\underset{\sim}{A}_q(0) = \sum_{p=\pm 1} a_p \underset{\sim}{A}_p \tag{6}$$

with

$$\underset{\sim}{A}_p = -p\hat{\xi}_p \sqrt{\frac{2\pi}{\epsilon}}\, e^{-iqz} = -p\hat{\xi}_p \sqrt{\frac{2\pi}{\epsilon}} \sum_{\ell} (-i)^{\ell} [4\pi(2\ell+1)]^{\frac{1}{2}} Y_{\ell,0}(\hat{r}) j_{\ell}(qr) \tag{7}$$

where ϵ and q are the photon energy and momentum, respectively, $\hat{\xi}_p$ are the spherical basis[8], $j_{\ell}(qr)$ are the spherical Bessel functions, and a_p are constants (to be discussed below). The wave $\underset{\sim}{A}_{+1}$ is l.h. circularly polarized and $\underset{\sim}{A}_{-1}$ is r.h. circularly polarized. Rotation of the coordinate system so that the photon is propagated in an arbitrary direction with respect to the new system yields the required expression

$$\underset{\sim}{A}_p = -p \frac{4\pi}{\sqrt{2\epsilon}} \sum_{\ell} \sum_{\lambda} \sum_{m'=-\lambda}^{+\lambda} (-i)^{\ell}(2\ell+1)^{\frac{1}{2}} j_{\ell}(qr) C(\ell\ 1\lambda; 0p)\ D^{\lambda}_{m'p}(\varphi_q, \theta_q, 0) \underset{\sim}{T}_{\lambda \ell m'} \tag{8}$$

where φ_q and θ_q are the azimuthal and polar angle, respectively, of the propagation vector, $D^{\lambda}_{m'p}$ is a rotation matrix[8] and $\underset{\sim}{T}_{\lambda \ell m'}$ is given by[8]

$$\underset{\sim}{T}_{\lambda \ell m'} = \sum_{\nu} C(\ell\ 1\lambda;\ m'-\nu, \nu)\ Y_{\ell, m'-\nu}(\hat{r})\ \ \hat{\xi}_{\nu} \tag{9}$$

The constants a_p in (6) can be shown to have the forms:

$$\left.\begin{array}{ll} \text{for a l.h. circularly polarized photon,} & a_p = \delta_{p,+1} \\ \text{for a r.h. circularly polarized photon,} & a_p = \delta_{p,-1} \\ \text{for a linearly polarized photon,} & a_p = 2^{-\frac{1}{2}} e^{-ip\varphi} \end{array}\right\} \tag{10}$$

In the case of linear polarization the angles of interest are indicated in Fig.

In the subsequent development it is appropriate to consider the matrix element with $\underset{\sim}{A}_q$ replaced with $\underset{\sim}{A}_p$. The new matrix element M(p) shows p as an argument. Hence

$$M = \sum_{p=\pm 1} a_p \, M(p). \tag{11}$$

The matrix element M(p) is obtained with the use of (1) through (11)

$$M(pm\nu'\mu') = \frac{4\pi^2 e}{\sqrt{\epsilon E k}} \sum_{\ell} \sum_{\lambda} p \, C(\ell 1 \lambda; 0 p) \sum_{m'=-\lambda}^{+\lambda} D^{\lambda}_{m',p}(\varphi_q, \theta_q, 0) \, F(\lambda \ell \nu' \mu' m m') \tag{12}$$

where $$F(\lambda \ell \nu' \mu' m m') = (-i)^{\ell} (2\ell+1)^{\frac{1}{2}} \sum_{\nu} C(\ell 1 \lambda; m'-\nu, \nu) \sum_{\nu} C_{\nu, m}$$

$$\int j_{\ell}(qr) \, Y_{\ell, m'-\nu}(\hat{r}) \, \phi^{+}_{\nu', \mu'} \alpha_{\nu} \phi_{\nu, m} d^3x \tag{13}$$

and $$\alpha_{\nu} = \hat{\epsilon}_{\nu} \cdot \underset{\sim}{\alpha}.$$

The integral over the angles in (13) is readily found with the result that (13) becomes

$$F(\lambda \ell \nu' \mu' m m') = \delta_{\mu'-m, m'} \sqrt{\frac{3}{4\pi}} \quad A(\lambda \ell \nu' \mu' m) \tag{14}$$

and, as a result, (12) becomes

$$M(pm\nu'\mu') = \frac{2\pi e \sqrt{3\pi}}{\sqrt{qEk}} \sum_{\lambda} \sum_{\ell} pC(\ell 1 \lambda; 0 p) \, D^{\lambda}_{\mu'-m, p}(\varphi_q, \theta_q, 0) A(\lambda \ell \nu' \mu' m). \tag{15}$$

If we set $\mu' - m = \epsilon$ in (15) and eliminate μ' we have

$$A(\lambda \ell \nu' \mu' m) \equiv A(\lambda \ell \nu' \epsilon m) \tag{16}$$

where

$$A(\lambda \ell \nu' \epsilon m) = \sum_{\nu} \left[H_1(\nu' \nu m) \, B_{\ell \lambda}(\nu', -\nu \epsilon m) - H_2(\nu' \nu m) \, B_{\ell \lambda}(-\nu' \nu \epsilon m) \right], \tag{17}$$

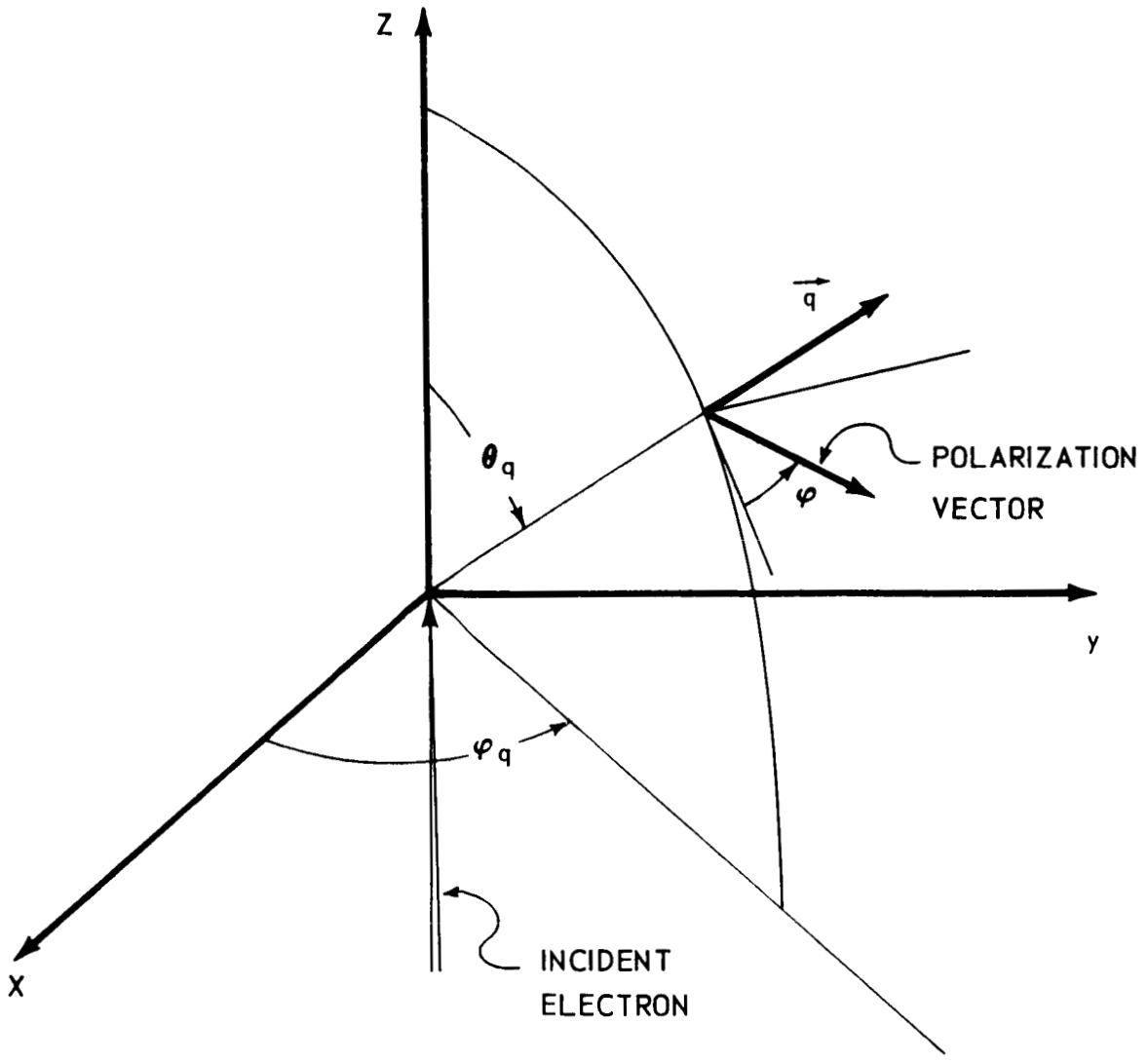

FIGURE 1 GEOMETRY OF BREMSSTRAHLUNG INTERACTION INDICATING THE CASE OF A LINEARLY POLARIZED PHOTON.

with

$$H_1(\nu'\nu m) = i^{\ell+\ell_\nu+1}\,(-1)^{m+1/2}\,(2\ell+1)(2j_\nu+1)^{1/2}\,e^{i\delta_\nu}\,C(j_\nu\ \tfrac{1}{2}\ \ell_\nu;\ -m,m)$$
$$C(\ell\ell_{\nu'}\ \ell_{-\nu};00)\ K_\ell(\nu'\nu) \tag{18}$$

and

$$H_2(\nu'\nu m) = i^{\ell+\ell_\nu+1}\,(-1)^{m+1/2}\,(2\ell+1)(2j_\nu+1)^{1/2}\,e^{i\delta_{\nu'}}\,C(j_\nu\ \tfrac{1}{2}\ \ell_\nu;-m,m)$$
$$C(\ell\ell_{-\nu},\ell_\nu;00)\ K_\ell\ (\nu\nu') \tag{19}$$

The $B_{\ell\lambda}$ functions in (17) can be determined from

$$B_{\ell\lambda}\ (\nu',-\nu\epsilon m) = \sum_\delta C(\ell_{\nu'}\tfrac{1}{2}j_{\nu'};\ \epsilon+m-\delta,\delta)\sum_\beta (-1)^{\delta-\beta}\,C(\tfrac{1}{2}1\tfrac{1}{2};\delta,-\delta+\beta)x$$
$$C(\ell 1\lambda;\epsilon-\delta+\beta,\delta-\beta)C(\ell_{-\nu}\tfrac{1}{2}j_{-\nu};m-\beta,\beta)C(\ell\ell_{-\nu}\ell_{\nu'};\epsilon-\delta+\beta,m-\beta). \tag{20}$$

The K_ℓ functions in (18) and (19) can be determined from

$$\left.\begin{aligned} K_\ell(\nu'\nu) &= \int_0^\infty j_\ell(qr)\ G_{\nu'}\ (E',r)\ F_\nu\ (E,r)dr \\ \text{and} \\ K_\ell(\nu\nu') &= \int_0^\infty j_\ell(qr)\ G_\nu\ (E,r)\ F_{\nu'}\ (E',r)dr \end{aligned}\right\} \tag{21}$$

IV. DERIVATION OF THE CROSS SECTION

The cross section for production of Bremsstrahlung by unpolarized incident electrons and without regard for the direction or polarization of the scattered electron or the azimuthal angle of the photon is given by[10]

$$d\sigma = \frac{2\pi}{v} \sum_{m\mu'\mu'} \int_{\varphi_q=0}^{2\pi} \left| M\,(m\mu'\mu') \right|^2 \frac{q^2 dq\, d\mu_q d\varphi_q}{(2\pi)^3} \qquad (1)$$

where the polarization states of the incident electron (designated by m) have been averaged over, v is the incident velocity of the electron and $\mu_q = \cos\theta_q$. Using the matrix elements defined by (11) and (13) of Chapter III,

$$d\sigma = \sum_{p=\pm 1} \left\{ \left| a_p \right|^2 d\sigma_{pp} + a_{-p}^{*} a_p \, d\sigma_{-p,p} \right\} \qquad (2)$$

where a_p is defined in (10) of Chapter III and

$$d\sigma_{p'p} = \frac{\pi}{v} \sum_{m\mu'\mu'} \int_{\varphi_q=0}^{2\pi} M^{+}(p'm\mu'\mu')\, M\,(pm\mu'\mu') \frac{q^2 dq\, d\mu_q d\varphi_q}{(2\pi)^3} \,. \qquad (3)$$

Hence, it follows that

$$\begin{aligned}
d\sigma &= d\sigma_{11} && \text{for l.h. circularly polarized photons} \\
d\sigma &= d\sigma_{-1-1} && \text{for r.h. circularly polarized photons} \qquad (4) \\
d\sigma &= \frac{1}{2} \sum_{p=\pm 1} \left(d\sigma_{pp} + e^{-2ip\varphi}\, d\sigma_{-p,p} \right) && \text{for linearly polarized photons}^{*} \\
d\sigma &= \sum_{p=\pm 1} d\sigma_{pp} && \text{for unpolarized photons.}
\end{aligned}$$

The integral over the photon azimuthal angle involves the integral

$$I = \int_0^{2\pi} D^{*\,\lambda_1}_{\mu'-m,p'}(\varphi_q \theta_q 0) D^{\lambda}_{\mu'-m,p}(\varphi_q \theta_q 0)\, d\varphi_q \qquad (5)$$

Actually the integrand of this integral is independent of φ and it is

---

* See Fig. 1.

easily shown that

$$I = 2\pi \sum_j (-1)^{\mu'-m-p'} C(\lambda_1 \lambda j; -\mu, m, \mu'-m) C(\lambda_1 \lambda j; -p', p)$$

$$\left[\frac{(j-|p'-p|)!}{(j+|p'-p|)!}\right]^{1/2} P_{j,|p'-p|}(\cos\theta_q) \qquad (6)$$

In (6), $P_{j,m}$ are the associated Legendre polynomials:

$$P_{j,m}(\theta_q) = \frac{1}{2^j j!} (1-\cos^2\theta_q)^{m/2} \left[\frac{d}{d(\cos\theta)}\right]^{j+m}_{\theta=\theta_q} (\cos^2\theta-1)^j \qquad (7)$$

where m is positive definite. The fact that

$$(-1)^{p'-p} = 1 \qquad (8)$$

in all cases has been used.

With the use of (15) of Chapter III and of (6), the cross section (3) becomes

$$d\sigma_{p'p} = \sum_j d\sigma^j_{p'p} \left[\frac{(j-|p'-p|)!}{(j+|p'-p|)!}\right]^{1/2} P_{j,|p'-p|}(\cos\theta_q) \qquad (9)$$

where, upon setting $v = k/E$ and using $k^2 = E^2-1$,

$$d\sigma^j_{p'p} = 3\pi^2 e^2 \frac{(E-1)}{(E+1)} B_j(p'p)\, y\,dy\,d(\cos\theta_q) \qquad (10)$$

with $y = q(E-1)^{-1}$. Also

$$B_j(p'p) = B^*_j(pp') = B_j(-p',-p) = \sum_{\ell\ell_1} A_j(p'p\ell_1\ell), \qquad (11)$$

where

$$A_j(p'p\ \ell_1\ell) = p\ p' \sum_{\lambda_1\lambda} C(\ell_1\ 1\ \lambda_1;0p')C(\ell 1\lambda;0p)C(\lambda_1\lambda j;-p',p)x$$

$$V_j(\lambda_1\lambda\ell_1\ell). \tag{12}$$

The function V_j in (12) is independent of p and p' and is given by

$$V_j(\lambda_1\lambda\ell_1\ell) = \sum_{\epsilon} (-)^{\epsilon+1}\ C(\lambda_1\lambda j;-\epsilon,\epsilon)\varphi(\lambda_1\lambda\ell_1\ell\epsilon) \tag{13}$$

where, in turn, the φ function is

$$\varphi(\lambda_1\lambda\ell_1\ell\epsilon) = \sum_{\kappa'}\sum_{m} A^*(\lambda_1\ell_1\kappa'\epsilon m)\ A(\lambda\ell\kappa'\epsilon m). \tag{14}$$

The A functions in (14) are given in (17) of Chapter III.

Since $B_j(p'p)$ is symmetric with respect to the pairs (ℓ,λ) and (ℓ_1,λ_1), both having been summed over in the same way, only the real part of φ contributes.

The computer program calculates

$$A_j^{(1)}(\ell_1\ell) = (-1)^j\ 6(2\ell_1+1)^{-1}\ (2\ell+1)^{-1}\ A_j(1\ 1\ \ell_1\ell), \tag{15a}$$

$$A_j^{(2)}(\ell_1\ell) = (-1)^j\ 6(2\ell_1+1)^{-1}\ (2\ell+1)^{-1}\ A_j(1,\ -1\ \ell_1\ell), \tag{15b}$$

and then

$$B_j^{(i)} = \sum_{\ell}\ (2\ell+1) \sum_{\ell_1\le\ell}\ (2-\delta_{\ell_1\ell})(2\ell_1+1)A_j^{(i)}(\ell_1\ell). \tag{16}$$

V. ANGULAR MOMENTUM SUMS

The expressions given for the Bremsstrahlung cross section contain several sums over angular momentum quantum numbers, the summands consisting of products of Clebsch-Gordan coefficients. These sums can be carried out formally, yielding Racah and X coefficients.

Going up the hierarchy of magnetic quantum number sums, there is first

$$B_{\ell\lambda}(\kappa',-\kappa\epsilon m) = \sum_{\delta} C(\ell_\kappa, \tfrac{1}{2}\, j_\kappa,; \epsilon+m-\delta,\delta) \sum_{\beta} (-1)^{\delta-\beta} C(\tfrac{1}{2}\, 1\, \tfrac{1}{2}; \delta, -\delta+\beta)$$

$$C(\ell 1\lambda; \epsilon-\delta+\beta, \delta-\beta) C(\ell_{-\kappa}\, \tfrac{1}{2}\, j_\kappa; m-\beta, \beta) C(\ell\ell_{-\kappa}\ell_\kappa,; \epsilon-\delta+\beta, m-\beta). \qquad (1)$$

After two recouplings (yielding two Racah coefficients) and an orthogonality relation, the β sum reduces to

$$\sum_{\beta} \text{---} = (-1)^{\ell-j_\kappa-\epsilon-m+\frac{1}{2}+\delta} \left[(2\lambda+1)(2\ell_\kappa,+1)(2j_\kappa+1)\right]^{1/2} \sum_{f} (2f+1)^{1/2}$$

$$W(\ell\ell_\kappa, \tfrac{1}{2}\, j_\kappa; \ell_{-\kappa} f) W(1\lambda\tfrac{1}{2}f; \ell\tfrac{1}{2}) C(\ell_\kappa, j_\kappa f; -\epsilon-m+\delta, m) C(\lambda f\tfrac{1}{2}; \epsilon, -\epsilon+\delta). \qquad (2)$$

Similarly, the remaining δ-sum yields

$$\sum_{\delta} (-1)^{\delta+\frac{1}{2}} C(\ell_\kappa, j_\kappa f; -\epsilon-m+\delta, m) C(\lambda f\tfrac{1}{2}; \epsilon, -\epsilon+\delta) C(\ell_\kappa, \tfrac{1}{2}\, j_\kappa,; \epsilon+m-\delta, \delta)$$

$$=(-1)^{\lambda+f+j_\nu,\ -\frac{1}{2}+\epsilon+m}\,[2(2f+1)]^{1/2}\,W(\ell_\nu, j_\nu \tfrac{1}{2}\lambda; f j_{\nu'})C(j_\nu \lambda j_{\nu'}; m\ \epsilon). \tag{3}$$

On splitting off the magnetic quantum number dependence by

$$B_{\ell\lambda}(\nu', -\nu\epsilon\ m) \equiv \bar{B}_{\ell\lambda}(\nu', -\nu)C(\lambda j_\nu j_{\nu'}; \epsilon\ m), \tag{4}$$

there remains

$$\bar{B}_{\ell\lambda}(\nu', -\nu) = (-1)^{\ell}\,[2(2\lambda+1)(2\ell_{\nu'}+1)(2j_\nu+1)]^{1/2} \sum_f (-1)^{f-\frac{1}{2}}(2f+1)$$

$$W(\ell\ell_{\nu'}, \tfrac{1}{2}j_\nu; \ell_{-\nu} f)W(1\lambda\tfrac{1}{2}f; \ell\tfrac{1}{2})W(\ell_{\nu'}, j_\nu\ \tfrac{1}{2}\ \lambda; f\ j_{\nu'}). \tag{5}$$

The f sum is an X-coefficient (after rearrangement), so that

$$\bar{B}_{\ell\lambda}(\nu', -\nu) = -[2(2\lambda+1)(2\ell_{\nu'}+1)(2j_\nu+1)]^{1/2}\, X(\lambda j_\nu j_{\nu'}; \ell\ell_{-\nu}\ell_{\nu'}; 1\ \tfrac{1}{2}\ \tfrac{1}{2}). \tag{6}$$

This X-coefficient can be expressed in a more elementary form.

Together with the matching parity Clebsch-Gordan coefficient, the result reduces to[11,12]

$$\bar{\bar{B}}_{\ell\lambda}(\nu', -\nu) = C(\ell\ell_{\nu'}, \ell_{-\nu}; 00)\bar{B}_{\ell\lambda}(\nu', -\nu)$$

$$= -3^{-1/2} C(\lambda j_{\nu'}, j_\nu; 0, \tfrac{1}{2})\ G_{\ell\lambda}(\nu', -\nu), \tag{7}$$

where the dependence on the signs of ν and ν' is contained in

$$G_{\ell\lambda}(\nu', \nu) = (\nu' - \nu)[\ell(\ell+1)]^{-1/2}\delta_{\lambda,\ell} + (\nu' + \nu + \ell)[\ell(2\ell+1)]^{-1/2}\delta_{\lambda,\ell-1}$$
$$+ (\nu' + \nu - \ell - 1)[(\ell+1)(2\ell+1)]^{-1/2}\delta_{\lambda,\ell+1} \tag{8}$$

Equation 8 does not hold for $\lambda = 0$, but this value is eliminated by the occurrence of a factor $C(\ell 1\lambda; 0, \pm 1)$ further upstream.

Separating the magnetic quantum number dependence of $A(\lambda \ell \nu' \epsilon m)$,

$$A(\lambda \ell \nu' \epsilon m) \equiv (-1)^{m+1/2} \sum_{\nu} \bar{H}(\lambda \ell \nu' \nu) C(\lambda j_\nu j_\nu, ; \epsilon m) C(j_\nu \tfrac{1}{2} \ell_\nu \; ; -m, m) \qquad (9)$$

where, in turn,

$$\bar{H}(\lambda \ell \nu' \nu) = i^{\ell_\nu + \ell + 1} (2\ell+1)(2j_\nu+1)^{1/2} e^{i\delta_\nu} [\bar{\bar{B}}_{\ell\lambda}(\nu', -\nu) K_\ell(\nu'\nu) - \bar{\bar{B}}_{\ell\lambda}(-\nu', \nu) K_\ell(\nu\nu')]. \qquad (10)$$

In the next round of summations now,

$$\varphi(\lambda_1 \lambda \ell_1 \ell \epsilon) = \sum_{\nu'} \sum_{m} A^*(\lambda_1 \ell_1 \nu' \epsilon m) A(\lambda \ell \nu' \epsilon m)$$

$$= \sum_{\nu' \nu \nu_1} \bar{H}^*(\lambda_1 \ell_1 \nu' \nu_1) \bar{H}(\lambda \ell \nu' \nu) \sum_{m} C(\lambda_1 j_{\nu_1} j_\nu, ; \epsilon m) C(j_{\nu_1} \tfrac{1}{2} \ell_{\nu_1} ; -m, m)$$

$$C(\lambda j_\nu j_\nu, ; \epsilon m) C(j_\nu \tfrac{1}{2} \ell_\nu ; -m, m). \qquad (11)$$

Recoupling to separate the ϵ and m dependence,

$$C(\lambda_1 j_{\nu_1} j_\nu, ; \epsilon m) C(\lambda j_\nu j_\nu, ; \epsilon m)$$

$$= (-1)^{\lambda - j_\nu, + m} (2j_\nu, +1)(2\lambda+1)^{-1/2} \sum_{s} (2s+1)^{1/2} W(\lambda_1 j_{\nu_1} \lambda j_\nu ; j_\nu, s)$$

$$C(j_{\nu_1} j_\nu s ; m, -m) C(\lambda_1 s \lambda ; \epsilon 0). \qquad (12)$$

The next step is

$$V_j(\lambda_1 \lambda \ell_1 \ell) = \sum_{\epsilon} (-)^{\epsilon+1} C(\lambda_1 \lambda j ; -\epsilon, \epsilon) \varphi(\lambda_1 \lambda \ell_1 \ell \epsilon).$$

The ϵ sum reduces to

$$\sum_{\epsilon} (-)^{\epsilon+1} C(\lambda_1 \lambda j ; -\epsilon, \epsilon) C(\lambda_1 s \lambda ; \epsilon 0) = (-1)^{s-\lambda+1} [(2\lambda+1)/(2s+1)]^{1/2} \delta_{sj} \qquad (13)$$

leaving

$$V_j(\lambda_1\lambda\ell_1\ell)=(-1)^{j+1}\sum_{\nu'}(2j_{\nu'}+1)\sum_{\nu\nu_1}H^*(\lambda_1\ell_1\nu'\nu_1)\bar{H}(\lambda\ell\nu'\nu)W(\lambda_1 j_{\nu_1}\lambda j_\nu;j_{\nu'}j)$$

$$\sum_m(-1)^{-j_{\nu'}+m}C(j_{\nu_1}j_\nu j;m,-m)C(j_{\nu_1}\tfrac{1}{2}\ell_{\nu_1};-m,m)C(j_\nu\tfrac{1}{2}\ell_\nu;-m,m). \qquad (14)$$

The m sum consists of the two terms $m=\pm\frac{1}{2}$ which differ only in the sign of the magnetic quantum numbers of the Clebsch-Gordan coefficients, hence by a phase factor of (-1) to the power

$$\begin{aligned}&1+(j_{\nu_1}+j_\nu-j)-(j_{\nu_1}+\tfrac{1}{2}-\ell_{\nu_1})-(j_\nu+\tfrac{1}{2}-\ell_\nu)=\ell_\nu+\ell_{\nu_1}-j\\&=(\ell_\nu+\ell_{-\nu'}+\ell)+(\ell_{\nu_1}+\ell_{-\nu'}+\ell_1)-2(\ell_{-\nu'}+\ell+\ell_1)+(\ell+\ell_1-j).\end{aligned} \qquad (15)$$

This yields the selection rule that $V_j(\lambda_1\lambda\ell_1\ell)$ vanishes unless

$$\ell+\ell_1-j = \text{even integer}. \qquad (16)$$

Subject to this condition, the m sum is twice the $m=-\frac{1}{2}$ term. On substituting the explicit value

$$C(j_\nu\tfrac{1}{2}\ell_\nu;\tfrac{1}{2},-\tfrac{1}{2}) = 2^{-1/2}, \qquad (17)$$

$$\begin{aligned}V_j(\lambda_1\lambda\ell_1\ell)=(-1)^j\sum_{\nu'\nu\nu_1}(-1)^{j_{\nu'}-1/2}(2j_{\nu'}+1)\\ \bar{H}^*(\lambda_1\ell_1\nu'\nu_1)\bar{H}(\lambda\ell\nu'\nu)W(\lambda j_\nu\lambda_1 j_{\nu_1};j_{\nu'}j)C(j_\nu j_{\nu_1}j;-\tfrac{1}{2},\tfrac{1}{2}).\end{aligned} \qquad (18)$$

Equation (18) requires the computation of general Racah coefficients, whose arguments are then summed over. It turns out to be more efficient to revertto the magnetic quantum number sums for ϵ and m. Returning to (11) and writing out explicitly the terms for $m = \pm 1/2$ with the help of (17),

$$\varphi(\lambda_1\lambda\ell_1\ell\epsilon)=\frac{1}{2}\sum_{\nu'\nu\nu_1}\bar{H}^*(\lambda_1\ell_1\nu'\nu_1)\bar{H}(\lambda\ell\nu'\nu)[C(\lambda_1 j_{\nu_1} j_{\nu'};\epsilon,-\tfrac{1}{2})C(\lambda j_\nu j_{\nu'};\epsilon,-\tfrac{1}{2}) \tag{19}$$

$$+(-1)^{\ell_\nu-\ell_{\nu_1}-j_\nu+j_{\nu_1}}C(\lambda_1 j_{\nu_1} j_{\nu'};\epsilon\ \tfrac{1}{2})C(\lambda j_\nu j_{\nu'};\epsilon\ \tfrac{1}{2})].$$

On splitting the ϵ sum into positive and negative values,

$$V_j(\lambda_1\lambda\ell_1\ell)=\frac{1}{2}\sum_{\epsilon\geq 0}(-1)^{\epsilon+1}(2-\delta_{\epsilon 0})\ [C(\lambda_1\lambda j;-\epsilon\epsilon)\varphi(\lambda_1\lambda\ell_1\ell\epsilon) \tag{20}$$

$$+C(\lambda_1\lambda j;\epsilon,-\epsilon)\varphi(\lambda_1\lambda\ell_1\ell,-\epsilon)].$$

Reversing the sign of the magnetic quantum numbers in the C-coefficients of (19) introduces for each a factor of (-1) to the power

$$(\lambda_1+j_{\nu_1}-j_{\nu'})-(\lambda+j_\nu-j_{\nu'})=\lambda_1-\lambda+j_{\nu_1}-j_\nu. \tag{21}$$

Interchanging the order of the two terms in (19), it is then found that

$$\varphi(\lambda_1\lambda\ell_1\ell,-\epsilon)=(-1)^{\ell_\nu+\ell_{\nu_1}+\lambda+\lambda_1}\varphi(\lambda_1\lambda\ell_1\ell\epsilon). \tag{22}$$

A charge of sign of the magnetic quantum numbers in the C-coefficient multiplying φ in (20) results in a factor of (-1) to the power $\lambda+\lambda_1-j$. Thus, in view of (15), the two terms in (20) are equal and

$$V_j(\lambda_1\lambda\ell_1\ell)=\sum_{\epsilon\geq 0}(-1)^{\epsilon+1}(2-\delta_{\epsilon 0})C(\lambda_1\lambda j;-\epsilon\epsilon)\varphi(\lambda_1\lambda\ell_1\ell\epsilon). \tag{23}$$

On defining

$$H(\lambda\ell\nu'\nu)=C(\lambda j_\nu, j_{\nu'};0\tfrac{1}{2})\ [G_{\ell\lambda}(\nu',-\nu)K_\ell(\nu'\nu)-G_{\ell\lambda}(-\nu',\nu)K_\ell(\nu\nu')], \tag{24}$$

there results

$$\mathrm{Re}[\varphi(\lambda_1\lambda\ell_1\ell\epsilon)]=(-1)^{\lambda+\lambda_1}(1/3)(2\ell+1)(2\ell_1+1)\bar{\varphi}(\lambda_1\lambda\ell_1\ell\epsilon) \tag{25}$$

where

$$\bar{\varphi}(\lambda_1 \lambda \ell_1 \ell \epsilon) = \sum_{\nu' \nu \nu_1} (2j_{\nu'}+1) H(\lambda_1 \ell_1 \nu' \nu_1) H(\lambda \ell \nu' \nu) i^{\ell_\nu - \ell_{\nu_1} + \ell - \ell_1} \cos(\delta_\nu - \delta_{\nu_1})$$

$$\frac{1}{2}\left[C(\lambda_1 j_\nu, j_{\nu_1}; \epsilon, -\epsilon+\frac{1}{2}) C(\lambda j_\nu, j_\nu; \epsilon, -\epsilon+\frac{1}{2})\right. \tag{26}$$

$$\left. + (-1)^{\ell_\nu - j_\nu - \ell_{\nu_1} + j_{\nu_1}} C(\lambda_1 j_\nu, j_{\nu_1}; \epsilon, -\epsilon - \frac{1}{2}) C(\lambda j_\nu, j_\nu; \epsilon, -\epsilon - \frac{1}{2})\right],$$

making use of the fact that

$$\ell_\nu - \ell_{\nu_1} + \ell - \ell_1 = (\ell_\nu + \ell + \ell_{-\nu}) - (\ell_{\nu_1} + \ell_1 + \ell_{-\nu'}) = \text{even integer} \tag{27}$$

whence i to this power is real.

VI. ASYMPTOTIC EVALUATION OF MATRIX ELEMENT TAILS

The radial integrals for the matrix elements (Chapter III, Eq. 21) are infinite integrals. In practice, they obviously have to be truncated. Asymptotically, F and G are sinusoidal while the spherical Bessel function goes as 1/r times a sinusodial (all of different arguments). If the value of the truncated integral is plotted against the cutoff radius, it rises smoothly till around a third of a Bohr radius, then oscillates with decreasing amplitude about the desired terminal value. Unfortunately, the damping is too slow for the integration to be continued till it settles (at least five Bohr radii, maybe ten). Instead, it is necessary to truncate the integral earlier and estimate the contribution of the tail. This is done most conveniently by performing integration-by-parts at the cutoff radius, thus obtaining an asymptotic expansion in inverse powers of r. Using[13]

$$\left.\begin{aligned} \int dr\, r^{\ell+2}\, j_\ell(qr) &= (1/q) r^{\ell+2}\, j_{\ell+1}(qr) \\ \int dr\, r^{-\ell}\, j_{\ell+1}(qr) &= -(1/q)\, r^{-\ell}\, j_\ell(qr) \end{aligned}\right\} \qquad (1)$$

and the differential equations for the wavefunctions (Chapter II, Eq. 7), there results

$$\begin{aligned} q\int_R^\infty G_\nu'(k'r) F_\nu(kr) j_\ell(qr)\,dr &= -G_\nu'(k'R) F_\nu(kR) j_{\ell+1}(qR) \\ &-(E'+1)\int_R^\infty F_\nu' F_\nu j_{\ell+1}\,dr + (E-1)\int_R^\infty G_\nu' G_\nu j_{\ell+1}\,dr \qquad (2) \\ &-(\nu-\nu'-\ell-2)\int_R^\infty r^{-1} G_\nu' F_\nu j_{\ell+1}\,dr + \int_R^\infty V(F_\nu' F_\nu - G_\nu' G_\nu) j_{\ell+1}\,dr, \end{aligned}$$

$$q\int_R^\infty F_\nu, G_\nu j_\ell dr = -F_\nu, G_\nu j_{\ell+1} - \int_R^\infty [(E+1)F_\nu, F_\nu + (E'-1)G_\nu, G_\nu] j_{\ell+1} dr$$

$$-(\nu'-\nu-\ell-2)\int_R^\infty r^{-1} F_\nu, G_\nu j_{\ell+1} dr + \int_R^\infty V(F_\nu, F_\nu - G_\nu, G_\nu) j_{\ell+1} dr, \tag{3}$$

$$q\int_R^\infty F_\nu, F_\nu j_{\ell+1} dr = F_\nu, F_\nu j_\ell - \int_R^\infty [(E'-1)G_\nu, F_\nu + (E-1)F_\nu, G_\nu] j_\ell dr$$

$$+(\nu'+\nu+\ell)\int_R^\infty r^{-1} F_\nu, F_\nu j_\ell dr + \int_R^\infty V(G_\nu, F_\nu + F_\nu, G_\nu) j_\ell dr, \tag{4}$$

$$q\int_R^\infty G_\nu, G_\nu j_{\ell+1} dr = G_\nu, G_\nu j_\ell + \int_R^\infty [(E+1)G_\nu, F_\nu + (E'+1)F_\nu, G_\nu] j_\ell dr$$

$$-(\nu'+\nu-\ell)\int_R^\infty r^{-1} G_\nu, G_\nu j_\ell dr - \int_R^\infty V(G_\nu, F_\nu + F_\nu, G_\nu) j_\ell dr. \tag{5}$$

The integration-by-parts can be continued to higher order.

If only the terms in lowest order in 1/r are retained, Eqs. 2-5 reduce to

$$2q\int_R^\infty G_\nu, F_\nu j_\ell dr = (EE'-1)G_\nu, F_\nu j_{\ell+1} - (E'+1)(E-1)F_\nu, G_\nu j_{\ell+1}$$

$$- E(E'+1)F_\nu, F_\nu j_\ell - E'(E-1)G_\nu, G_\nu j_\ell, \tag{6}$$

$$2q\int_R^\infty F_\nu, G_\nu j_\ell dr = -(E'-1)(E+1)G_\nu, F_\nu j_{\ell+1} + (EE'-1)F_\nu, G_\nu j_{\ell+1}$$

$$+ E'(E+1)F_\nu, F_\nu j_\ell + E(E'-1)G_\nu, G_\nu j_\ell. \tag{7}$$

The convergence of the procedure of truncating the integral and adding the asymptotic evaluation of the tail is tested by comparing results obtained with different cutoff radii. The accuracy is tested by constructing artificial problems for which the answer can be obtained in closed form.

It turns out that a cutoff radius of about one and one-half Bohr radii (or about 200 in limits of $\hbar/mc$) is satisfactory, and that adding the next order of terms beyond Eqs. 6 and 7 has little effect, at least in the energy range of prime interest.

At lower photon energies (say 25 kev), the form of the matrix element tail correction is more critical, as it is essentially an asymptotic expansion in inverse powers of qR, and inclusion of the next order terms is desirable. Carrying the integration-by-parts to the next order and solving the resulting simultaneous equations (and using the differential equations to simplify the expressions), Eqs. 6 and 7 are replaced by

$$
\begin{aligned}
2q\int_R^\infty G_\kappa, F_\kappa j_\ell dr &= j_\ell\,[E'G_\kappa, F_\kappa' - EF_\kappa G_\kappa',] + j_{\ell+1}[(E'+1)F_\kappa, F_\kappa' \\
&\quad - (E-1)G_\kappa G_\kappa', + (E-1)(E'+1)F_\kappa, G_\kappa + (EE'-1)G_\kappa, F_\kappa] \\
&+ (qR)^{-1} j_\ell\{[q^2+2k^2k'^2-\ell(EE'-1)]G_\kappa, F_\kappa + (E-1)(E'+1)[\ell-2(EE'-1)]G_\kappa F_\kappa,\} \\
&+ (qR)^{-1} j_{\ell+1}[(E'+1)(2E^2E'-E-E')F_\kappa, F_\kappa + (E-1)(2EE'^2-E-E')G_\kappa, G_\kappa] \\
&+ Vj_\ell[(E'+1)F_\kappa, F_\kappa + (E-1)G_\kappa, G_\kappa] - Vj_{\ell+1}(E+E')G_\kappa, F_\kappa
\end{aligned}
\tag{8}
$$

and

$$
\begin{aligned}
2q\int_R^\infty F_\kappa, G_\kappa j_\ell dr &= j_\ell[E'F_\kappa, G_\kappa' - EG_\kappa F_\kappa',] + j_{\ell+1}[(E+1)F_\kappa F_\kappa', \\
&\quad - (E'-1)G_\kappa, G_\kappa' + (EE'-1)F_\kappa, G_\kappa + (E+1)(E'-1)G_\kappa, F_\kappa] \\
&+ (qR)^{-1} j_\ell\{(E+1)(E'-1)[\ell-2(EE'-1)]G_\kappa, F_\kappa + [q^2+2k^2k'^2-\ell(EE'-1)]G_\kappa F_\kappa,\}
\end{aligned}
$$

$$- (qR)^{-1} j_{\ell+1}[(E+1)(2EE'^{2}-E-E')F_n, F_n + (E'-1)(2E^{2}E'-E-E')G_n, G_n] \qquad (9)$$

$$- Vj_{\ell}[(E+1)F_n, F_n + (E'-1)G_n, G_n] - Vj_{\ell+1}(E+E')F_n, G_n .$$

VII. PHASE OF COULOMB WAVEFUNCTIONS

The continuum solution of the Dirac equation for a Coulomb field contains an ambiguity of π in the phase shift. In computing matrix elements for electromagnetic interactions, it is necessary to adhere to a consistent convention for the phase. Since there is an established sign convention for the free particle solution, it provides a suitable calibration.

The relative phase of the two radial functions is well-defined, so it is sufficient to examine only the "large" component. The Coulomb function, normalized on the energy scale, is

$$G_\kappa = \frac{|\Gamma(\gamma+i\nu)|e^{\nu\pi/2}(E+1)^{1/2}(2kr)^\gamma}{2\Gamma(2\gamma+1)(\pi k)^{1/2}} \left[e^{-ikr+i\eta}(\gamma+i\nu)\,{}_1F_1(\gamma+1+i\nu,\right.$$
$$\left.2\gamma+1;2ikr) + c.c.\right] \qquad (1)$$

with $\exp(2i\eta) = -(\kappa - i\nu/E)/(\gamma+i\nu)$ (2)

and $\nu = (E/k)\,\alpha Z$, $\gamma = (\kappa^2-\alpha^2Z^2)^{1/2}$. (3)

The corresponding free particle function is

$$G_\kappa^o = (E+1)^{1/2}(\pi k)^{-1/2} kr\; j_{\ell(\kappa)}(kr). \qquad (4)$$

It is desired to fix the choice of the square root of Eq. 2 to be used for $\exp(i\eta)$ by requiring that Eq. 1 reduce to Eq. 4 in the limit as Z goes to zero (causing the potential to vanish).

As $Z \longrightarrow 0$, $\nu \longrightarrow 0$, and $\gamma \longrightarrow |\kappa|$. With these, Eq. 2 reduces to $-\kappa/|\kappa|$. By Kummer's transformation,[9]

$${}_1F_1(|\kappa|+1,2|\kappa|+1;-2ikr) = \exp(-2ikr)\;{}_1F_1(|\kappa|,2|\kappa|+1;2ikr). \qquad (5)$$

Factoring $\exp(-i\eta)$ out of Eq. 1 and letting $\exp(2i\eta) = -\kappa/|\kappa|$:

$$G_\kappa \longrightarrow \frac{\Gamma(|\kappa|)}{4\Gamma(2|\kappa|)} \left(\frac{E+1}{\pi k}\right)^{1/2} (2kr)^{|\kappa|} e^{-i\eta - ikr} \left[-\frac{\kappa}{|\kappa|} {}_1F_1\,(|\kappa|+1, 2|\kappa|+1; 2ikr)\right.$$

$$\left. + {}_1F_1\,(|\kappa|, 2|\kappa|+1; 2ikr)\right]. \qquad (6)$$

For $\kappa < 0$, the expression in brackets is[9]

$${}_1F_1\,(|\kappa|+1, 2|\kappa|+1; 2ikr) + {}_1F_1\,(|\kappa|, 2|\kappa|+1; 2ikr) = 2\,{}_1F_1\,(|\kappa|, 2|\kappa|; 2ikr) \qquad (7)$$

while $\ell(\kappa) = |\kappa| - 1$ and

$$kr\, j_{|\kappa|-1}(kr) = \frac{\pi^{1/2}}{\Gamma(|\kappa|+1/2)} \left(\frac{kr}{2}\right)^{|\kappa|} e^{-ikr}\, {}_1F_1(|\kappa|, 2|\kappa|; 2ikr). \qquad (8)$$

Using the doubling formula[9]

$$\Gamma(2|\kappa|) = 2^{2|\kappa|-1}\, \pi^{-1/2}\, \Gamma(|\kappa|)\Gamma(|\kappa| + 1/2), \qquad (9)$$

the limit reduces to

$$G_{-|\kappa|} \longrightarrow \exp(-i\eta)\, (E+1)^{1/2}\, (\pi k)^{-1/2}\, kr\, j_{|\kappa|-1}(kr). \qquad (10)$$

Comparing with Eq. 4, the correct limit is $\exp(i\eta) \longrightarrow 1$.

For $\kappa > 0$, the expression in brackets is[9]

$${}_1F_1(\kappa+1, 2\kappa+1; 2ikr) - {}_1F_1(\kappa, 2\kappa+1; 2ikr) = 2ikr\, {}_1F_1(\kappa+1, 2\kappa+2; 2ikr)/(2\gamma+1) \qquad (11)$$

while $\ell(\kappa) = \kappa$. The limit then reduces to

$$G_{|\kappa|} \longrightarrow -i\, \exp(-i\eta)(E+1)^{1/2}(\pi k)^{-1/2}\, kr\, j_{|\kappa|}(kr). \qquad (12)$$

The equivalence now yields $\exp(i\eta) \longrightarrow -i$.

Splitting $\exp(2i\eta)$ into real and imaginary parts,

$$\cos 2\eta = (-\kappa\gamma + \nu^2/E)/(\gamma^2+\nu^2) = 2\cos^2\eta - 1, \qquad (13)$$

$$\sin 2\eta = \nu(\kappa + \gamma/E)/(\gamma^2+\nu^2) = 2 \sin \eta \cos \eta. \qquad (14)$$

Thus, with $\epsilon = +1$ or -1,

$$\cos \eta = \epsilon[(1+\cos 2\eta)/2]^{1/2}, \qquad (15)$$

$$\sin \eta = \sin 2\eta/(2 \cos \eta). \qquad (16)$$

If Eqs. 15 and 16 are used to compute $\exp(i\eta)$, the phase convention is given by specifying ϵ. This choice is determined by the $Z \longrightarrow 0$ limits obtained above.

For $\varkappa < 0$, the limit is $\cos\eta \longrightarrow 1$, so $\epsilon = 1$. For $\varkappa > 0$, the limit yields $\cos\eta \longrightarrow 0$ (which leaves the sign undetermined) and $\sin\eta \longrightarrow -1$. Since $|\varkappa| > |\varkappa|\ /E > \gamma/E$, the sign of $\sin 2\eta$ is unaffected by E; for electrons, it is the same as that of $\varkappa$. For $\varkappa > 0$, then, $\sin 2\eta > 0$ while $\sin\eta < 0$, so $\cos\eta < 0$, and $\epsilon = -1$. Positron wave functions are obtained by taking Z negative, hence ν negative. The sign of $\sin 2\eta$ then reverses, and $\epsilon = 1$ for $\varkappa > 0$. Combining the results, the prescription for using Eqs. 15 and 16 is:

electrons: $\epsilon = -\varkappa/|\varkappa|$,

positrons: $\epsilon = 1$.

The relation between positron and electron phases is

$$(e^{i\eta})_{pos} = -\frac{\varkappa}{|\varkappa|}\ (e^{-i\eta})_{el}. \qquad (17)$$

In the limit of small kinetic energy ($k \longrightarrow 0$), ν becomes infinite and $\exp(2i\eta) \longrightarrow 1$. For an electron, then, $\exp(i\eta) = -\varkappa/|\varkappa|$. The limit is meaningless for a positron, as its wavefunction then tends to a delta function at infinity.

VIII. ZERO KINETIC ENERGY LIMIT OF COULOMB WAVEFUNCTION

The "small" component Coulomb wavefunction is

$$F_\kappa = \frac{i\,|\Gamma(\gamma+i\nu)|\,e^{\nu\pi/2}(E-1)^{1/2}(2kr)^\gamma}{2\Gamma(2\gamma+1)(\pi k)^{1/2}}\,[e^{-ikr+i\eta}(\gamma+i\nu)_1F_1(\gamma+1+i\nu;2\gamma+1;2ikr)-c.c.]. \quad (1)$$

In the limit of vanishing kinetic energy ($k \longrightarrow 0$), $\nu = E\alpha Z/k$ goes to infinity (through positive values for an electron, negative for a positron). Going to the limit on the various factors:[13]

$$\lim_{|\nu|\longrightarrow\infty} (2\pi)^{-1/2}\,|\Gamma(\gamma+i\nu)|\,e^{|\nu|\pi/2}|\nu|^{1/2-\gamma} = 1. \quad (2)$$

$$\lim_{a\longrightarrow\infty} {}_1F_1(a,b;-z/a)/\Gamma(b) = z^{(1/2)(1-b)}J_{b-1}\,(2\,z^{1/2}) \quad (3)$$

yields, with

$$a = \gamma+1+i\nu, \qquad b = 2\gamma+1,$$

$$z = -2ikr\,(\gamma+1+i\nu) \approx 2\,\alpha\,Zr,$$

$$\lim_{\nu\longrightarrow\infty} {}_1F_1(\gamma+1+i\nu,2\gamma+1;2ikr) = \Gamma(2\gamma+1)(2\,\alpha\,Zr)^{-\gamma}\,J_{2\gamma}\,[2(2\,\alpha\,Zr)^{1/2}]. \quad (4)$$

Inasmuch as the source reference did not establish the validity of this result when a increases through imaginary rather than real values, it was verified by writing out the confluent hypergeometric series and going to the limit term by term.

$$[E-1]^{1/2} = [(k^2+1)^{1/2}-1]^{1/2} \approx [(1+k^2/2)\,-1]^{1/2} = 2^{-1/2}k \quad (5)$$

$$(\gamma+i\nu)\,-\,c.c.\,=\,2i\nu \approx 2i(\alpha\,Z/k) \quad (6)$$

$$\exp\,(-ikr+i\eta) \approx -\kappa/|\kappa| \text{ for an electron.} \quad (7)$$

Combining these results, the electron wavefunction ($\nu > 0$) is

$$F_\kappa = (\kappa/|\kappa|)\,(\alpha\,Z)^{1/2}\,J_{2\gamma}\,[2(2\,\alpha\,Zr)^{1/2}]. \quad (8)$$

For G_κ, the analogous procedure becomes more complicated because there occurs

$$(\gamma+i\nu) + \text{c.c.} = 2\gamma, \tag{9}$$

so that a term involving ν times the second term in the limiting expression for the confluent hypergeometric function must be included. It is simpler to resort to the differential equation:

$$F_\kappa' - (\kappa/r)F_\kappa = -(E-V-1)G_\kappa \approx -(\alpha Z/r)G_\kappa . \tag{10}$$

$$F_\kappa' = (\kappa/|\kappa|)(\alpha Z)^{1/2}\ (d/dr)\ J_{2\gamma}\ [2(2\alpha Zr)^{1/2}] \tag{11}$$

$$= (\kappa/|\kappa|)(\alpha Z)^{1/2}(2\alpha Z/r)^{1/2}\left\{ J_{2\gamma-1}[2(2\alpha Zr)^{1/2}] - \gamma(2\alpha Zr)^{-1/2}J_{2\gamma}[2(2\alpha Zr)^{1/2}] \right\}.$$

$$G_\kappa = -(\kappa/|\kappa|)(\alpha Z)^{-1/2}\left\{(2\alpha Zr)^{1/2}J_{2\gamma-1}[2(2\alpha Zr)^{1/2}] - (\gamma+\kappa)J_{2\gamma}[2(2\alpha Zr)^{1/2}]\right\}. \tag{12}$$

In the positron case ($\nu < 0$), the exponentials no longer cancel, but lead instead to a factor $\exp(-|\nu|\pi)$ which tends to zero. Thus, the positron wavefunction vanishes. Actually, the limits were attained while holding r fixed and finite, and the wave function behaves as a delta function at infinity. The physical explanation is that the positron is repelled by the (positive charge) Coulomb center of force, and can only approach it to the extent that its momentum can overcome the repulsion; as the kinetic energy goes to zero, so does the probability of the positron being within a finite distance of the center of force.

The matrix elements for Bremsstrahlung involve in the integrand a product of initial and final Coulomb wave functions. For a positron, the overlap vanishes as the kinetic energy of the positron after interaction goes to zero. Thus, while an electron can give up essentially all its kinetic energy to the photon in Bremsstrahlung, a positron cannot. On the other hand, the positron can lose all of its energy (including rest energy) through annihilation to photons.

IX. STARTING CONDITIONS FOR INTEGRATION

Near the origin the screened nuclear potential has the form $V \approx C - Ze^2/r$, where C is a constant contributed by the potential of the electrons. If this relation is substituted into (Chapter II, Eq. 7) the coupled equations become

$$\left.\begin{aligned} F' - \frac{\kappa}{r} F + \frac{Ze^2}{r} G + (E_o - 1)\, G = 0 \\ G' + \frac{\kappa}{r} G - \frac{Ze^2}{r} F - (E_o + 1)\, F = 0 \end{aligned}\right\}, \tag{1}$$

where $E_o = E + C$. These equations, which are essentially radial equations with a pure Coulomb potential, provide a means of obtaining the form of the wavefunction and its derivative near the origin.

In (1) use will be made of the operator notation

$$r^n \frac{d^n}{dr^n} = \delta(\delta-1) \cdots\cdots (\delta-n+1). \tag{2}$$

This is done for convenience since it can be shown that

$$f(\delta) r^n = f(n) r^n. \tag{3}$$

Thus, when (1) is multiplied through by r and the operator notation is used, the coupled equations become

$$\left.\begin{aligned} \delta F - \kappa F + Ze^2 G + (E_o - 1) rG = 0 \\ \delta G + \kappa G - Ze^2 F - (E_o + 1) rF = 0 \end{aligned}\right\}. \tag{4}$$

Now substitute

$$\left.\begin{aligned} G = \sum_{\ell=0}^{\infty} a_\ell r^{\ell+\gamma} \\ F = \sum_{\ell=0}^{\infty} b_\ell r^{\ell+\gamma} \end{aligned}\right\} \tag{5}$$

into (4) and use (3) to get

$$[(\gamma+\kappa)a_o - Ze^2 b_o]r^\gamma + \sum_{\ell=1}^{\infty} [(\ell+\gamma)a_\ell + \kappa a_\ell - Ze^2 b_\ell - (E_o+1)b_{\ell-1}]r^{\ell+\gamma} = 0$$

$$[(\gamma-\kappa)b_o + Ze^2 a_o]r^\gamma + \sum_{\ell=1}^{\infty} [(\ell+\gamma)b_\ell - \kappa b_\ell + Ze^2 a_\ell + (E_o-1)a_{\ell-1}]r^{\ell+\gamma} = 0. \quad (6)$$

When the coefficient of each power of r is set equal to zero in (6), then the relationship between the coefficients of (5) is established. The relationship between the first few coefficients is given by the following equations:

$$a_o = \frac{Ze^2}{\gamma+\kappa} b_o, \quad (7)$$

$$b_o = \frac{Ze^2}{\kappa-\gamma} a_o, \quad (8)$$

with

$$\gamma = \left|(\kappa^2 - Z^2e^4)^{1/2}\right| \quad (9)$$

and

$$a_1 = -b_o \left[\frac{(\kappa-\gamma-1)(E_o+1) + (\kappa-\gamma)(E_o-1)}{1+2\gamma}\right], \quad (10)$$

$$b_1 = -a_o \left[\frac{(\kappa+\gamma+1)(E_o-1) + (\kappa+\gamma)(E_o+1)}{1+2\gamma}\right]. \quad (11)$$

The positive square root in (9) is taken so the wavefunctions will be finite at the origin.

For Z close to zero, the denominator in (7) is small for $\kappa<0$ while the denominator of (8) is small for $\kappa>0$. Thus, in general, to make the program applicable for all Z, the procedure adopted for selecting the coefficients was to choose b_o and use (7) to obtain a_o when $\kappa>0$, and to choose a_o and use (8) to obtain b_o when $\kappa<0$.

Since the wavefunction and its first derivative are both zero at the origin, as can be seen from (5) through (11), there is difficulty with starting the numerical integration. This trouble can be circumvented by making a transformation to the functions

$$\left.\begin{aligned} G_o &= r^{-\gamma} G \\ F_o &= r^{-\gamma} F \end{aligned}\right\} \tag{12}$$

which satisfy the equations

$$\left.\begin{aligned} F_o' + \frac{(\gamma-\kappa)}{r} F_o + (E-V-1)G_o &= 0 \\ G_o' + \frac{(\gamma+\kappa)}{r} G_o - (E-V+1)F_o &= 0 \end{aligned}\right\} . \tag{13}$$

Thus, at the origin,

$$\left.\begin{aligned} G_o = a_o \quad , \quad F_o = b_o, \\ G_o' = a_1 \quad , \quad F_o' = b_1. \end{aligned}\right\} \quad r = 0 \tag{14}$$

The procedure is to integrate (13) out to some convenient radius (e.g. r = 1) and then transform to (Chapter II, Eq. 7).

X. PHASE SHIFTS (NON-COULOMB)

The free particle solution to (Chapter II, Eq. 7) is required to determine the normalization and phase shift for the numerical solution of the radial equation. These are obtained from (Chapter II, Eq. 7) with V=0 as

$$G_f = \sqrt{\frac{E+1}{\pi k}}\ \ kr\ \ j_{\ell_\nu}(kr) \sim \sqrt{\frac{E+1}{\pi k}}\ \sin\left(kr - \frac{\ell_\nu \pi}{2}\right)$$

$$F_f = \sqrt{\frac{E-1}{\pi k}}\ \frac{\nu}{|\nu|}\ kr\ \ j_{\ell_{-\nu}}(kr) \sim \sqrt{\frac{E-1}{\pi k}}\ \cos\left(kr - \frac{\ell_\nu \pi}{2}\right), \tag{1}$$

where j_ℓ is a spherical Bessel function. Also required are the two new functions S(r) and C(r) which are defined by the equations

$$G = \sqrt{\frac{E+1}{E-1}}\ S\,F_f + C G_f \sim \sqrt{\frac{E+1}{\pi k}}\left[\sin\delta_\nu \cos\left(kr - \frac{\ell_\nu \pi}{2}\right) + \cos\delta_\nu \sin\left(kr - \frac{\ell_\nu \pi}{2}\right)\right]$$

$$F = C\,F_f - \sqrt{\frac{E-1}{E+1}}\ S G_f \sim \sqrt{\frac{E-1}{\pi k}}\left[\cos\delta_\nu \cos\left(kr - \frac{\ell_\nu \pi}{2}\right) - \sin\delta_\nu \sin\left(kr - \frac{\ell_\nu \pi}{2}\right)\right]. \tag{2}$$

From the asymptotic forms given in Chapter III, Eqs. 13 and 14 one can deduce

$$\left.\begin{aligned} C(r) &\sim \cos\delta_\nu \\ S(r) &\sim \sin\delta_\nu \end{aligned}\right\}, \tag{3}$$

$$C^2(r) + S^2(r) \sim 1. \tag{4}$$

If G_N and F_N are the numerical solutions to the radial wave equations, they will not be properly normalized because of the arbitrary selection of one constant in the starting conditions. (The procedure is to start with the normalization appropriate to a pure Coulomb potential.) Thus they will differ from the correct solution by a normalization constant N such that $F=NF_N$ and $G=NG_N$. Substitution of these equations into (2) yields

$$S(r) = N \frac{\left(\frac{G_N}{\sqrt{E+1}} \frac{F_f}{\sqrt{E-1}} - \frac{F_N}{\sqrt{E-1}} \frac{G_f}{\sqrt{E+1}} \right)}{\left(\frac{F_f^2}{E-1} + \frac{G_f^2}{E+1} \right)} \qquad (5)$$

$$C(r) = N \frac{\left(\frac{G_N G_f}{E+1} + \frac{F_N F_f}{E-1} \right)}{\left(\frac{F_f^2}{E-1} + \frac{G_f^2}{E+1} \right)}$$

Upon completion of the numerical integration, Eqs. 4 and 5 are used to determine the phase shifts.

XI. W.K.B. APPROXIMATION AND NORMALIZATION

The radial equations can be combined into the second order equation

$$G_\kappa'' + (E-V+1)^{-1}V'G_\kappa' + \left\{ (E-V)^2-1-[\kappa/(\kappa+1)]/r^2]+(\kappa/r)(E-V+1)^{-1}V' \right\} G_\kappa=0. \quad (1)$$

To get a form appropriate for a W.K.B. treatment, the first derivative term must be eliminated. This is achieved by the change of dependent variable

$$R_\kappa(r) = (E-V+1)^{-1/2}\, G_\kappa(r) \quad (2)$$

which results in

$$R_\kappa'' + \left[- \frac{V''}{2(E-V+1)} - \frac{3}{4}\frac{V'^2}{(E-V+1)^2} + \frac{\kappa}{r}\frac{V'}{E-V+1} + (E-V)^2-1- \frac{\kappa(\kappa+1)}{r^2}\right] R_\kappa=0. \quad (3)$$

This can now be assumed of the form

$$R_\kappa'' + p^2(r)\, R_\kappa = 0 \quad (4)$$

for r sufficiently large that the quantity in brackets is positive. The formal analogy with the non-relativistic solution can be extended by writing

$$p^2(r) = k^2-U(r)-\ell_\kappa(\ell_\kappa+1)/r^2, \quad (5)$$

so that the known W.K.B. approximation of the more familiar case can be taken over, using the "equivalent potential" U(r). For this purpose, note that

$$\kappa(\kappa+1) = \ell_\kappa(\ell_\kappa+1), \quad (6)$$

$$(E-V)^2-1 = k^2-2EV+V^2, \quad (7)$$

so that

$$U(r) = (1/2)V''(E-V+1)^{-1}+(3/4)V'^2(E-V+1)^{-2}-(\kappa/r)V'(E-V+1)^{-1}+2EV-V^2. \quad (8)$$

The W.K.B. solution is

$$R_\kappa = (\pi p)^{-1/2} \sin\left(\int p\, dr + \delta_\kappa\right), \quad (9)$$

with the normalization set by comparison with the asymptotic value of the exact solution for G_κ.

The numerical solution of the exact radial equations yields unnormalized wavefunctions. The normalization is obtained by proceeding out to an r sufficiently large that the asymptotic form of the wavefunction is attained, and setting the normalization by comparison with the known normalized asymptotic wavefunction. This usually means that, for the sake of ascertaining the normalization, the numerical integration has to be carried well past the values of r necessary for the evaluation of the matrix element integrals.

For a well-behaved potential, the W.K.B. solution will satisfactorily approximate the exact wavefunction long before the asymptotic region is reached. This suggests the alternative approach of obtaining the normalization by comparing the numerical unnormalized wavefunction with the W.K.B. solution. The procedure then becomes that of carrying out the numerical solution as far as it is needed for the evaluation of the matrix elements. At the cut-off radius, the wave matching is then carried out as described below. Note that the normalization must precede correction of the matrix elements for the tail contribution (Chapter VI).

Denoting by $\bar{R}_\nu$ the unnormalized numerical solution of the exact equations for R_ν and by A the associated normalization constant, matching $\bar{R}_\nu$ to the W.K.B. solution yields

$$\bar{R}_\nu = A(\pi p)^{-1/2} \sin \left(\int p \, dr + \delta_\nu \right). \tag{10}$$

If the W.K.B. solution is a reasonable approximation at the value of r being considered, the derivatives can also be matched:

$$\bar{R}'_\nu = -(p'/2p)\, \bar{R}_\nu + A(\pi p)^{-1/2}\, p \cos \left(\int p \, dr + \delta_\nu \right), \tag{11}$$

or more conveniently,

$$A(\pi p)^{-1/2} \cos\left(\int p\,dr + \delta_\mu\right) = p^{-1}\,\bar{R}_\mu' + (p'/2p^2)\,\bar{R}_\mu . \qquad (12)$$

Squaring and summing Eqs. 10 and 12,

$$A^2 = \pi p \left\{ \bar{R}_\mu^2 + \left[p^{-1}\bar{R}_\mu' + (p'/2p^2)\,\bar{R}_\mu\right]^2 \right\} . \qquad (13)$$

For an r below the validity limit for the W.K.B. approximation, the above expression for $\bar{R}_\mu$ can still be used provided that A and δ_μ are considered functions of r instead of constants. When both are allowed to vary, the description is underdetermined. The resultant freedom permits an arbitrary choice of a supplementary relation. A legitimate choice is to require that $\bar{R}_\mu'$ retain the form given above, i.e. that the additional terms obtained on differentiating the expression for $\bar{R}_\mu$ cancel:

$$A' \sin\left(\int p\,dr + \delta_\mu\right) + A\,\delta_\mu' \cos\left(\int p\,dr + \delta_\mu\right) = 0. \qquad (14)$$

The expression given for A^2 is then still valid, but the value obtained may vary with r. The normalization constant has been attained, then, if $|dA/d(kr)| \ll |A|$ for all r above the value in question. To obtain A', differentiate Eq. 12:

$$A' \cos\left(\int p\,dr + \delta_\mu\right) - A\,p \sin\left(\int p\,dr + \delta_\mu\right) - A\,\delta_\mu' \sin\left(\int p\,dr + \delta_\mu\right)$$

$$= \pi^{1/2}\,p^{-1/2}\,\bar{R}_\mu'' + (\pi^{1/2}/2)\,\bar{R}_\mu\,[p''p^{-3/2} - (3/2)p'^2p^{-5/2}]. \qquad (15)$$

Substituting Eqs. 4 and 10,

$$\pi^{1/2}p^{-1/2}\bar{R}_\mu'' = -\pi^{1/2}p^{3/2}\bar{R}_\mu = -Ap \sin\left(\int p\,dr + \delta_\mu\right). \qquad (16)$$

Using also Eq. 14,

$$A' \cos\left(\int p\,dr + \delta_\mu\right) - A\,\delta_\mu' \sin\left(\int p\,dr + \delta_\mu\right) = A'/\cos\left(\int p\,dr + \delta_\mu\right)$$

$$= A \sin\left(\int p\,dr + \delta_\mu\right)[(1/2)\,p''p^{-2} - (3/4)p'^2p^{-3}]. \qquad (17)$$

To eliminate the masking effect of oscillations of the sinusoidal factor, note that

$$\left|\sin\left(\int p\,dr + \delta_\ell\right)\cos\left(\int p\,dr + \delta_\ell\right)\right| \leq 1/2. \qquad (18)$$

The relation $k^{-1}|A'| \ll A$ then reduces to

$$(8kp^3)^{-1}\left|2p''p-3p'^2\right| \ll 1. \qquad (19)$$

It should be noted that the procedure outlined results in the evaluation of the normalization factor while leaving the phase shift undetermined. In principle, the W.K.B. approximation can also be used to compute the phase shift. This requires, however, the evaluation of $\int p\,dr$ out to infinity (or at least to the asymptotic region) and this has to be done numerically because of the complicated expression for p.

The W.K.B. approximation as applied to the normalization of the wavefunctions requires an expression involving the potential and its first three derivatives. For a screened potential, which is known only numerically, computation of these derivatives by differencing would be unreliable. Instead, use is made of the fact that the screening factor (ratio of screened to unscreened potential) is nearly exponential. The exponent appropriate to the radius at which the numerical integration stops is obtained by taking the logarithm of the ratio of the screening factor at that radius to the screening factor at a slightly larger radius. The analytical form of the screening factor thus arrived at is then used to calculate the derivatives. This procedure has been used to obtain the normalization of the wavefunctions in the screened case.

XII. SCREENED POTENTIAL

The electron wavefunctions are obtained by integrating the Dirac radial equations numerically. When screening is invoked, the potential used in these equations is the Coulomb potential multiplied by the ratio of screened-to-unscreened potentials. This ratio is obtained by reading in a tabulation from tape and interpolating (the ratio is used instead of the potential itself because it is a smoother, more slowly varying function of r). The tabulation came from the output of a relativistic Hartree-Fock-Slater self-consistent-field calculation[14] made available to us by Dr. James T. Waber.

XIII. ANGULAR DISTRIBUTION OF THE SCATTERED ELECTRONS

In the earlier chapters, the Bremsstrahlung cross section has been considered under the constraint that the photon is observed but the scattered electron is not. The final electron state is then represented by a spherical wave $\varphi_{\kappa',\mu'}$, and the cross section calculation includes a sum over the quantum numbers κ' and μ'.

The present chapter deals with the cross section under the converse constraint: the photon is not observed, but the scattered electron is. The final state electron function $\Psi_f(E')$ must now be in a definite spin state (z-component m") and be a wave which behaves asymptotically as a plane wave plus convergent spherical wave. Explicitly, it is given by[7]

$$\Psi_{m''}(E') = 4\pi^{3/2} (2E'k')^{-1/2} \sum_{\kappa'\mu'} C'_{\kappa',\mu',m''} \varphi_{\kappa',\mu}, Y^*_{\ell_{\kappa'},\mu'-m''}(\hat{k}') \qquad (1)$$

with

$$C'_{\kappa',\mu',m''} = i^{\ell_{\kappa'}+1} e^{-i\delta_{\kappa'}} C(\ell_{\kappa'}, \tfrac{1}{2}, j_{\kappa'}; \mu'-m'',m''), \qquad (2)$$

differing from the form of the initial state electron function (Chapter III, Eqs. 4 and 5) by a complex conjugation (convergent instead of divergent spherical wave) and by omission of the specialization to propagation along the z-axis.

In the cross section itself, the density of final electron states must now appear, leading to an additional multiplicative factor of $k'E'd\Omega_e/(2\pi)^3$. On the other hand, since the photon is not observed there is an integration over its direction $d\Omega_q$ and a sum over its polarization p (in fact, it is obvious that the angular integration will eliminate any polarization-dependent terms).

The matrix element $M(pmn'\epsilon)$ of Chapter III (with the change of variable $\epsilon = \mu' - m$ used there) is now replaced by

$$M'(pmm'') = 4\pi^{3/2}(2E'k')^{-1/2} \sum_{\kappa'\epsilon} i^{-\ell_{\kappa'}-1} e^{i\delta_{\kappa'}} C(\ell_{\kappa'}, \tfrac{1}{2}, j_{\kappa'}; \epsilon+m-m'', m'') \tag{3}$$

$$Y_{\ell_{\kappa'}, \epsilon+m-m''}(\hat{k}') \; M(pm\kappa'\epsilon).$$

Factoring out the dependence on the photon angles,

$$M'(pmm'') \equiv \sum_{\lambda\epsilon} \bar{M}'(\lambda\epsilon pmm'') \; D^{\lambda}_{\epsilon p}(\varphi_q, \theta_q, 0)$$

and the integration over photon angles is trivial[8]

$$\int d\Omega_q \; D^{\lambda}_{\epsilon p}(\varphi_q, \theta_q, 0) \; D^{\lambda_1 *}_{\epsilon' p'}(\varphi_q, \theta_q, 0) = \frac{4\pi}{2\lambda+1} \delta_{\lambda\lambda_1} \delta_{\epsilon\epsilon'} \delta_{pp'}, \tag{4}$$

leading to

$$\int d\Omega_q \; |M'(pmm'')|^2 = 4\pi \sum_{\lambda\epsilon} (2\lambda+1)^{-1} \; |\bar{M}'(\lambda\epsilon pmm'')|^2 \tag{5}$$

where explicitly (see Chapter III, Eq. 15 and Chapter V, Eq. 9)

$$\bar{M}'(\lambda\epsilon pmm'') = 4\pi^3 e \sqrt{\frac{6}{qEE'kk'}} \sum_{\kappa'\ell} i^{-\ell_{\kappa'}-1} e^{i\delta_{\kappa'}} C(\ell_{\kappa'}, \tfrac{1}{2} j_{\kappa'}; \epsilon+m-m'', m'') Y_{\ell_{\kappa'}, \epsilon+m-m''}(\hat{k}')$$

$$pC(\ell 1\lambda; 0p)(-)^{m+\frac{1}{2}} \sum_{\kappa} \bar{H}(\lambda\ell\kappa'\kappa) C(\lambda j_\kappa j_{\kappa'}; \epsilon m) C(j_\kappa \tfrac{1}{2} \ell_\kappa; -mm). \tag{6}$$

Application of the addition theorem for spherical harmonics

$$Y_{\ell_{\kappa'}, \epsilon+m-m''}(\hat{k}') \; Y^{*}_{\ell_{\kappa_1'}, \epsilon+m-m''}(\hat{k}') = (-1)^{\epsilon+m-m''} (4\pi)^{-1} \sqrt{(2\ell_{\kappa_1'}+1)(2\ell_{\kappa'}+1)} \tag{7}$$

$$\sum_j C(\ell_{\kappa_1'}, \ell_{\kappa'}, j; 00) C(\ell_{\kappa_1'}, \ell_{\kappa'}, j; -\epsilon-m+m'', \epsilon+m-m'') P_j(\cos\theta_{k'})$$

reduces the angular distribution to an expansion in Legendre polynomials of the cosine of the angle between incident and scattered electron.

In the calculation of the cross section, there is an average over m (the spin orientation of the incident electron is immaterial) and a sum over m" and p (the Bremsstrahlung process will be observed regardless of the spin orientation of the scattered electron or the polarization state of the photon). Gathering all the factors,

$$d\sigma = \frac{\pi E}{k} \frac{k'E'q^2}{(2\pi)^6} d\Omega_e(-dE') \sum_{mm''p} \int d\Omega_q \, |M'|^2. \tag{8}$$

The polarization sum is

$$\sum_p p^2 C(\ell 1\lambda;0p)C(\ell_1 1\lambda;0p) = 2C(\ell 1\lambda;01)C(\ell_1 1\lambda;01) \tag{9}$$

provided that

$$\ell + \ell_1 = \text{even integer} \tag{10}$$

and vanishes otherwise.

The magnetic quantum number sums can be carried out explicitly. For m", there results

$$\sum_{m''}(-1)^{\epsilon+m-m''} C(\ell_\nu,\tfrac{1}{2}j_\nu;\epsilon+m-m'',m'')C(\ell_{\nu_1},\tfrac{1}{2}j_{\nu_1};\epsilon+m-m'',m'')C(\ell_{\nu_1},\ell_\nu,j;-\epsilon-m+m'',\epsilon+m-m'')$$

$$=(-1)^{\ell_{\nu_1},-\ell_\nu,+j_\nu,-\frac{1}{2}} \sqrt{(2j+1)(2j_{\nu_1}+1)}\; W(\ell_{\nu_1},j_{\nu_1},\ell_\nu,j_\nu;\tfrac{1}{2}j)C(j_{\nu_1},jj_\nu;\epsilon+m,0). \tag{11}$$

Together with factors from Eq. 7, the Racah coefficient reduces to

$$\sqrt{(2\ell_\nu+1)(2\ell_{\nu_1}+1)}\; C(\ell_{\nu_1},\ell_\nu,j;00)W(\ell_{\nu_1},j_{\nu_1},\ell_\nu,j_\nu;\tfrac{1}{2}j)=C'(j_\nu,j_{\nu_1},j;\tfrac{1}{2},-\tfrac{1}{2}) \tag{12}$$

provided that the left-hand side does not vanish, i.e.

$$\ell_\nu,+\ell_{\nu_1},+j = \text{even integer} \tag{13}$$

and the three angular momenta in Eq. 13 satisfy a triangular inequality; the prime will be carried along on the C as a reminder of the implied constraint. Next, the ϵ sum is

$$\sum_{\epsilon} C(j_{n_1} j j_n ; \epsilon+m, 0) C(\lambda j_n j_n' ; \epsilon m) C(\lambda j_{n_1} j_n' ; \epsilon m)$$

$$= (-1)^{j_{n_1}+j_{n_1}'+j_n-\lambda-j-m} (2j_{n_1}'+1)\sqrt{(2j_{n_1}'+1)/(2j+1)}\ W(j_n j_{n_1} j_n', j_{n_1}' ; j\lambda) C(j_{n_1} j_n j ; m, -m). \tag{14}$$

The m sum is the same as in Chapter V, Eq. 14 and again gives

$$\sum_{m} (-1)^{m+\frac{1}{2}} C(j_{n_1} j_n j ; m, -m) C(j_{n_1} \tfrac{1}{2} \ell_{n_1} ; -m, m) C(j_n \tfrac{1}{2} \ell_n ; -m, m)$$

$$= (-1)^{j_n+j_{n_1}-j} C(j_n j_{n_1} j ; \tfrac{1}{2}, -\tfrac{1}{2}) \tag{15}$$

though the argument for the equality of the two terms with $m=\pm\frac{1}{2}$ is a bit different, namely a change of sign in m introduces the factor (-1) to the power

$$1+(j_{n_1}+j_n-j)-(j_{n_1}+\tfrac{1}{2}-\ell_{n_1})-(j_n+\tfrac{1}{2}-\ell_n) = \ell_{n_1}+\ell_n-j \tag{16}$$

$$= 2(j_{n_1}+j_n-j)-(\ell_{-n_1}+\ell_{n_1}'+\ell_1)-(\ell_{-n}+\ell_n'+\ell)+(\ell+\ell_1)+(\ell_{n_1}'+\ell_n'+j)$$

which is even since the second and third terms in the last expression must be for non-vanishing matrix elements, the last two according to Eqs. 10 and 13.

The differential cross section is

$$d\sigma = -3\pi e^2 (q/k^2) d\Omega_e dE' \sum_j (-1)^j P_j(\cos\theta_{k'}) \sum_{\lambda} (-1)^{\lambda} (2\lambda+1)^{-1}$$

$$\sum_{n n_1 n' n_1'} (-1)^{j_{n'}+j_{n_1}'} i^{\ell_{n_1}'-\ell_n'} e^{i(\delta_{n'}-\delta_{n_1'})} (2j_{n'}+1)(2j_{n_1}'+1) C(j_n j_{n_1} j ; \tfrac{1}{2}, -\tfrac{1}{2}) C'(j_{n'} j_{n_1}' j ; \tfrac{1}{2}, -\tfrac{1}{2})$$

$$\sum_{\ell \ell_1}{}' C(\ell 1 \lambda ; 0 1) C(\ell_1 1 \lambda ; 0 1) W(j_n j_{n_1} j_{n'} j_{n_1}' ; j\lambda) \bar{H}(\lambda \ell n' n) \bar{H}^*(\lambda \ell_1 n_1' n_1) \tag{17}$$

where the primes refer to the constraints of Eqs. 13 and 10, respectively. To eliminate complex numbers, make the variable change (see Chapter V, Eqs. 7, 10, and 24).

$$i^{-\ell_\nu'}\,\bar{H}(\lambda\ell\nu'\nu) = -i^{\ell+\ell_\nu-\ell_\nu',+1}\,e^{i\delta_\nu}\,(2\ell+1)\sqrt{(2j_\nu+1)/3}\;H(\lambda\ell\nu'\nu). \quad (18)$$

The power of i is even

$$\ell+\ell_\nu-\ell_\nu',+1 = (\ell+\ell_\nu+\ell_{-\nu}',)+1-(\ell_\nu',+\ell_{-\nu}',) = (\ell+\ell_\nu+\ell_{-\nu}',)+(1-2j_\nu',) \quad (19)$$

so that

$$i^{\ell+\ell_\nu-\ell_\nu',+1} = (-1)^{(\ell+\ell_\nu+\ell_{-\nu}',)/2}\,(-1)^{1/2-j_\nu'}. \quad (20)$$

The ℓ and ℓ_1 sums can be done separately provided Eq. 10 is simultaneously satisfied. This requires splitting the sum into an even ℓ and an odd ℓ part, namely

$$T_1(\lambda\nu'\nu) \equiv \sum_{\ell\ \text{even}} (-1)^{(\ell+\ell_\nu+\ell_{-\nu}',)/2}(2\ell+1)C(\ell 1\lambda;01)H(\lambda\ell\nu'\nu), \quad (21)$$

$$T_2(\lambda\nu'\nu) \equiv \sum_{\ell\ \text{odd}} (-1)^{(\ell+\ell_\nu+\ell_{-\nu}',)/2}(2\ell+1)C(\ell 1\lambda;01)H(\lambda\ell\,\nu'\nu). \quad (22)$$

The cross section then reads

$$d\sigma = \pi e^2(q/k^2)dE'd\Omega_e \sum_j (-1)^j P_j(\cos\theta_{k'}) \sum_{\nu\nu_1\nu'\nu_1'} e^{i(\delta_\nu+\delta_{\nu'},-\delta_{\nu_1}-\delta_{\nu_1'})}$$

$$(2j_{\nu'},+1)(2j_{\nu_1'},+1)\sqrt{(2j_\nu+1)(2j_{\nu_1}+1)}\;C(j_\nu j_{\nu_1} j;\tfrac{1}{2},-\tfrac{1}{2})C'(j_{\nu'},j_{\nu_1'},j;\tfrac{1}{2},-\tfrac{1}{2}) \quad (23)$$

$$\sum_\lambda (-1)^\lambda (2\lambda+1)^{-1}(2\lambda+1)^{-1}\,W(j_\nu j_{\nu_1} j_{\nu'}, j_{\nu_1'};j\lambda)\sum_{n=1}^{2} T_n(\lambda\nu'\nu)T_n(\lambda\nu_1'\nu_1).$$

An interchange of the designation of the pairs of summation variables (ν,ν_1) and (ν',ν_1') must leave the cross section unchanged. This interchange affects only the appearance of the W and the two C coefficients, none of which change value as a result, and changes the phase factor to its complex conjugate. On averaging the results of the two equivalent conventions, the exponential is replaced by the cosine of the phase. Then

$$d\sigma = \pi e^2 (q/k^2) dE' d\Omega_e \sum_j (-1)^j P_j(\cos\theta_{k'}) \sum_{\nu\nu_1\nu'\nu_1'} \cos(\delta_\nu + \delta_{\nu'} - \delta_{\nu_1} - \delta_{\nu_1'})$$

$$(2j_{\nu'}+1)(2j_{\nu_1'}+1)\sqrt{(2j_\nu+1)(2j_{\nu_1}+1)}\; C(j_\nu j_{\nu_1} j;\tfrac{1}{2},-\tfrac{1}{2}) C'(j_{\nu'} j_{\nu_1'} j;\tfrac{1}{2},-\tfrac{1}{2}) \tag{24}$$

$$\sum_\lambda (-1)^\lambda (2\lambda+1)^{-1} W(j_\nu j_{\nu_1} j_{\nu'} j_{\nu_1'};j\lambda) \sum_n T_n(\lambda\nu'\nu)\, T_n(\lambda\nu_1'\nu_1).$$

For the cross section integrated over all angles,

$$\int d\Omega_e\, P_j(\cos\theta_{k'}) = 4\pi\, \delta_{j0}. \tag{25}$$

With j=0, the W and C coefficients vanish unless $j_\nu = j_{\nu_1}$ and $j_{\nu'} = j_{\nu_1'}$. But in addition, from Eqs. 16 and 13, $\ell_\nu + \ell_{\nu_1}$ = even and $\ell_{\nu'} + \ell_{\nu_1'}$ = even. These together require $\nu = \nu_1$ and $\nu' = \nu_1'$. The coefficients then reduce to statistical factors, and the cross section becomes

$$d\sigma_{int} = -4\pi^2 e^2 (q/k^2) dE' \sum_{\lambda\nu\nu'} [(2j_{\nu'}+1)/(2\lambda+1)] \sum_n [T_n(\lambda\nu'\nu)]^2. \tag{26}$$

The Bremsstrahlung cross section integrated over all angles could equally well have been obtained by starting from the photon angular distribution and integrating it over the photon angles. This will now be done as a check on the calculations. From Chapter IV, Eq. 9,

$$\int d(\cos\theta_q)\, (d\sigma_{p'p}) = 2\,\delta_{pp'}\, d\sigma^0_{pp}. \tag{27}$$

Using the fact that the two p values make equal contributions and the relations among y, q, E, and k, Eqs. 10 and 11 of Chapt. IV yield

$$d\sigma_{int} = 4\, d\sigma^{\partial}_{11} = 12\pi^2 e^2 (q/k^2)\, dq \sum'_{\ell\ell_1} A_0\,(11\ell_1\ell) \tag{28}$$

with the constraint of Chapt. V Eq. 16 on the double sum specializing to Eq. 10. Substituting the value of the C-coefficient with j = 0, Chapt. IV Eq. 12 reduces to

$$A_0\,(11\ell_1\ell) = \sum_{\lambda} (-1)^{\lambda+1}(2\lambda+1)^{-1/2}\, C(\ell_1 1\lambda;01)C(\ell 1\lambda;01)V_0(\lambda\lambda\ell_1\ell) \tag{29}$$

and Chapt. IV Eq. 13 to

$$V_0\,(\lambda\lambda\ell_1\ell) = (-1)^{\lambda+1}\sum_{\epsilon}\phi(\lambda\lambda\ell_1\ell\epsilon). \tag{30}$$

From Chapt. V Eqs. 25 and 26, the ϵ sum is now merely

$$\sum_{\epsilon} C(\lambda_1 j_\nu, j_{\nu_1};\ \epsilon, -\epsilon+1/2)\ C(\lambda j_\nu, j_\nu;\ \epsilon,\ -\epsilon+1/2) = \delta_{j_\nu j_{\nu_1}} \tag{31}$$

and the parity condition expands this selection rule to $\nu=\nu_1$. Substituting Chapt. V Eq. 27 into Chapt. V Eq. 26, there remains

$$A_0(11\ell_1\ell)=(1/3)(2\ell+1)(2\ell_1+1)\sum_{\lambda}(2\lambda+1)^{-1}C(\ell_1 1\lambda;01)\ C(\ell 1\lambda;\ 01)$$

$$\sum_{\nu'\nu}(2j_{\nu'}+1)(-1)^{(\ell_1+\ell_\nu+\ell_{-\nu'})/2}\ H(\lambda\ell_1\nu'\nu)(-1)^{(\ell+\ell_\nu+\ell_{-\nu'})/2} \tag{32}$$

$$H(\lambda\ell\nu'\nu).$$

Using Eqs. 21 and 22, this leads to Eq. 26 as expected (remembering that $dq = -dE'$).

Returning to the electron angular distribution (Eq. 23), there is a computational advantage to the elimination of the Racah coefficient at the cost of reintroducing a magnetic quantum number sum:

$$(-1)^{j+\lambda}\ W(j_\nu j_{\nu_1} j_{\nu'} j_{\nu'_1};j\lambda)\ C(j_\nu j_{\nu_1} j;\ 1/2,\ -1/2)$$

$$= -(2\lambda+1)^{-1}\sum_{\beta}(-1)^{\beta}\ C(j_\nu j_{\nu'},\lambda;\ -1/2,\ \beta+1/2)\ C(j_{\nu_1} j_{\nu'_1},\lambda;\ -1/2,\ \beta+1/2)$$

$$C(j_{\nu'}, j_{\nu'_1} j;\ \beta+1/2,\ -\beta-1/2). \tag{33}$$

The κ and κ_1 sums are now decoupled. Setting

$$U_n(\lambda\kappa'\beta) \equiv \sum_{\kappa} e^{i\delta_\kappa} \sqrt{2j_\kappa+1} \; C(j_\kappa j_\kappa, \lambda; -1/2, \beta+1/2)\, T_n(\lambda\kappa'\kappa), \tag{34}$$

$$d\sigma = \pi e^2 (q/k^2) dE' \, d\Omega_e \sum_j P_j(\cos\theta_{k'}) \sum_{\kappa'\kappa_1'} e^{i(\delta_{\kappa'} - \delta_{\kappa_1'})} (2j_{\kappa'}+1)(2j_{\kappa_1'}+1)$$

$$C'(j_{\kappa'}, j_{\kappa_1'}, j; 1/2, -1/2) \sum_\beta (-1)^\beta C(j_{\kappa'}, j_{\kappa_1'}, j; \beta+1/2, -\beta-1/2)$$

$$\sum_{\lambda n} (2\lambda+1)^{-2} U_n(\lambda\kappa'\beta)\, U_n^*(\lambda\kappa_1'\beta). \tag{35}$$

Just as after Eq. 23, only the real part of Eq. 35 is needed.

A comparison of the electron and photon angular distributions shows them to be of the same order of complexity. The calculations are identical through the computation of the reduced matrix elements $H(\lambda\ell\kappa'\kappa)$, but the subsequent angular momentum sums differ.

XIV. THE COMPUTER PROGRAM

The main routine (BREMS) sets up some numerical constants, reads the input parameters and checks them, then calls subroutines to do the bulk of the calculations, and finally computes the Legendre functions and puts together the cross sections and writes them out. All ordinary input/output is done in BREMS. Supplementary diagnostic and checkout output is generated by some subroutines.

The structure of subroutine calls is schematized in Fig. 2.

The first subroutine called is SELECT. It starts from the input specification of the maximum orbital angular momenta to be considered, runs through the selection rules to determine what matrix elements will occur, and indexes these matrix elements and the corresponding quantum numbers.

If screening is to be considered, SETAPE scans the tape bearing the screening factors and finds the section of it containing the data for the required element.

FINTEG is the control subroutine for the radial matrix elements. It sets up the required arrays for the wavefunctions and matrix elements and their derivatives, computes their initial values and the coefficients of the differential equations, calls other subroutines to do the actual integrating, computes the phase shifts, then calls subroutines to normalize the matrix elements, asymptotically evaluate their tail, and form reduced matrix elements including an X-coefficient.

The integration is performed by Gill's form of the Runge-Kutta method[15] in RKUTTA, which calls DIRAC to supply the derivatives from

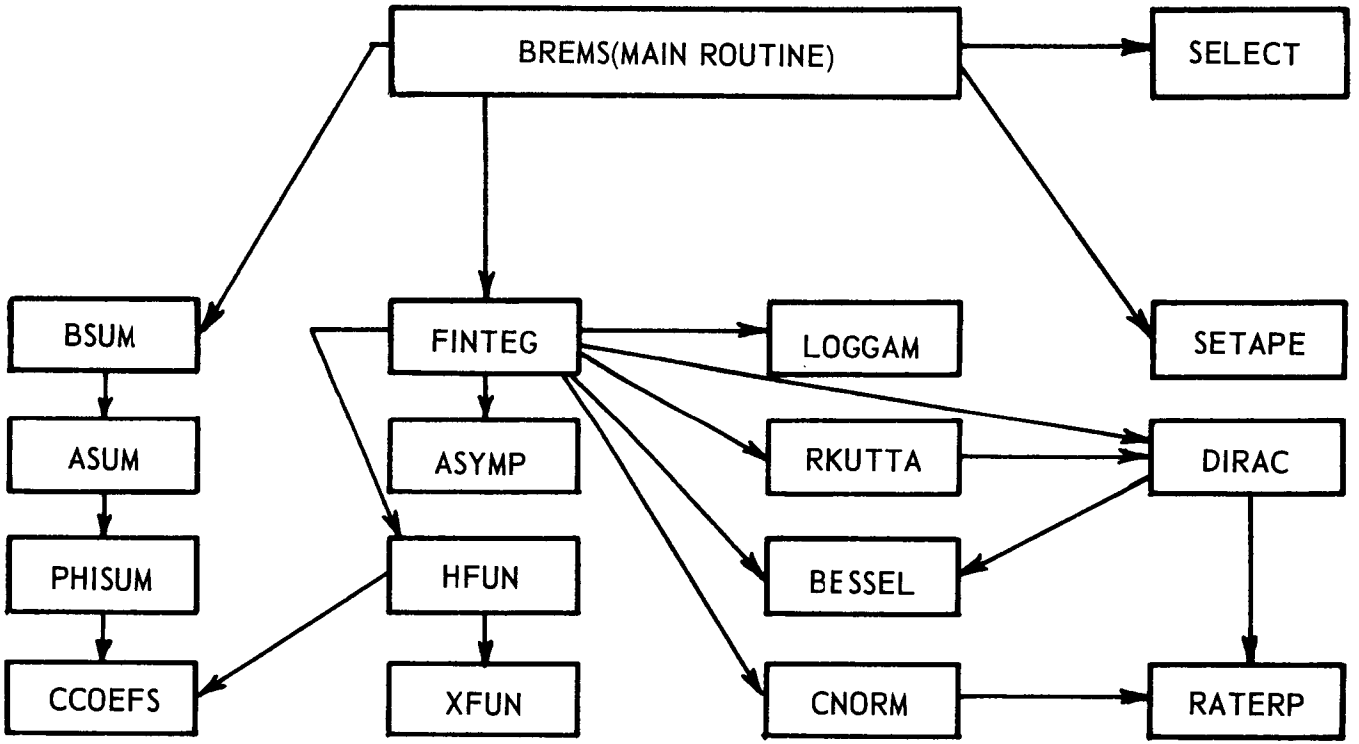

FIGURE 2 CALLING SEQUENCES

AP198-1

This flow-chart shows the communication among subroutines of the Bremsstrahlung program through calls. The arrow goes from the subroutine in which the call occurs to the subroutine being called (a return is implied). In addition, there is transfer of information through Common (labelled and unlabelled).

the differential equations. If screening is taken into account, the screening factor (ratio of screened potential to Coulomb potential) is supplied by RATERP, which reads values from tape and interpolates.

Upon completion of the integration, normalization factors are computed in CNORM by matching the wavefunctions to their W.K.B. approximation (see Chapter XI) and the matrix elements are then normalized.

The asymptotic evaluation of the tail of the matrix elements (see Chapter VI) is performed in ASYMP, and the correction is added on.

In HFUN, the calculation is carried forward to the reduced matrix elements $H(\lambda\ \ell\nu'\nu)$, incorporating an implicit sum over a couple of magnetic quantum numbers (see Chapter V, Eq. 24).

The remaining angular momentum sums are performed through the chain BSUM-ASUM-PHISUM. The sums over all ν's are performed in PHISUM, with appropriate phase shifts and Clebsch-Gordan coefficients, to yield $\overline{\varphi}(\lambda_1\lambda\ell_1\ell\epsilon)$ (see Chapter V, Eq. 26). Further sums over ϵ and the λ's are carried out in ASUM, again with Clebsch-Gordan coefficients, to yield $A_j^{(i)}(\ell_1\ell)$ (see Chapter IV, Eq. 15). BSUM sums over the ℓ's to yield $B_j^{(i)}$ (see Chapter IV, Eq. 16).

The remaining subroutines compute special functions. LOGGAM is an NYU program[46] to compute the logarithm of the gamma function for complex argument (converted to FORTRAN IV at UCC), used for the Coulomb wavefunctions and phase shifts. BESSEL computes spherical Bessel functions, using the explicit expressions in terms of trigonometric functions for ℓ=0 and 1, and recursion relations for larger ℓ. BESLIT is the same as BESSEL except that if the argument is less than 1 the power series are used for $\ell > 0$. CCOEFS computes Clebsch-Gordan coefficients, using

explicit formulas if the smallest j is 2 or less[17], the general formula otherwise[18], with a specialization for the parity C-coefficients[19]. XFUN yields the factor denoted by $G_{\ell\lambda}(\kappa',\kappa)$ in the explicit expression for the special X-coefficient arising from the magnetic quantum number sums (see Chapter V, Eq. 8).

Following Koch and Motz[2], the angular distribution is given in units of (the photon energy $/Z^2$) times the differential cross section (per unit interval of photon energy per steradian) in millibarns/steradian. This is the display favored by the experimenters, and hence deemed most convenient for practical use. On the other hand, the calculations are performed in dimensionless nuclear units ($\hbar = m = c = 1$).

XV. OPERATING INSTRUCTIONS

This chapter gives all the operational details needed in order to run the Bremsstrahlung program. The input variables and format are listed, along with the program diagnostics. Test options and their uses are described. Tape unit assignments are given. The output is outlined. The strategy of successive choices of input variables in production operation is discussed.

Input Variables

The fixed point quantities described below are read in on a single data card in format 10I5:

KEY Equal zero for pure Coulomb run, one for screened run, and minus one for program termination.

JM The highest Legendre coefficient in the angular distribution.

LM The highest orbital angular momentum coefficient for photon.

LM1 The highese orbital angular momentum coefficient for incident electron.

LM2 The highest orbital angular momentum coefficient for scattered electron.

NTAPE The logical tape unit assigned for tape containing the screening data. (This number need be assigned only for a screened run.)

NTEST Equal one for matrix reduction procedure, equal zero otherwise.

ITEST Equal one for radius increment procedure, equal zero otherwise.

INUM The maximum number of times the cut-off radius will be incremented. (This number need be assigned only if the ITEST option is utilized.)

IREP Equal one to pick up from tape of previous run and proceed to the angular momentum sums, equal two to carry the radial integration forward under the ITEST option without first summing, equal zero otherwise.

The quantities NTEST, ITEST, and IREP designate optional program features described below.

The floating-point quantities described below are read in on a single data card in format 8F10.0:

Z Nuclear charge (atomic number).

EO Kinetic energy of incident electron (in Mev).

XO Ratio of photon energy to incident kinetic energy.

RATIO The ratio of particle mass to electron mass.

ZEL Ratio of particle charge to electron charge (1.0 for electron, -1.0 for positron).

RCUT Radius at which numerical integration is terminated.

TOL The admissible fractional change specified for certain output quantities. This quantity need be assigned only if either of the program options (NTEST or ITEST) is used.

DRCUT The amount by which the terminal radius (RCUT) is incremented. This quantity need be assigned only if the ITEST procedure is utilized.

A complete set of data consists of two data cards. Upon completion of a problem, the program recycles, reading in the next pair of data cards. Termination of the entire program is obtained by setting KEY=-1 on a first card (no second card needed).

Input Testing

The program sifts the input data to insure that certain criteria are not violated. If any difficulty is observed, the specific violation is printed and the run terminated.

Listed below in abbreviated form are the criteria that must be satisfied and that the program tests for:

$JM \leq (2*LM+2)$

$MIN(JM, LM, LM1, LM2) \geq 0$

$MAX(LM, LM1, LM2) \leq 10$

$LM1 + LM2 \geq LM$

$|LM1 - LM2| \leq LM$

$0 < X0 \leq 1$

$Z \geq 0$

$ZEL \neq 0$

$NEQ \leq 1500$* where NEQ = number of matrix elements plus wavefunctions.

* The number 1500 was chosen to insure that the entire program would not exceed the storage capacity of the IBM 7094.

Program Options

The program has two option procedures designated by NTEST and ITEST. Either or both options may be used in the same run. If both are desired, the priority goes to ITEST. The results thus obtained are then used for NTEST.

NTEST (or matrix reduction procedure) examines the possibility of reducing the original angular momentum quantities, JM, LM, and LM2, without influencing the output quantities more than a given fractional amount (TOL). Since the running time of the program drastically increases as these values are increased, the knowledge of a minimum set can be of use for future runs. Another consideration for the use of this option is, if no reduction is possible (i.e. any decrement results in a "large" change of output), there exists a possibility that the

original values were not sufficiently large.

Method: After a run is completed, including the printout, the test initiates by decrementing JM by one and recomputing, then comparing the new values with the original set. If the fractional change is within the required tolerance (TOL) the procedure repeats. The first time the comparison fails, JM is incremented by one (back to the previous successful value). This value of JM is printed, then the program proceeds to decrement LM in like manner, and the process continues until we obtain a "minimum" set of values JM, LM, and LM2 (which may be the original set) that does not change the output quantities by more than the required amount.

ITEST (or radius increment procedure) investigates the radial matrix element stability. This option allows us to examine the stability of the output for different but progressively increasing cut-off radii without re-running the entire integration from the beginning.

Method: This test initiates by temporarily storing on tape the present cut-off radius along with the matrix elements computed at that radius and other necessary data. After the run is completed, i.e. the required output data is generated and stored but not printed, the tape is read back in, the integration carried forward an additional increment, DRCUT units of radius, and the matrix elements updated. Computation then proceeds as required. The newly generated output is compared to the previous set. If the fractional change is not within tolerance (TOL) the procedure repeats until success is reached or the maximum number of increments (INUM) is exceeded. After each increment, the matrix elements and their sum squared are printed for visual comparison. The

testing is always made to the previous set of values, i.e. the output variables are constantly updated. The comparison of results is printed after each computation. The entire output is printed either at the successful cut-off radius or after INUM is exceeded.

IREP was inserted to permit the program to be run in installments, whether by design or necessity. At the end of the radial integration, the matrix elements and quantum numbers in indexed form are written out on tape. The IREP option permits the program to read in the information from this tape (saved from a previous run), bypass the radial integration and proceed onward. This allows a run to be saved if it was cut off after completion of the radial integration but before the angular momentum sums were done. It also allows the test options to be applied to a run previously done without them.

Systems Information

The program as run at the Marshall Space Flight Center utilizes the following tapes:

Logical tape unit 5	Read
Logical tape unit 6	Write
Logical tape units 8, 9	Temporary storage
Logical tape unit 10	Screening data.

The FORTRAN program, of course, refers only to symbolic input-output unit designations. The actual physical unit corresponding to the symbolic unit reference has been established at the initialization of IOCS. Since symbolic unit references may differ from installation to installation, care must be taken in attempting to execute the program in making the tape unit assignments compatible with the installation in question.

The input tape unit to be mounted on logical tape unit 10 is a pre-created save tape (screening factors). The system BCD input containing the program and data cards is mounted on logical unit 5; the BCD output for later off-line printing is stored on logical unit 6. Logical tape unit 8 is used by the program to store intermediate information for later recall. In the ITEST option, this information is subsequently read back in and the corresponding data for a larger radius is written out. In the restart option (IREP), the tape saved from unit 8 of a previous run is mounted on unit 9 and read in; it is not written on.

Output

The output starts with a restatement of the input data. Intermediate output includes the number of matrix elements, the integration step interval, the normalization factors, the wavefunctions and matrix elements before applying the asymptotic correction for the tail of the matrix elements, the matrix elements after this correction, and the Legendre coefficients of the cross section. The cross section tabulations come last. Cross section output is for the photon energy/Z^2 times the differential cross section (per unit interval of photon energy per steradian) in millibarns steradian.

For the unpolarized photon, the differential cross section is tabulated per unit interval of cos θ, where θ is the angle between the photon and the incident particle, for θ values between 0 and 180 degrees in steps of 2 degrees. Finally the cross section integrated over all angles and values of cos θ and $\cos^2 \theta$ averaged over the differential cross section are listed. For the linearly polarized photon the differential cross section is listed for six values of the polarization angle (0,30, 60,90,120, and 150 degrees).

If the radius increment option (ITEST) is used, the initial

cut-off radius along with the computed values of the matrix elements and their sum squared are printed. After each radius increment, the recomputed matrix elements are printed along with the new Legendre coefficients of cross section, and a comparison of designated values of output to their newly computed counterparts. At the termination of this process the cross section tabulations for the most recent cut-off radii are listed.

If the matrix reduction option (NTEST) is utilized, we first obtain a print-out of the output computed as a result of the initial input data. As each of the angular momentum quantities JM, LM, and LM2 are consecutively decremented, their new value along with a comparison of designated initial output to the newly computed counterparts are listed. The listing continues until the test procedure is completed.

Strategy of Choice of Input

Invoking the test options obviously costs in running time. Their purpose is twofold: to ascertain that the integration cut-off radius is sufficiently large and that enough terms are kept in the partial wave expansions to achieve the desired accuracy, and also as a guide to a more restrictive (hence more economical) set of input data for subsequent runs.

The use of the ITEST option is straightforward. On the first run, one would call this option with arbitrary values of RCUT (say 100) and DRCUT (say 20); the outcome of the option is a value of RCUT satisfying the imposed tolerance condition. This new value of RCUT would then be used for the next run (it is assumed that input energy values are varied slowly and systematically). After a few runs (the energies now being appreciably different), one would again invoke ITEST,

using an RCUT input value somewhat below the current value. The option will then settle on a new acceptable RCUT value, which may lie above or below the old one.

The number of terms required in a partial wave expansion increases with the energy (more precisely, with the momentum). Hence, if a certain number of terms suffice at one energy, fewer terms will be needed at a lower energy. In a series of production runs, one would start with the highest incident energy of interest and plan to run through the spectrum of photon energies for that particular incident energy. The NTEST option yields the minimum acceptable LM and LM2 for a particular run. The recommended procedure is to run first a case with the smallest desired fraction of the energy going to the photon, using large input values of LM1 and LM2 with a reasonably low value of LM and invoking NTEST. For this case, the incident and scattered electron have comparable energies, and the output minimum tolerable LM2 is thus also a proper value for LM1. In all subsequent runs at the same incident energy, this same value is to be used for LM1. For larger photon energy, a larger LM will be required, whereas a lower LM2 will do (lower energy for scattered electron). The next run then will be for an appreciably larger fraction of the energy going to the photon, with LM1 and LM2 obtained from the previous run but LM set larger, and again invoking NTEST. Upon iteration, sets of acceptable parameters are obtained at intervals across the photon spectrum. At intermediate values, the runs would then be made without NTEST, using the LM1 value previously determined and the larger of the values of LM and LM2 from the bracketing points (one from each side). For a lower incident energy, the same procedure can be repeated, with the added knowledge that the minimum acceptable

LM, LM1, and LM2 will have the corresponding values at the higher incident energy as upper bounds. The required value of JM is also expected to decreases as the incident energy decreases (angular distribution less peaked). The number of matrix elements (hence the numerical integration time) is roughly proportional to the product of LM, LM1, and LM2, so that there is considerable machine time at stake in minimizing these values. On the other hand, the angular distribution calculation is fast, so reducing JM is not crucial.

XVI. RESULTS

A partial wave expansion (such as is used here) converges most effectively at lower energies, and the angular momentum sums can then be truncated relatively early with greater confidence. In view of the existence of experimental data at 50 kev[20], it seemed reasonable to look at this energy for an indication that the program was behaving properly.

Figure 3 presents the differential cross section for a 50 kev electron, incident on an aluminum target, radiating a 25 kev photon (unpolarized). The solid curve is the present computation. The dashed curve is the Born approximation result (the Sauter differential cross section[21] corresponding to the integrated Bethe-Heitler result[1] The experimental points of Motz and Placious[20] are superimposed; they have an estimated accuracy of 10%. Figure 4 presents the corresponding results with a 50 kev electron but a 40 kev photon. The computer output tends to fall above the experimental points, though the discrepancy could be compatible with the experimental error. The Born approximation falls below the experimental values, significantly so for the higher energy photon (in fact, it goes to zero at the spectrum end-point while the experimental cross section does not).

At the time of writing of this report, no higher energy results were available.

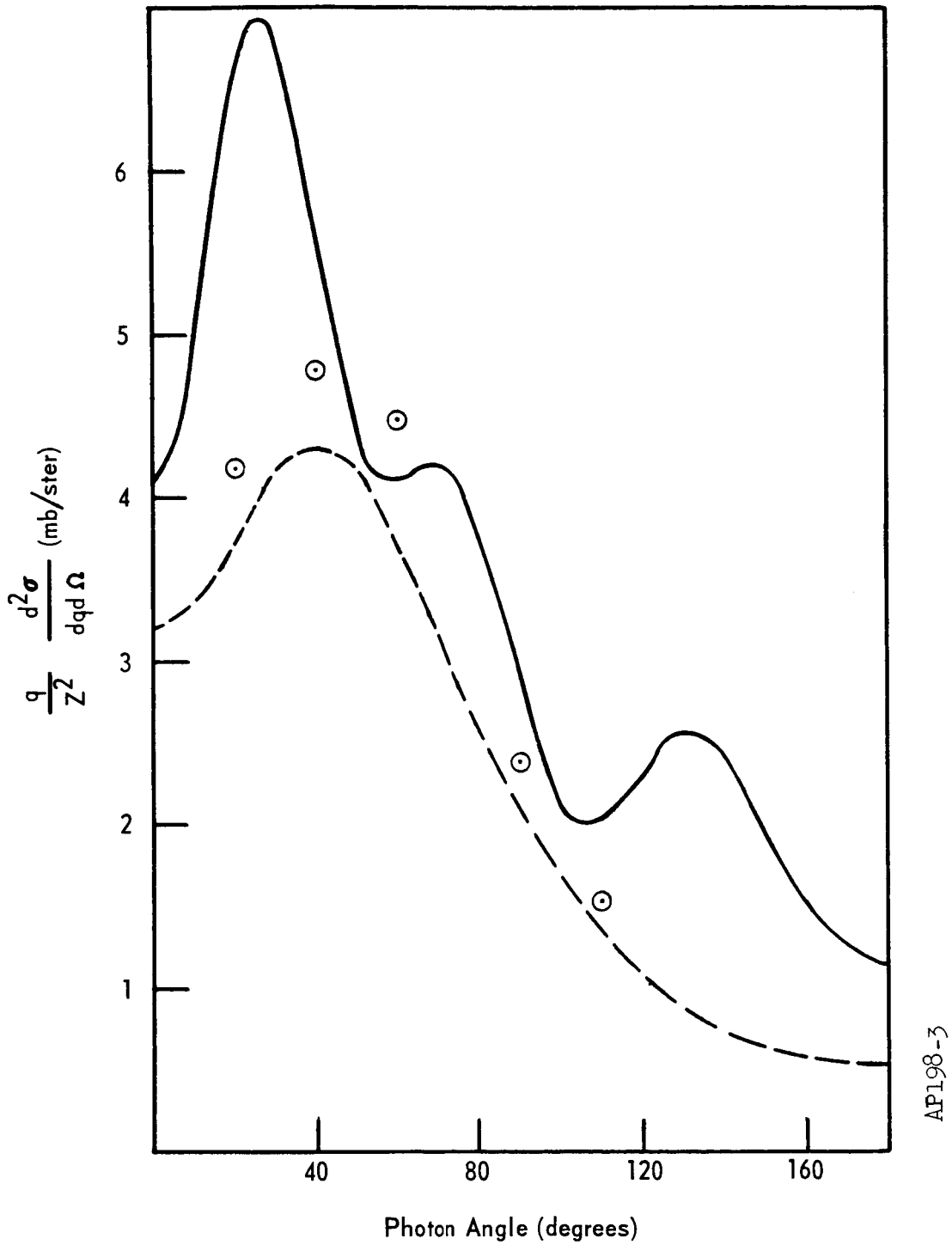

FIGURE 3 PHOTON ANGULAR DISTRIBUTION

The solid curve is the calculation for a 50 keV electron, incident on aluminum, radiating a 25 keV photon. The dashed curve is the corresponding Born approximation. The circles are experimental points from Motz and Placious.

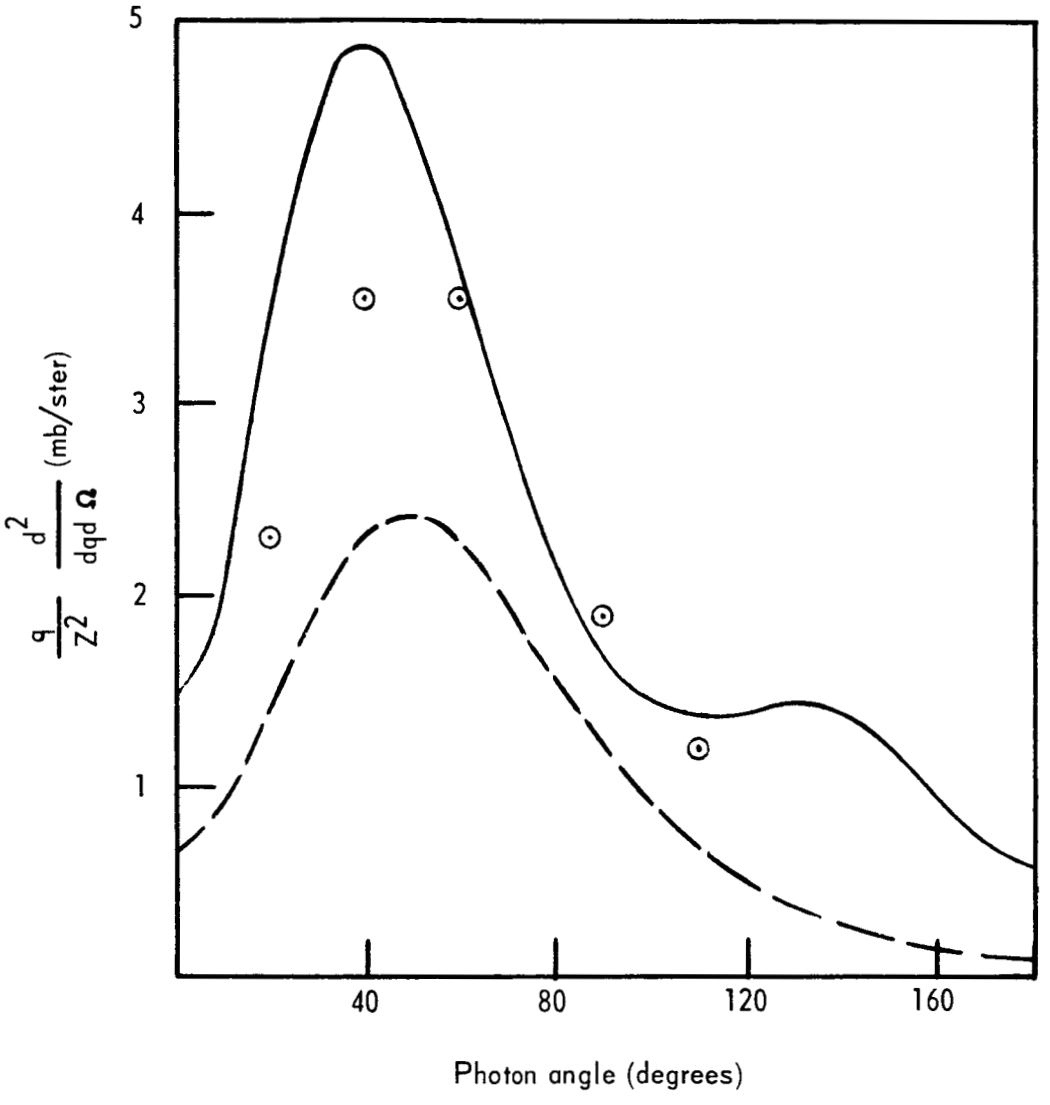

FIGURE 4 PHOTON ANGULAR DISTRIBUTION

The solid curve is the calculation for a 50 keV electron, incident on aluminum, radiating a 40 keV photon. The dashed curve is the corresponding Born approximation. The circles are experimental points from Motz and Placious.

APPENDIX

LISTING OF THE BREMSSTRALUNG PROGRAM

(FORTRAN IV, Huntsville Operating System)

```
$IBFTC BREMS   NOLIST,NODECK
C*******  MAIN PROGRAM BREMSSTRAHLUNG   FORTRAN IV  VERSION
      COMMON KEY, ZA, Q,
     1       M1,M2,NEW,B(13),GAM(44),CF(44),CG(44),EF(44),EG(44),NF,V
      COMMON / BESL / FL(13), PC(13), OF(13,13)
      COMMON /FAC/FACT(67),RTFAC(95),ROOT(50)
      COMMON /FUNCT/S(1500),DF(1500),F(1500),NEQ,X,H
      COMMON /INDEX/LBES(1500),KF(1500),KG(1500),NK,NFG,LVEC(11,22)
      COMMON /KAP/LMK(44),LPK(44),SI(44),CR(44),FKAP(44)
      COMMON /KUT/RK1(4),RK2(4),RK3(4),RK4(4)
      COMMON /MAXL/JM,LM,LM1,LM2,IEND,JFLAG,ITEST,IREP
      COMMON /SCF/NTAPE,NTOT,NSKIP,NCOUNT
      DIMENSION B1(23),B2(23),SIGA(23),SIGB(23),E(2),B1P(23,11)
      DIMENSION TERM(22),SIGBP(6),SIGAC(91)
      FL(1) = 1.0
      PC(1) = 1.0
      DO 54 L = 2, 13
      FL(L) = 2 * L - 1
      DO 53 J = 1, 13
      FLJ = J * ( 2 * ( L + J ) - 1 )
   53 OF(L,J) =  1.0 / FLJ
   54 PC(L) = PC(L-1) / FL(L)
      ROOT(1) = 1.0
      DO 55 I = 2, 50
      FAT = I
   55 ROOT(I) = SQRT (FAT)
      FACT(1) = 1.0
      FACT(3) = 1.0
      RTFAC(1) = 1.0
      RTFAC(3) = 1.0
      FAT = 1.0
      DO 56 I = 2, 33
      FI = I
      FAT = FAT * FI
      FACT(2*I+1) = FAT
   56 RTFAC(2*I+1) = SQRT(FAT)
      FAT = 1.0
      DO 57 I = 34, 47
      FI = I
      FAT = FAT * FI
   57 RTFAC(2*I+1) = SQRT(FAT) * RTFAC(67)
      SQ2 = 1.0 / ROOT(2)
      RK1 ( 1 ) = 0.5
      RK1 ( 2 ) = 1.0 - SQ2
      RK1 ( 3 ) = 1.0 + SQ2
      RK1 ( 4 ) = 1.0 / 6.0
      RK2 ( 1 ) = 2.0
      RK2 ( 2 ) = 1.0
      RK2 ( 3 ) = 1.0
      RK2 ( 4 ) = 2.0
      RK3 ( 1 ) = 0.5
      RK3 ( 2 ) = 1.0 - SQ2
      RK3 ( 3 ) = 1.0 + SQ2
      RK3 ( 4 ) = 0.5
      RK4 ( 1 ) = 0.5
      RK4 ( 2 ) = 0.0
      RK4 ( 3 ) = 0.5
      RK4 ( 4 ) = 0.0
      CCM = 0.5110062
      PI = 3.14159265
```

```
      OCTOPI = 8.0 * PI
      THOM = OCTOPI / 3.0
      RAD = PI/180.0
      ALFA = 1.0/137.0367
      REL = 0.281777
C     REL = CLASSICAL ELECTRON RADIUS IN CM ** (-12)
      FIB = 500.0 * PI * ALFA * REL * REL
C     UNITS ARE MILLIBARNS
      OSQ6 = 0.4/ROOT(6)
  100 READ (5,1)     KEY,JM,LM,LM1,LM2,NTAPE,NTEST,ITEST,INUM,IREP
    1 FORMAT (10I5)
      JFLAG = 0
      ITEST1 = 0
C****   KEY = 0  FOR PURE COULOMB
C****   KEY = 1  SCREENED CASE
C****   KEY = -1  END OF RUN
C*******  JM IS THE HIGHEST LEGENDRE COEFF. (J) IN ANG. DIST.
C*******  LM IS THE HIGHEST ORBITAL ANG. MOM. COEFF. (L) FOR PHOTON
C*******  LM1 IS THE HIGHEST ORBITAL ANG. MOM. COEFF. FOR INCIDENT ELEC.
C*******  LM2 IS THE HIGHEST ORBITAL ANG. MOM. COEFF. FOR SCATTERED ELEC
C*******  NTEST = 1 FOR MATRIX REDUCTION PROCEDURE, = 0 OTHERWISE
C*******  ITEST = 1 FOR RADIUS INCREMENT @ROCEDURE, = 0 OTHERWISE
C*******  INUM IS THE MAXIMUM NUMBER OF TIMES RADIUS WILL BE INCREMENTED
C*******  IREP = 1 TO RESTART RUN INTERRUPTED DURING SUMMING
C*******  IREP = 2 RESTARTS RUN FROM INTEGRATION WITHOUT FIRST SUMMING
C*******  IREP = 0 OTHERWISE
      IF (KEY.LT.0) CALL EXIT
      READ ( 5, 2 )  Z, E0, X0, RATIO, ZEL, RCUT, TOL, DRCUT
    2 FORMAT (8F10.0)
C*******  Z       -  NUCLEAR CHARGE
C*******  E0      -  K.E. OF INCIDENT ELECTRON IN MEV
C*******  X0      -  RATIO OF PHOTON ENERGY TO E0
C*******  RATIO   -  PARTICLE MASS TO ELECTRON MASS
C*******  ZEL     -  RATIO OF PART. CHARGE TO ELECTRON CHARGE
C*******  RCUT    -  RADIUS TO END NUMERICAL INTEGRATION
C*******  TOL IS ADMISSIBLE FRACTIONAL CHANGE OF SPECIFIED OUTPUT QUANTI
C*******  DRCUT IS RADIUS INCREMENT
      WRITE (6,4)
    4 FORMAT (1H1//2X41HINPUT DATA FOR BREMSSTRAHLUNG CALCULATION )
      IF (KEY.EQ.0) GO TO 102
  101 WRITE (6,5)
    5 FORMAT(1H0,2X,31HTHIS PROBLEM INCLUDES SCREENING/)
      GO TO 103
  102 WRITE (6,6)
    6 FORMAT(1H0,2X,39HTHIS PROBLEM DOES NOT INCLUDE SCREENING/)
  103 Q = E0 * X0
      WRITE ( 6, 7 ) Z,E0,X0,Q,RATIO,ZEL,RCUT,JM,LM,LM1,LM2
    7 FORMAT(1H0,2X,F9.4,4X,17H=  NUCLEAR CHARGE,//,
     1     3X,F9.4,4X,42H=  INCIDENT PARTICLE KINETIC ENERGY IN MEV,//,
     2     3X,F9.4,4X,42H=  RATIO OF PHOTON ENERGY TO INCIDENT K.E.//,
     3     3X,F9.4,4X,22H= PHOTON ENERGY IN MEV ////,
     4          13X,F9.4,4X,30H=  RATIO MASS TO ELECTRON MASS,//,
     5          13X,F9.4,4X,36H=  RATIO PARTICLE TO ELECTRON CHARGE,//,
     6          13X,F9.4,4X,29H=  INTEGRATION CUT-OFF RADIUS,////,
     7     22X,I5,10H  =  MAX J,//,22X,I5,21H  =  MAX L FOR PHOTON,//,
     8          22X,I5,32H  =  MAX L FOR INCIDENT PARTICLE,//,
     9          22X,I5,31H  =  MAX L FOR EXITING PARTICLE,//)
C****     DATA CHECKING LOGIC
      IF (JM.GT.(2*LM+2))     GO TO 1104
      MIL=MIN0(JM,LM,LM1,LM2)
```

```
      IF(MIL.LT.0)   GO TO 1104
      MAL=MAX0(LM,LM1,LM2)
      IF (MAL.GT.10)    GO TO 1104
      IF (      ( LM1 + LM2 ) .LT. LM )          GO TO 1104
      IF ( IABS ( LM1 - LM2 ) .GT. LM )          GO TO 1104
      GO TO 105
 1104 WRITE(6,19)
   19 FORMAT(1H0,42HANGULAR MOMENTUM INPUT VIOLATES CONDITIONS)
      GO TO 100
  105 IF (E0.LE.0.) GO TO 107
      IF (X0.LT.0.0.OR.X0.GT.1.0)  GO TO 110
      IF (X0.EQ.0.0) GO TO 109
      IF (Z.LT.0.0)     GO TO 1001
      IF (Z.EQ.0.0)   Z=0.1E-08
C**** AVERTS NUMERICAL DIFFICULTIES AT OR TOO NEAR TO Z = 0
      IF (ZEL.EQ.0.0)   GO TO 108
      IF (X0.LE.0.99)  GO TO 106
      IF (ZEL.LE.0.0) GO TO 108
      IEND = 1
      KEY = 0
      GO TO 104
  107 WRITE(6,15)
   15 FORMAT(1H0,2X,25H INPUT ENERGY IS NEGATIVE)
      GO TO 100
  110 WRITE(6,16)
   16 FORMAT(1H0,2X,39HFRACTION OF ENERGY TO PHOTON IMPOSSIBLE  )
      GO TO 100
  108 WRITE (6,17)
   17 FORMAT(1H0,2X,35HCROSS SECTION IS ZERO FOR THIS CASE)
      GO TO 100
  109 WRITE (6,18)
   18 FORMAT(1H0,2X,27HNO SCATTERING FOR THIS CASE)
      GO TO 100
 1001 WRITE (6,1002)
 1002 FORMAT (1H0,2X,32HATOMIC NUMBER CANNOT BE NEGATIVE )
      GO TO 100
  106 IEND = 2
  104 CALL SELECT
      IF ((NTEST+ITEST).LT.1)     GO TO 349
      WRITE (6,497)
  497 FORMAT (///50X9HOPTION(S)/)
      REWIND 8
      IF (ITEST.LT.1)     GO TO 351
      WRITE (6,352) DRCUT, INUM,TOL
  352 FORMAT (/   42X26HRADIUS INCREMENT PROCEDURE/
     135X36HCUT-OFF RADIUS WILL BE INCREASED BY F4.1, 1X5HUNITS /
     235X17HFOR A MAXIMUM OF I2,1X 5HTIMES/
     335X32HADMISSIBLE FRACTIONAL CHANGE IS F5.3//)
  351 IF (NTEST.EQ.0)     GO TO 349
      WRITE (6,353) TOL
  353 FORMAT (/ 42X26HMATRIX REDUCTION PROCEDURE/
     135X32HADMISSIBLE FRACTIONAL CHANGE IS F5.3//)
  349 NF = 2*(LM1+LM2+2)
      NFG = 2*NF
      NEQ = NK+NFG
      WRITE(6,1105) NK,NEQ
 1105 FORMAT (1H1//55X11HOUTPUT DATA////5X25HNUMBER OF MATRIX ELEMENTS/
     16X5HNK = I4,5X6HNEQ = I4///)
      IF (NEQ.LE.1500)  GO TO 1107
      WRITE (6,1106)
```

```
1106 FORMAT(1H0,2X,34HMATRIX ELEMENT DIMENSIONS EXCEEDED)
     GO TO 100
1107 IF(KEY.EQ.1)  CALL SETAPE(Z)
     ZA = ALFA * ZEL * Z
     T = E0/(RATIO*CCM)
     E(1) = T + 1.0
     Q = T*X0
     E(2) = E(1) - Q
     IF (IREP.GE.1)      RCUT=DRCUT
 223 CALL FINTEG (E,RCUT)
 832 CALL BSUM (B1,B2,B1P)
     TER = FI3 * X0 * Q / ( ZA * ( E(1)+1.0 ) * ZA )
     JMX = JM + 1
     DO 1108 I=1,JMX
     SIGA(I) = B1 (I)*TER
     SIGB(I) = B2 (I)*TER
1108 TER=-TER
     IF (ITEST.GE.1)      GO TO 111
     IF (NTEST.EQ.7777)      GO TO 802
 111 WRITE(6,8)
   8 FORMAT (///6X38HLEGENDRE COEFFICIENTS OF CROSS SECTION/
    1        / 6X,1HJ,9X,4HSIGA,13X,4HSIGB//)
     WRITE(6,9)                SIGA(1)
   9 FORMAT(5X,2H 0,5X,E12.5 /)
     IF (JM.EQ.0)      GO TO 354
     WRITE(6,10)                SIGA(2)
  10 FORMAT (5X,2H 1,5X,E12.5/)
     IF (JM.EQ.1)      GO TO 354
     DO 11 J=2,JM
     JP=J+1
     WRITE(6,12)                J,SIGA(JP),SIGB(JP)
  12 FORMAT(2X,I5,5X,E12.5,5X,E12.5 /)
  11 CONTINUE
 354 IF (ITEST.GE.1)      GO TO 802
     WRITE(6,13)
  13 FORMAT(1H1,30X,11HUNPOLARIZED,5X,75HTHIS IS (Q/Z**2) * D(SIGMA) /
    1( D(Q) * D(OMEGA) ) IN MILLIBARNS / STERADIAN / 5X,5HTHETA,7X,9HCO
    2S THETA / 30X,13HCROSS SECTION,4X,78HWHERE Q=PHOTON ENERGY, Z=ATOM
    3IC NUMBER, SIGMA=CROSS SECTION, OMEGA=SOLID ANGLE // )
 802 ITHETA = 0
     FMUQ=1.0
     KMUQ=1
 200 PMI=1.0
     SIGAC(KMUQ)=SIGA(1)+FMUQ*SIGA(2)
     PN=FMUQ
     J=2
 210 IF(J-JM)21,21,220
  21 FN=J-1
     PPL=(PN*FMUQ*(2.0*FN+1.0)/(FN + 1.0))-(PMI*FN)/(FN+1.0)
     PMI=PN
     PN=PPL
     SIGAC(KMUQ)=SIGAC(KMUQ)+(PPL* SIGA(J+1))
     TM=FN*(FN+1.0)*(FN+2.0)*(FN+3.0)
     TERM(J)= 1.0/SQRT  (TM)
     J=J+1
     GO TO 210
 220 IF (ITEST.NE.0)      GO TO 221
     IF (NTEST.EQ.9999)      GO TO 804
     IF ((NTEST.EQ.8888).OR.(NTEST.EQ.7777))      GO TO 814
 221 SIP = 2.0*SIGAC(KMUQ)
```

```
      GO TO 830
  804 SA = 0.0
      GO TO 816
  814 SA = ABS(SIGA(1)-SIG)/SIG
  816 S1 = ABS(SIGAC(1)-SI1)/SI1
      S46 = ABS(SIGAC(46)-SI46)/SI46
      S91 = ABS(SIGAC(91)-SI91)/SI91
      IF (ITEST.GT.1)      GO TO 751
      PS91 = 0.5*S91
      TEST = AMAX1 (SA,S1,S46,PS91)
      IF (NTEST.EQ.9999)      GO TO 880
      GO TO 834
  751 TEST = AMAX1 (SA,S1,S46,S91)
      WRITE (6,608) JM,LM,LM2,SIG,SIGA(1),SA,SI1,SIGAC(1),S1,SI46,
     1SIGAC(46),S46,SI91,SIGAC(91),S91,UO,U,SA
  608 FORMAT (///35X5HJM = I2,10X5HLM = I2,10X6HLM2 = I2//
     122X14HPREVIOUS VALUE,21X9HNEW VALUE,20X10HFRACT DIFF //
     25X7HSIGA(1),12XE12.5,18XE12.5,18XE12.5/
     35X8HSIGAC(1),11XE12.5,18XE12.5,18XE12.5/
     45X9HSIGAC(46),10XE12.5,18XE12.5,18XE12.5/
     55X9HSIGAC(91),10XE12.5,18XE12.5,18XE12.5/
     65X15HUNPOLARIZED SIG,4XE12.5,18XE12.5,18XE12.5//)
      IF (TEST.GT.TOL)      GO TO 548
      WRITE (6,952)
  952 FORMAT (/// 35X54HFRACTIONAL CHANGE OF MATRIX ELEMENTS WITHIN TOLE
     1RANCE.///)
      ITEST1 = ITEST1+1
      IF (ITEST1.LE.1)      GO TO 548
      ITEST = 0
      GO TO 354
  880 WRITE (6,806) JM,LM,LM2,SI1,SIGAC(1),S1,SI46,SIGAC(46),S46,
     1SIGAC(46),S46,SI91,SIGAC(91),S91,UO,U,SA
  806 FORMAT (///35X5HJM = I2,10X5HLM = I2,10X6HLM2 = I2//
     123X13HINITIAL VALUE,  22X9HNEW VALUE,20X10HFRACT DIFF //
     25X8HSIGAC(1),11XE12.5,18XE12.5,18XE12.5/
     35X9HSIGAC(46),10XE12.5,18XE12.5,18XE12.5/
     45X9HSIGAC(91),10XE12.5,18XE12.5,18XE12.5/
     55X15HUNPOLARIZED SIG,4XE12.5,18XE12.5,18XE12.5//)
      IF (TEST.GT.TOL)  GO TO 808
      IF (JM.LE.0)      GO TO 850
      JM = JM-1
      GO TO 802
  808 JM = JM+1
  850 WRITE (6,810) JM
  810 FORMAT (25X34HMINIMUM JM SATISFYING TOLERANCE = I2////
     125X37HPROCEDURE CONTINUING, DECREMENTING LM ///)
  818 IF ( LM .EQ. 0 )                    GO TO 819
      LM = LM-1
      JMX = JM+1
      IF (TER.LT.0.)     TER=-TER
      DO 812 I=1,JMX
      B1(I) = B1(I)-B1P(I,LM)
      SIGA(I) = B1(I)*TER
  812 TER = -TER
      NTEST = 8888
      GO TO 802
  834 LOX = LM2-JFLAG
      U = OCTOPI * SIGA(1)
  824 WRITE (6,826) JM,LM,LOX,SIG,SIGA(1),SA,SI1,SIGAC(1),S1,SI46,
     1SIGAC(46),S46,SI91,SIGAC(91),S91,UO,U,SA
```

```
826 FORMAT (///35X5HJM = I2,10X5HLM = I2,10X6HLM2 = I2//
   123X13HINITIAL VALUE,  22X9HNEW VALUE,20X10HFRACT DIFF //
   25X7HSIGA(1),12XE12.5,18XE12.5,18XE12.5/
   35X8HSIGAC(1),11XE12.5,18XE12.5,18XE12.5/
   45X9HSIGAC(46),10XE12.5,18XE12.5,18XE12.5/
   55X9HSIGAC(91),10XE12.5,18XE12.5,18XE12.5/
   65X15HUNPOLARIZED SIG,4XE12.5,18XE12.5,18XE12.5//)
    IF (NTEST.EQ.7777)     GO TO 846
    IF (TEST.LE.TOL)    GO TO 818
    LM = LM+1
819 WRITE (6,820) LM
820 FORMAT (25X34HMINIMUM LM SATISFYING TOLERANCE = I2//)
    NTEST = 7777
    JFLAG = 1
    WRITE (6,844)
844 FORMAT (//20X48HCONTINUING REDUCTION PROCEDURE, DECREMENTING LM2/)
    GO TO 832
846 IF (TEST.GT.TOL)      GO TO 838
    JFLAG = JFLAG+1
    IF (JFLAG-LM2)832,100,100
838 LM2 = LM2-JFLAG+1
    WRITE (6,848) LM2
848 FORMAT (//20X35HMINIMUM LM2 SATISFYING TOLERANCE = I1//////
   145X29HREDUCTION PROCEDURE COMPLETED )
    GO TO 100
830 IF (ITEST.GE.1)      GO TO 798
    WRITE (6,22)      ITHETA,FMUQ,SIP
 22 FORMAT ( I10, 5X, F9.5, 7X, E12.5 )
798 ITHETA = ITHETA+2
    THE = RAD*FLOAT(ITHETA)
    FMUQ=COS (THE)
    KMUQ=KMUQ + 1
    IF (ITHETA - 180 )200,200,300
300 IF (ITEST.GE.1)      GO TO 797
    WRITE (6,30)
 30 FORMAT (/ 55X,23HPOLARIZED CROSS SECTION //
   163X,7HPHI DEG//
   26X,5HTHETA,5X,9HCOS THETA,12X,2H 0,13X,2H30,13X,2H60,13X,2H90,
   311X,4H 120,11X,4H 150//)
797 ITHETA = 0
    FMUQ=1.0
    KMUQ=1
310 PMI=3.0*(1.0-(FMUQ*FMUQ))
    PN = 5.0*FMUQ*PMI
    SIGBC=TERM(2)*SIGB(3)*PMI+TERM(3)*SIGB(4)*PN
    J=4
311 IF(J-JM)320,320,331
320 FN=J-1
    PPL=((PN*FMUQ*(2.0*FN+1.0))/(FN-1.0))-PMI*(FN+2.0)/(FN-1.0)
    PMI=PN
    PN=PPL
    SIGBC=SIGBC+PPL*SIGB(J+1)*TERM(J)
    J=J+1
    GO TO 311
331 SIGBP(1)=SIGAC(KMUQ)+SIGBC
    SIGBP(2)=SIGAC(KMUQ)+0.5*SIGBC
    SIGBP(3)=SIGAC(KMUQ)-0.5*SIGBC
    SIGBP(4)=SIGAC(KMUQ)-1.0*SIGBC
    SIGBP ( 5 ) = SIGBP ( 3 )
    SIGBP(6)=SIGBP(2)
```

```
330 IF (ITEST.GE.1)      GO TO 796
    WRITE (6,31)      ITHETA,FMUQ,(SIGBP(I),I=1,6)
 31 FORMAT ( I10, 5X, F9.5, 7X, 6 ( E12.5, 3X ) )
796 ITHETA = ITHETA+2
    THE=RAD*FLOAT(ITHETA)
    FMUQ=COS (THE)
    KMUQ=KMUQ+1
    IF( ITHETA - 180 )      310,310,400
400 U = OCTOPI * SIGA(1)
    AVMU = THOM * SIGA(2) /  U
    AVMS = THOM * ( SIGA(1) + 1.2 * SIGA(3) ) / U
700 IF (ITEST.GE.1)      GO TO 552
    WRITE (6,71)  U,AVMU,AVMS
 71 FORMAT (///////7X60HUNPOLARIZED CROSS SECTION INTEGRATED OVER THE
   1SOLID ANGLE = E12.5,11H MILLIBARNS//50X17HAVERAGE COSINE = E12.5//
   242X25HAVERAGE COSINE SQUARED = E12.5//////)
551 IF (NTEST.EQ.0)      GO TO 100
    SIG = SIGA(1)
    SI1 = SIGAC(1)
    SI46 = SIGAC(46)
    SI91 = SIGAC(91)
    UO = U
    NTEST = 9999
    IF (JM.EQ.0)      GO TO 861
    JM = JM-1
    WRITE (6,800)
800 FORMAT (//50X35HSTARTING MATRIX REDUCTION PROCEDURE /
   150X15HDECREMENTING J //)
    GO TO 802
552 IF (ITEST.GT.1)      GO TO 814
548 SIG = SIGA(1)
    SI1 = SIGAC(1)
    SI46 = SIGAC(46)
    SI91 = SIGAC(91)
    UO = U
    IF (ITEST.GT.INUM)      GO TO 228
    RCUT = DRCUT
    XR = FLOAT(ITEST)*RCUT
    WRITE (6,951) XR
951 FORMAT (/////35X31HINCREMENTING INITIAL RADIUS BY F5.1 /////)
    ITEST = ITEST+1
    GO TO 223
228 WRITE (6,953)
953 FORMAT (///30X45HRADIUS INCREMENT PROCEDURE TERMINATED WITHOUT /
   130X44H OBTAINING DESIRED MATRIX ELEMENT STABILITY.  ///)
    ITEST = 0
    GO TO 354
861 WRITE (6,799)
799 FORMAT (///50X35HSTARTING MATRIX REDUCTION PROCEDURE ///)
    GO TO 850
    END
```

```
SIBFTC ASUMP   NOLIST,NODECK
C      SUM OVER EPSILON AND LAMBDAS
       SUBROUTINE ASUM(L,L1,A1,A2)
       COMMON /MAXL/JM,LM,LM1,LM2,IEND,JFLAG,ITEST,IREP
       DIMENSION A1(23), A2(23), PH(3, 3, 12)
       JMX=JM+1
       DO 1 J=1,JMX
       A1(J)=0.
     1 A2(J)=0.
       LMOD = MOD(L+L1,2)
       IF ((JM.EQ.0).AND.(LMOD.NE.0))  GO TO 9
       NA = 1
       IF(L.LT.2)     NA = 3-L
       NB = 1
       IF(L1.LT.2)  NB=3-L1
       CALL PHISUM(L,L1,PH)
       LL = 2 * L
       LL1 = 2 * L1
       IF (LMOD) 5,2,5
     2 JA=1
       IF(MOD(JM,2)) 4,3,4
     3 JB=JMX
       GO TO 6
     4 JB=JM
       GO TO 6
     5 JA=2
       IF(MOD(JM,2)) 3,4,3
     6 DO 8 NLA=NA,3
       DO 8 NLB=NB,3
       LA = L - 2 + NLA
       LA1 = L1 - 2 + NLb
       NEM = MIN0 ( LA, LA1 ) + 1
       LA = 2 * LA
       LA1 = 2 * LA1
        DO 8 J=JA,JB,2
       JJ=2*(J-1)
       IF((LA+LA1).LT.JJ)     GO TO 8
       IF(IABS(LA-LA1).GT.JJ)      GO TO 8
       IF ( MOD ( LA + LA1 + JJ, 4 ) .EQ. 0 )           GO TO 15
       VIJ = 0.0
       GO TO 16
    15 CALL CCOEFS(LA,LA1,JJ,0,0,C)
       VIJ=-PH(NLA,NLB,1)*C
    16 F=2.0
       DO 7 NE=2,NEM
       LE=2*(NE-1)
       CALL CCOEFS(LA,LA1,JJ,LE,-LE,C)
       VIJ=VIJ+F*C*PH(NLA,NLB,NE)
     7 F=-F
       CALL CCOEFS(LA1,LA,JJ,-2,2,CJ)
       CALL CCOEFS ( LL, 2,  LA, 0, 2, C2 )
       CALL CCOEFS ( LL1, 2, LA1, 0, 2, C2P )
       A1(J)=A1(J)+VIJ*CJ*C2*C2P
       IF(J.LT.3)     GO TO 8
       CALL CCOEFS(LA1,LA,JJ,-2,-2,CJ)
       IF(NLA.NE.2)    C2=-C2
       A2(J)=A2(J)+VIJ*CJ*C2*C2P
     8 CONTINUE
     9 RETURN
       END
```

```
$IBFTC ASYMPP  NOLIST,NODECK
       SUBROUTINE ASYMP (E)
C      ASYMPTOTIC EVALUATION OF TAIL OF MATRIX ELEMENTS
       COMMON KEY, ZA, Q,
      1        M1,M2,NEW,B(13),GAM(44),CF(44),CG(44),EF(44),EG(44),NF,V
       COMMON /FUNCT/S(1500),DF(1500),F(1500),NEQ,X,H
       COMMON /INDEX/LBES(1500),KF(1500),KG(1500),NK,NFG,LVEC(11,22)
       COMMON /MAXL/JM,LM,LM1,LM2,IEND,JFLAG,ITEST,IREP
       DIMENSION E(2),ENP(2),EWP(2),COF(5),COG(9,2)
       M1=LM+1
       M2 = M1 + 1
       QR=Q*X
       CALL BESSEL(QR)
       OTQ=0.5/Q
       OQR=OTQ/QR
       EP=E(1)*E(2)
       EPM=EP-1.0
       TEP=2.0*EP
       ENP(1)=E(1)-1.0
       ENP(2)=-E(1)-1.0
       EWP(1)=E(2)+1.0
       EWP(2)=-E(2)+1.0
       DO 1 IP=1,2
       COG(5,IP)=OTQ*EWP(IP)
       COG(6,IP)=-OTQ*ENP(IP)
       COG(1,IP)=V*COG(5,IP)
       COG(2,IP)=-V*COG(6,IP)
       COG(3,IP)=OQR*ENP(IP)*EWP(IP)
       COG(4,IP)=-TEP*COG(3,IP)
       COG(7,IP)=ENP(IP)*COG(5,IP)
       COG(8,IP)=OQR*EWP(IP)*(E(1)*(TEP-1.0)-E(2))
       COG(9,IP)=OQR*ENP(IP)*(E(2)*(TEP-1.0)-E(1))
1      CONTINUE
       COF(1)=OTQ*E(2)
       COF(2)=-OTQ*E(1)
       COF(3)=OQR*(TEP*EPM-Q*Q)
       COF(4)=-OQR*EPM
       COF(5)=OTQ*EPM-COG(1,1)-COG(2,1)
       DO 5 N=1,NK
       M=N+NFG
       LB=LBES(N)
       LBP=LB+1
       FLP = LBP
       JF=KF(N)
       JG=KG(N)
       MF = JF + NF
       MG = JG + NF
       IF(JF - JG)2,2,3
     2 IP = 1
       FA =F(JF)
       FB = F(MF)
       FAP = F(JG)
       FBP = F(MG)
       DFA = DF(JF)
       DFBP = DF(MG)
       GO TO 4
     3 IP =2
       FA = F(MG)
       FB = F(JG)
       FAP = F(MF)
```

```
      FBP = F(JF)
      DFA = DF(MG)
      DFBP = DF(JF)
    4 TB=COF(1)*FBP*DFA+COF(2)*FA*DFBP+COG(1,IP)*FA*FAP+COG(2,IP)*FB*FBP
      TC=(COF(3)+COF(4)*FLP)*FA*FBP+(COG(3,IP)*FLP+COG(4,IP))*FB*FAP
      TD = COG(5,IP)*FAP*DFA+COG(6,IP)*FB*DFBP+COG(7,IP)*FAP*FB
      TE= COF(5)*FA*FBP+COG(8,IP)*FA*FAP+COG(9,IP)*FB*FBP
      F(M)    = F(M) + (TB+TC) * B(LB) + (TD+TE) * B(LBP)
    5 CONTINUE
      RETURN
      END
```

```
$IBFTC BESSE   NOLIST,NODECK,M94
      SUBROUTINE BESSEL ( R )
C     SPHERICAL BESSEL FUNCTION
      COMMON KEY, ZA, Q, M1, M2, NEW, B(13)
      COMMON / BESL / FL(13), PC(13), OF(13,13)
      OR = 1.0 / R
      B(1) = SIN(R) * OR
      IF ( M1 .EQ. 1 )                                  GO TO 16
      NR = R + 2.0
      IF ( NR .LT. 2 )                                  GO TO 12
      B(2) = ( B(1) - COS(R) ) * OR
      IF ( M1 .EQ. 2 )                                  GO TO 16
      IF ( NR .GT. M1 )                                 NR = M1
      DO 11 L = 2, NR
   11 B(L+1) = FL(L) * B(L) * OR - B(L-1)
      IF ( NR - M1 )                                    13, 16, 16
   12 NR = 2
   13 HAS = 0.5 * R * R
      DO 15 L = NR, M2
      J = 0
      SER = 1.0
      TER = 1.0
   14 J = J + 1
      TER = - TER * HAS * OF(L,J)
      SER = SER + TER
      IF ( ABS(TER) .GT. ( 0.0001 * ABS(SER) ) ) GO TO 14
   15 B(L) = SER * PC(L) * ( R**(L-1) )
   16 RETURN
      END
```

```
$IBFTC BSUMP   NOLIST,NODECK
       SUBROUTINE BSUM (B1,B2,B1P)
C      SUM OVER L VALUES
       COMMON /MAXL/JM,LM,LM1,LM2,IEND,JFLAG,ITEST,IREP
       DIMENSION A1(23),A2(23),B1(23),B2(23),B1P(23,11)
       JMX=JM+1
       CALL ASUM (0,0,A1,A2)
       DO 1 J=1,JMX
       B1(J) = A1(J)
       B2(J) = A2(J)
       DO 1 L=1,LM
     1 B1P(J,L) = 0.0
       IF (LM.EQ.0)    GO TO 5
       DO 4 L=1,LM
       CALL ASUM (L,L,A1,A2)
       LL = 2*L+1
       FP = LL*LL
       DO 2 J=1,JMX
       B1P(J,L) = B1P(J,L) + FP*A1(J)
       B1(J) = B1(J) + B1P(J,L)
     2 B2(J)=B2(J)+FP*A2(J)
       DO 4 LP1=1,L
       L1=LP1-1
       CALL ASUM(L,L1,A1,A2)
       FP = 2*LL*(L1+LP1)
       DO 3 J=1,JMX
       B1P(J,L) = B1P(J,L) + FP*A1(J)
       B1(J) = B1(J) + B1P(J,L)
     3 B2(J)=B2(J)+FP*A2(J)
     4 CONTINUE
     5 RETURN
       END
```

```
$IBFTC CCOEF   NOLIST,NODECK
      SUBROUTINE CCOEFS ( J1, J2, J3, M1, M2, C )
      COMMON /FAC/F(67),RT(95),R(50)
      M3  =  M1 + M2
      C = 0.0
      SIGM = 1.0
      JMIN  =   MIN0 (J1,J2,J3)
      IF ( JMIN .GT. 4  )                                  GO TO 600
      IF ( JMIN .EQ. J2 )                                  GO TO 220
      IF ( JMIN .EQ. J3 )                                  GO TO 230
  210 L1 = J2
      L2  = J1
      L3  = J3
      LM1 = -M2
      LM2 = -M1
      LM3 = -M3
      GO  TO                    240
  220 L1 = J1
      L2  = J2
      L3  = J3
      LM1 = M1
      LM2 = M2
      LM3 = M3
      GO  TO                    240
  230 L1  = J1
      L2 = J3
      L3  = J2
      LM1 = M1
      LM2 = -M3
      LM3 = -M2
      SIGM = R(L2+1)/R(L3+1)
      IF ( MOD ( J1 - M1 , 4 ) .NE. 0 )                    SIGM = - SIGM
  240 IF ( LM2 ) 245, 250, 250
  245 LM1 = - LM1
      LM2 = - LM2
      LM3 = - LM3
      IF ( MOD ( L1 + L2 - L3 , 4 ) .NE. 0 )               SIGM = - SIGM
  250 JMIN = JMIN+1
      K = L1+LM3
      L = L1-LM3
      GO  TO         (255,260,300,700,400),   JMIN
  255 IF (L1-L3) 800,256,800
  256 IF (LM1-LM3) 800,257,800
  257 C  =  SIGM
      GO TO  800
  260 IF (L3-L1-LM2) 265,280,270
  265 SIGM = -SIGM
  270 K = L
  280 KP1 = K + 1
      C= R(KP1)/R(2*L1+2)
  290 C = SIGM*C
      GO  TO                    800
  300 IF (L3-L1) 305,310,315
  305 IF (LM2) 800,325,330
  310 IF (LM2) 800,340,345
  315 IF (LM2) 800,355,360
  325 C = -R(L)*R(K)/(R(2*L1)*R(L1+1))
      GO  TO                    380
  330 C = R(L)*R(L+2)/(2.0*R(L1)*R(L1+1))
      GO  TO                    380
```

```
340 C = FLOAT(LM3)/(R(L1)*R(L1+2))
    GO  TO                      380
345 LOX = L+2
    C = -R(K)*R(LOX)/(R(2*L1)*R(L1+2))
    GO  TO                      380
355 LOX = L+2
    LAX = K+2
    C = R(LOX)*R(LAX)/(R(2*L1+2)*R(L1+2))
    GO  TO                      380
360 C = R(K)*R(K+2)/(2.0*R(L1+1)*R(L1+2))
380 C = SIGM*C
    GO  TO                      800
400 M = LM2/2+1
    J = (L3-L1)/2 +3
    GO  TO                (480,510,540),        M
480 GO  TO       (485,490,495,500,505),         J
485 C = R(3)*R(L)*R(L-2)*R(K)*R(K-2)/(R(8)*R(L1-2)*R(L1-1)*R(L1)
   1*R(L1+1))
    GO  TO                      575
490 C = -0.5*FLOAT(LM3)*R(6)*R(L)*R(K)/(R(L1)*R(L1-2)*R(L1+1)*R(L1+2))
    GO  TO                      575
495 C = 0.5 *FLOAT(3*LM3*LM3-L1*(L1+2))
   1                                  /(R(L1)*R(L1-1)*R(L1+2)*R(L1+3))
    GO  TO                      575
500 LOX = L+2
    LAX = K+2
    C = 0.5*FLOAT(LM3)
   1              *R(6)*R(LOX)*R(LAX)/(R(L1)*R(L1+1)*R(L1+2)*R(L1+4))
    GO  TO                      575
505 LOX = L+4
    LAX = L+2
    LLX =K+4
    LXX = K+2
    C = R(3)*R(LOX)*R(LAX)*R(LLX)*R(LXX)/(R(8)*R(L1+1)*R(L1+2)*R(L1+3)
   1*R(L1+4))
    GO  TO                      575
510 GO  TO       (515,520,525,530,535),         J
515 C = -R(L+2)*R(L)*R(L-2)*R(K-2)/(2.0*R(L1-2)*R(L1-1)*R(L1)*R(L1+1))
    GO  TO                      575
520 C = 0.5*FLOAT(L1+2*LM3-2)
   1                     *R(L+2)*R(L)/(R(L1)*R(L1-2)*R(L1+1)*R(L1+2))
    GO  TO                      575
525 LOX = L+2
    C = (1.0-FLOAT(LM3))*R(3)*R(LOX)*R(K)/(R(2*L1)*R(L1-1)*R(L1+2)
   1*R(L1+3))
    GO  TO                      575
530 C=.5*FLOAT(2*LM3-L1-4)
   1                     *R(K+2)*R(K)/(R(L1)*R(L1+1)*R(L1+2)*R(L1+4))
    GO  TO                      575
535 LOX = L+4
    C=R(LOX)*R(K+4)*R(K+2)*R(K)/(2.0*R(L1+1)*R(L1+2)*R(L1+3)*R(L1+4))
    GO  TO                      575
540 GO  TO       (545,550,555,560,565) ,        J
545 C = R(L-2)*R(L)*R(L+2)*R(L+4)/(4.0*R(L1)*R(L1-2)*R(L1-1)*R(L1+1))
    GO  TO                      575
550 C= -R(K-2)*R(L)*R(L+2)*R(L+4)/(2.0*R(L1)*R(L1-2)*R(L1+2)*R(L1+1))
    GO  TO                      575
555 LOX = L+2
    LAX = L+4
    C = R(3)*R(K-2)*R(K)*R(LOX)*R(LAX)/(R(8)*R(L1)*R(L1-1)*R(L1+2)
```

```
     1*R(L1+3))
      GO  TO                    575
  560 LOX = L+4
      C = -R(K-2)*R(K)*R(K+2)*R(LOX)/(2.0*R(L1)*R(L1+2)*R(L1+4)*R(L1+1))
      GO  TO                    575
  565 C=R(K-2)*R(K)*R(K+2)*R(K+4)/(4.0*R(L1+1)*R(L1+2)*R(L1+3)*R(L1+4))
  575 C = SIGM*C
      GO  TO                    800
  700 M = (LM2+1)/2
      J = (L3-L1+5)/2
      GO TO ( 710, 740 ) ,   M
  710 GO TO ( 720, 725, 730, 735 ) ,   J
  720 C = R(3)*R(K-1)*R(L-1)*R(L+1)/(R(8)*R(L1)*R(L1-1)*R(L1+1))
      GO TO 780
  725 C = -FLOAT((L1+3*LM3-1)/2)*R(L+1)/(R(2)*R(L1-1)*R(L1+1)*R(L1+2))
      GO TO 780
  730 KP1 = K + 1
      C = -FLOAT((3-3*LM3+L1)/2)*R(KP1)/(R(2)*R(L1)*R(L1+1)*R(L1+3))
      GO TO 780
  735 LOX = L+3
      KP1 = K + 1
      KP3 = K + 3
      C = R(3)*R(KP1)*R(KP3)*R(LOX)/(R(8)*R(L1+1)*R(L1+2)*R(L1+3))
      GO TO 780
  740 GO TO ( 750, 755, 760, 765 ) ,   J
  750 C = -R(L-1)*R(L+1)*R(L+3)/(R(8)*R(L1)*R(L1-1)*R(L1+1))
      GO TO 780
  755 LOX = L+3
      LAX = L+1
      C = R(3)*R(K-1)*R(LAX)*R(LOX)/(R(8)*R(L1-1)*R(L1+1)*R(L1+2))
      GO TO 780
  760 LOX = L+3
      C = -R(3)*R(K-1)*R(K+1)*R(LOX)/(R(8)*R(L1)*R(L1+1)*R(L1+3))
      GO TO 780
  765 C = R(K-1)*R(K+1)*R(K+3)/(R(8)*R(L1+1)*R(L1+2)*R(L1+3))
  780 C = C*SIGM
      GO TO 800
C     THIS IS THE BEGINNING OF COMPUTATION OF C-COEFFICIENT
C     USING THE GENERAL EXPRESSION.
  600 L1 = J1+J2-J3+1
      L2 = J1-J2+J3+1
      L3 = -J1+J2+J3+1
      L10 = J1+J2+J3+3
      IF ( M3 .EQ. 0 )              GO TO 615
  605 L4 = J1+M1+1
      L5 = J1-M1+1
      L6 = J2+M2+1
      L7 = J2-M2+1
      L8 = J3+M3+1
      L9 = J3-M3+1
      ST = RT(L10) / (   RT(L1)*RT(L4)*RT(L5)*RT(L6)*RT(L7) )
      ST = ST / (  R(J3+1)*RT(L2)*RT(L3)*RT(L8)*RT(L9)  )
      N7 = L1-L7
      N4 = L1-L4
      MIN = MAX0 (0,N4,N7)
      MAX = MIN0 (L1,L5,L6)
      IF (MOD(MIN,4).NE.0)      SIGM=-1.0
      MIN = MIN+1
      N1 = L1+1
      N5 = L5+1
```

```
      N6 = L6+1
      SUM = 0.0
      DO 610 LZ=MIN,MAX,2
      N1L = N1-LZ
      N5L = N5-LZ
      N6L = N6-LZ
      N4L =-N4+LZ
      N7L =-N7+LZ
      TERM = ST * F(LZ) * F(N1L) * F(N5L) * F(N6L) * F(N4L) * F(N7L)
      C = C + SIGM / TERM
  610 SIGM = -SIGM
      GO TO 800
  615 IF ( M2 .EQ. 0 )              GO TO 620
      IF ( IABS(M2) .NE. 2 )        GO TO 605
      JMOD = MOD ( (L10 + 1 ), 4 )
      IF ( JMOD .NE. 0 )            GO TO 605
      ST = J3 * (J3 + 2) - J1 * (J1 + 2) - J2 * (J2 + 2)
      SIGM = 0.5 * SIGM * ST / ( R(J1)*R(J1 + 2)*R(J2)*R(J2 + 2) )
  620 JMOD = MOD (  ( L1 - 1 ), 8  )
      IF ( JMOD .NE. 0 )            SIGM = - SIGM
      L4 = ( L1  + 1 ) / 2
      L5 = ( L2  + 1 ) / 2
      L6 = ( L3  + 1 ) / 2
      L7 = ( L10 - 1 ) / 2
      C = SIGM * R(J3 + 1) * F(L7) / (  F(L4)*F(L5)*F(L6)  )
      C = C * RT(L1) * RT(L2) * RT(L3) /  RT(L10)
  800 RETURN
      END
```

```
$IBFTC CNORMP  NOLIST,NODECK
       SUBROUTINE CNORM ( RNORM )
       COMMON KEY, ZA, Q,
      1        M1,M2,NEW,B(13),GAM(44),CF(44),CG(44),EF(44),EG(44),NF,V
       COMMON /FUNCT/S(1500),DF(1500),F(1500),NEQ,X,H
       DIMENSION RNORM(44)
       PI = 3.14159265
       OVERR = 1.0/X
       OVRRSQ = OVERR * OVERR
       DO 1 I=1,NF
       IG = I+NF
       EMVP1 = EG(I)-V
       EMV = EMVP1-1.0
       Y = -V/EMVP1
       XSQ = Y*Y
       FK =  CF ( I )
       FKSQ = FK*(FK+1.0)
       FKR = FK * OVERR
       FOR = FKSQ * OVRRSQ
       IF (KEY)7,4,5
     4 EP = 0.
       GO TO 6
     5 RA = RATERP(X)
       RB = RATERP(X+1.0)
       EP = ALOG(RA/RB)
     6 V1 = EP + OVERR
       V2 = V1 * V1 + OVRRSQ
       V3 = V1 * V2 + 2.0 * OVRRSQ * (V1 + OVERR)
       V1X = V1*Y
       V2X = V2*Y
       VXSQ = V1X * V1X
       FKVR = V1X * FKR
       PSQ = EMV * EMV - 1.0 + FKVR - 0.75 * VXSQ + 0.5 * V2X - FOR
       PSQP = 2.0 * EMV * V * V1 + (FKR - 1.5*V1X) * VXSQ + 2.0* V1X* V2X
      1       - 0.5 * V3 * Y - OVERR * (FKVR + FK*V2X - 2.0*FOR)
       TERM3 = F(IG)*(0.5*V1X-FKR+0.25*(PSQP/PSQ))+F(I)*EMVP1
       A = (PI/(SQRT (PSQ)*EMVP1))*(PSQ*F(IG)*F(IG)+TERM3*TERM3)
       RNORM(I) = 1.0/SQRT (A)
     1 CONTINUE
     7 RETURN
       END
```

```
$IBFTC DIRECT  NOLIST,NODECK
      SUBROUTINE DIRAC
C                     COMPUTE DERIVATIVES
      COMMON KEY, ZA, Q,
     1       M1,M2,NEW,B(13),GAM(44),CF(44),CG(44),EF(44),EG(44),NF,V
      COMMON /FUNCT/S(1500),DF(1500),F(1500),NEQ,X,H
      COMMON /INDEX/LBES(1500),KF(1500),KG(1500),NK,NFG,LVEC(11,22)
      IF(X) 50,5,10
    5 NEW = 1
      GO TO 50
   10 IF  (NEW)                  30,30,11
   11 NEW  =  0
      V = -ZA/X
      IF (KEY.EQ.1)   V=RATERP(X)*V
      Z = Q*X
      CALL BESSEL (Z)
      GO TO 31
   30 NEW  =  1
   31 DO 35 N=1,NF
      NG = N+NF
      DF(N)  =   CF(N)*F(N)/X  - (EF(N)-V)*F(NG)
      DF(NG) = CG(N)*F(NG)/X + (EG(N)-V)*F(N)
   35 CONTINUE
      DO  40                     N=1,NK
      I  =  KF(N)
      M = KG(N)
      J = M + NF
      K = N + NFG
      L = LBES(N)
      DF(K) = B(L) * F(I) * F(J)
      IF ( X .GE. 1.0 )                  GO TO 40
      A =  GAM(I) + GAM(M)
      DF(K) = DF(K) * ( X**A )
   40 CONTINUE
   50 RETURN
      END
```

```
$IBFTC FINTE   NOLIST,NODECK
      SUBROUTINE FINTEG ( E, RCUT )
C                  NUMERICAL INTEGRATION OF DIRAC EQUATIONS
      COMMON KEY, ZA, Q,
     1       M1,M2,NEW,B(13),GAM(44),CF(44),CG(44),EF(44),EG(44),NF,V
      COMMON /FUNCT/S(1500),DF(1500),F(1500),NEQ,X,H
      COMMON /INDEX/LBES(1500),KF(1500),KG(1500),NK,NFG,LVEC(11,22)
      COMMON /KAP/LMK(44),LPK(44),SI(44),CR(44),FKAP(44)
      COMMON /MAXL/JM,LM,LM1,LM2,IEND,JFLAG,ITEST,IREP
      DATA  BOOL/0777400000000/
      DIMENSION E(2),RK(2),GNU(2),MIN(2),MAX(2),LML(2),AFT(2),AGT(2),
     1     PIK(2),SQE(2),SQEG(2),SN(44),HP(44),HM(44),XIM(44),RNORM(44)
  120 FORMAT (7F10.0)
  501 FORMAT ( 7 E 14.5 )
  502 FORMAT (25H1 NORMALIZATION FACTORS     )
      IF (ITEST.GE.2)      GO TO 401
      PI  =  3.14159265
      TWOPI = 2.0*PI
      HALFPI =  0.5 * PI
      SQ2 = SQRT (0.5)
      ZAZA=ZA*ZA
  140 MIN(1) = 1
      MAX(1) =  2*LM1 + 2
      MIN(2) =  MAX(1) + 1
      MAX(2) =  MAX(1) + 2*LM2 + 2
      M1  =  LM1 + 1
      M2  =  LM2 + 1
      DO  155                        N=1,M1
      K  =   N -M1 -1
      LMK(N)  =  -K
      LPK(N)  =  -K -1
      FKAP(N)  =   K
  155 SN(N)    =  -1.0
      M  =  M1
      DO  160                        N=1,M1
      M  =  M + 1
      LMK(M)  =   N -1
      LPK(M)  =   N
      FKAP(M) =  N
  160 SN(M)   =  1.0
      DO  165                        N=1,M2
      M  =   M + 1
      K  =   N -M2 -1
      LMK(M)  =  -K
      LPK(M)  =  -K -1
      FKAP(M)  =  K
  165 SN(M)    = -1.0
      DO  170                        N=1,M2
      M  =   M + 1
      LMK(M)  =   N -1
      LPK(M)  =   N
      FKAP(M)  =  N
  170 SN(M)    =  1.0
  200 DO 300 N=1,IEND
      N1 = MIN(N)
      N2 = MAX(N)
      EFN = E(N)-1.0
      EGN = E(N)+1.0
      RK(N) = SQRT (EFN*EGN)
      SQEG(N) = SQRT (EGN)
```

```
      SQE (N) = RK(N)/SQEG(N)
      GNU(N) = ZA*E(N)/RK(N)
      AG = ABS (GNU(N))
      G2=AG*AG
      HZ = G2/E(N)
      XNUPI2 = EXP (GNU(N)*HALFPI)
      PIK(N) = 1.0/SQRT (TWOPI*RK(N))
      AFT(N)=SQE(N)*PIK(N)
      AGT(N)=SQEG(N)*PIK(N)
      TWORK = XNUPI2*PIK(N)
      TRK = 2.0*RK(N)
      DO 300 I=N1,N2
      IG = I+NF
      GAM(I) = SQRT (FKAP(I)*FKAP(I)-ZAZA)
      EF(I) = EFN
      EG(I) = EGN
      TUGAM  = 2.0*GAM(I)+1.0
      CALL LOGGAM (GAM(I),GNU(N),XRE,XIM(I) )
      ZK = TWORK*(TRK**GAM(I))*EXP (XRE)
      IF (FKAP(I)) 264,264,268
  264 CF(I) = FKAP(I)-GAM(I)
      CG(I) = - ZAZA / CF(I)
      HM(I) = ZAZA * EFN / ( CF(I) * ( E(N) * FKAP(I) - GAM(I) ) )
      HP(I) = 2.0 - HM(I)
      HM(I) = SQRT ( HM(I) )
      HP(I) = SQRT ( HP(I) )
      F(IG) = ZK*SQEG(N)*(GAM(I)*HP(I)+AG*HM(I))
      F(I) = ZA*F(IG)/CF(I)
      GO TO 270
  268 CG(I) = -FKAP(I)-GAM(I)
      CF(I) = - ZAZA / CG(I)
      HP(I) = -ZAZA*EGN*( E(N)*FKAP(I)+GAM(I) ) / ( CG(I)*(EFN*EGN
     1          *FKAP(I)*FKAP(I)+ZAZA) )
      HM(I) = 2.0 - HP(I)
      HM(I) = SQRT ( HM(I) )
      HP(I) = SQRT ( HP(I) )
      F(I) = ZK*SQE (N)*(GAM(I)*HM(I)+AG*HP(I))
      F(IG) = -ZA*F(I)/CG(I)
  270 DEN = 1.0/TUGAM
      DF(IG) = F(I)*((1.0-CF(I))*EG(I)-CF(I)*EF(I))*DEN
      DF(I) = -F(IG)*((1.0-CG(I))*EF(I)-CG(I)*EG(I))*DEN
  300 CONTINUE
  301  IF(IEND-1) 302,302,309
  401 NDON = RCUT/H + 0.1
      K = 8
      IF ( IREP .GT. 0 )          K = 9
      REWIND K
      READ(K) X,(S(I),DF(I),F(I),I=1,NEQ),(LBES(I),KF(I),KG(I),I=1,NK)
      REWIND K
      IF (IREP.GE.1)     GO TO 362
      GO TO 402
  302 N1 = MIN(2)
      N2 = MAX(2)
      HAZ = SQRT (ZA)
      TUAZ = 2.0*ZA
      DO 306 I=N1,N2
      GAM(I) = SQRT (FKAP(I)*FKAP(I)-ZAZA)
      EF(I) = 0.0
      EG(I) = 2.0
      IG = I+NF
```

```
      TUGAM  = 2.0*GAM(I)+1.0
      F(I) = HAZ*(TUAZ**GAM(I))
      IF(FKAP(I))303,303,304
  303 F(I) = -F(I)
      CF(I) = FKAP(I)-GAM(I)
      CG(I) = - ZAZA / CF(I)
      AZKAP = CF(I)/ZA
      GO TO 305
  304 CG(I) = -FKAP(I)-GAM(I)
      CF(I) = - ZAZA / CG(I)
      AZKAP = -ZA/CG(I)
  305 F(IG) = AZKAP*F(I)
      DF(I) = -F(I)*TUAZ/TUGAM
      DF(IG) = 2.0*F(I)*(1.0-CF(I))/TUGAM
  306 CONTINUE
  309 X  =   0.0
      M1 = LM+1
      M2 = LM+2
      H1 = 0.0078125
      H2 = 0.0625 / RK(1)
      H2 = AND ( H2, BOOL )
      H  =   H1
      N = 1.0/H + 0.5
      WRITE (6,886) H1,H2
  886 FORMAT (//9X41HINTEGRATION INCREMENT UP TO X EQUAL 1 IS F8.6/
     14X46HINTEGRATION INCREMENT FOR X GREATER THAN 1 IS F8.6//)
      DO 310      L = 1, 13
  310 B(L) = 0.0
      DO 315 NO = 1, NFG
  315 S(NO) = 0.0
      DO 320 NN = 1, NK
      NNN = NN + NFG
       F(NNN) = 0.0
      DF(NNN) = 0.0
  320 S (NNN) = 0.0
      IF (IREP.GE.1)      GO TO 360
      DO  350                   I=1,N
      CALL RKUTTA
  350 CONTINUE
  360 H  =  H2
      DO  365                   I=1,NF
      CF(I)  =  FKAP(I)
      CG(I)  = -FKAP(I)
  365 CONTINUE
      IF (IREP.GE.1)      GO TO 366
      DO  406   I = 1, NF
      IG = I + NF
      TUGAM  = 2.0*GAM(I)+1.0
      CALL LOGGAM (TUGAM ,0.,XRE2,UG)
      RNORM(I) = EXP ( - XRE2 )
       S(I ) =  S(I ) * RNORM(I)
       S(IG) =  S(IG) * RNORM(I)
      DF(I ) = DF(I ) * RNORM(I)
      DF(IG) = DF(IG) * RNORM(I)
      F(I ) = F(I ) * RNORM(I)
  406 F(IG) = F(IG) * RNORM(I)
      DO 407 K=1,NK
      N = NFG+K
      I = KF(K)
      J = KG(K)
```

```
      S(N) =  S(N) * RNORM(I) * RNORM(J)
     DF(N) = DF(N) * RNORM(I) * RNORM(J)
  407 F(N) = F(N)*RNORM(I)*RNORM(J)
  366 NEW = 1
      CALL DIRAC
      IF (IREP.GE.1)      GO TO 401
      NDON = (RCUT-1.0)/H + 0.1
  402 DO 370 NDO = 1, NDON
  370 CALL RKUTTA
  403 REWIND 8
      WRITE(8)X,(S(I),DF(I),F(I),I=1,NEQ),(LBES(I),KF(I),KG(I),I=1,NK)
      REWIND 8
  404 IF (KEY) 430,430,410
  362 IREP = IREP-1
      V = - ZA / X
      IF ( KEY .EQ. 1 )            V = RATERP(X) * V
      IF (IREP.EQ.0)      GO TO 403
      IREP = 0
      GO TO 366
  430 DO 440 N=1,2
      N1 = MIN(N)
      N2 = MAX(N)
      DO 440 I=N1,N2
      COSETA = -SQ2*SN(I)*HP(I)
      SINETA = -SQ2*HM(I)
      XXL1 = LPK(I)+1
      ALPHA = -XIM(I) + HALFPI*(XXL1-GAM(I))
      CSALPH = COS (ALPHA)
      SNALPH = SIN (ALPHA)
      SI(I) = SINETA*CSALPH+COSETA*SNALPH
      CR(I) = COSETA*CSALPH-SINETA*SNALPH
  440 CONTINUE
      GO TO 518
  410 WRITE(6,502)
      WRITE ( 6, 120 ) X
      LML(1) = LM1
      LML(2) = LM2
      DO  420                    N=1,2
      M1 = LML(N)+1
      M2 = M1 + 1
      T    =    X*RK(N)
      CALL BESSEL ( T )
      AF = AFT(N)/SQ2
      AG = AGT(N)/SQ2
      N1   =    MIN(N)
      N2   =    MAX(N)
      DO  420                    I=N1,N2
      IG = I + NF
      L1   =    LPK(I) + 1
      L2   =    LMK(I) + 1
      FZ       =    SN(I)*AF*T*B(L2)
      GZ       =    AG*T*B(L1)
C                                          COMPUTE PHASE FACTORS.
      RN    =   FZ*FZ/EF(I) + GZ*GZ/EG(I)
      RD    =   F(1)*F(I)/EF(I)  +  F(IG)*F(IG)/EG(I)
      RNORM(I)    = SQRT (RN/RD)
      SI(I) =((FZ*F(IG)/RK(N) - GZ*F(I)/RK(N))/RN)*RNORM(I)
      CR(I) = ((FZ*F(I)/EF(I) +GZ*F(IG)/EG(I))/RN)*RNORM(I)
      WRITE (6,501) FZ,F(I),GZ,F(IG),RNORM(I),SI(I),CR(I)
  420 CONTINUE
```

```
C                                                  COMPUTE NORMALIZATION FACTORS.
  518 CALL CNORM ( RNORM )
      WRITE(6,502)
      DO 789 I=1,NF
      IG = I + NF
      DF(I ) = DF(I ) * RNORM(I)
      DF(IG) = DF(IG) * RNORM(I)
      F(I ) = F(I ) * RNORM(I)
      F(IG) = F(IG) * RNORM(I)
      WRITE(6,501)                RNORM(I)
  789 CONTINUE
      DO 400 K=1,NK
      N = NFG+K
      I = KF(K)
      J = KG(K)
  400 F(N) = F(N)*RNORM(I)*RNORM(J)
      NX = NFG+1
      WRITE (6,500)  X,(F(I),I=1 ,NEQ)
      CALL ASYMP (E)
      SUM = 0.0
      DO 585 I=NX,NEQ
  585 SUM = SUM+F(I)**2
      WRITE (6,500)  X,(F(I),I=NX,NEQ)
  500 FORMAT (///50X15HMATRIX ELEMENTS//54X4HX = F6.1//(10F11.5))
      WRITE (6,498)  SUM
  498 FORMAT (//32X26HSUM OF ELEMENTS SQUARED = E15.8////)
  380 CALL HFUN
      RETURN
      END
```

```
$IBFTC HFUNP   NOLIST,NODECK
       SUBROUTINE HFUN
C      REDUCED MATRIX ELEMENTS H
       COMMON /FUNCT/S(1500),DF(1500),F(1500),NEQ,X,H
       COMMON /INDEX/LBES(1500),KF(1500),KG(1500),NK,NFG,LVEC(11,22)
       COMMON /KAP/LMK(44),LPK(44),SI(44),CR(44),FKAP(44)
       DIMENSION  HF(1500,3)
       EQUIVALENCE (S(1),HF(1,1))
       DO 23 N=1,NK
       M = N + NFG
       IF(KF(N)-KG(N))  1,1,2
     1 MF = KF(N)
       K1 = -FKAP(MF)
       NG = KG(N)
       K2 = FKAP(NG)
       GO TO 3
     2 F(M) = - F(M)
       MF = KG(N)
       K1 = FKAP(MF)
       NG = KF(N)
       K2=-FKAP(NG)
       KF(N) = MF
       KG(N)=NG
     3 JK1=LPK(MF)+LMK(MF)
       JK2 = LPK(NG)+LMK(NG)
       L = LBES(N)-1
       IF ( L .EQ. 0 )                GO TO 8
       L2=2*L
       JPL=JK1+ JK2-L2
       JML=IABS(JK1-JK2)-L2
       IF(JPL-2) 9, 4, 4
     4 IF ( L .EQ. 1 )                GO TO 6
       IF(JML+2) 5,5,6
     5 LBES(N) =1
       NA = 1
       NB = 3
       GO TO 14
     6 IF(JML) 7,7,8
     7 LBES(N)=2
       NA = 2
       NB = 3
       GO TO 14
     8 LBES(N) =3
       NA = 3
       NB = 3
       GO TO 14
     9 IF ( L .EQ. 1 )                GO TO 13
       IF(JPL) 10,11,11
    10 LBES(N) = 4
       NA = 1
       NB = 1
       GO TO 14
    11 IF(JML+2) 12,12,13
    12 LBES(N) = 5
       NA = 1
       NB = 2
       GO TO 14
    13 LBES(N) = 6
       NB = 2
       NA = 2
```

```
14 DO 22 NLA = NA,NB
   LA = 2*(L+NLA-2)
   CALL XFUN(L,K1,K2,NLA,GEL)
   CALL CCOEFS(LA,JK2,JK1,0,1,C)
   HF(N,NLA) = F(M)*GEL*C
22 CONTINUE
23 CONTINUE
   RETURN
   END
```

```
$IBFTC LOGGA   NOLIST,NODECK
      SUBROUTINE LOGGAM(X,Y,U,V)
C THIS SUBROUTINE COMPUTES THE NATURAL LOG OF THE GAMMA FUNCTION FOR
C COMPLEX ARGUMENTS.  THE ROUTINE IS ENTERED BY THE STATEMENT
C      CALL LOGGAM(X,Y,U,V)
C WHERE    X IS THE REAL PART OF THE ARGUMENT
C          Y IS THE IMAGINARY PART OF THE ARGUMENT
C          U IS THE REAL PART OF THE RESULT
C          V IS THE IMAGINARY PART OF THE RESULT
      DIMENSION H(7)
      H(1)=2.269488974
      H(2)=1.517473649
      H(3)=1.011523068
      H(4)=.5256064690
      H(5)=.2523809524
      H(6)=0.0333333333
      H(7)=0.0833333333
      E2=1.57079632679
      E8=3.14159265359
      B1=0.0
      B2=0.0
      J=2
      X2=X
    4 IF(X)1,2,3
    3 B6=ATAN (Y/X)
      T=X*X
    5 B7=Y*Y+T
C  REAL PART OF LOG
      T1=0.5*ALOG(B7)
      IF(X-2.0)7,7,6
    7 B1=B1+B6
      B2=B2+T1
      X=X+1.0
      J=1
      GO TO 4
    6 T3=-Y*B6+(T1*(X-.5)-X+.9189385332 )
      T2=B6*(X-.5)+Y*T1-Y
      T4=X
      T5=-Y
      T1=B7
      DO 8 I=1,7
      T=H(I)/T1
      T4=T*T4+X
      T5=-(T*T5+Y)
    8 T1=T4**2+T5**2
      T3=T4-X+T3
      T2=-T5-Y+T2
      GO TO (9,10),J
    9 T3=T3-B2
      T2=T2-B1
   10 IF(X2)11,12,12
   12 U=T3
      V=T2
      X=X2
      RETURN
   11 U=T3-E4
      V=T2-E5
      X=X2
      RETURN
C    X IS ZERO
```

```
     2 T=0.0
       IF(Y)13,14,15
    13 B6=-E2
       GO TO 5
    15 B6=E2
       GO TO 5
C     X IS NEGATIVE
     1 E4=0.0
       E5=0.0
       IE6=0
    16 E4=E4+.5*(ALOG(X*X +Y*Y ))
       E5=E5+ATAN (Y/X)
       IE6=IE6+1
       X=X+1.0
       IF(X)16,17,17
    17 IF( MOD (IE6,2))18,4,18
    18 E5=E5+E8
       GO TO 4
    14 WRITE(6,19)              X2,Y
    19 FORMAT(29H ATTEMPTED TO TAKE LOGGAM OF 2HX=F6.0,1X2HY=F6.0)
       CALL EXIT
       RETURN
       END
```

```
$IBFTC PHISUP  NOLIST,NODECK
      SUBROUTINE PHISUM ( L, L1, PH )
C     SUM OVER KAPPAS
      COMMON /FUNCT/HF(1500,3),NEQ,X,H
      COMMON /INDEX/LBES(1500),KF(1500),KG(1500),NK,NFG,LVEC(11,22)
      COMMON /KAP/LMK(44),LPK(44),SI(44),CR(44),FKAP(44)
      COMMON /MAXL/JM,LM,LM1,LM2,IEND,JFLAG,ITEST,IREP
      DIMENSION   P(3, 3, 12) , PH(3, 3, 12)
      NEM = L1 + 2
      DO 1 NLA = 1, 3
      DO 1 NLB = 1, 3
      DO 1 NE = 1, NEM
      PH ( NLA, NLB, NE ) = 0.
    1 CONTINUE
      IF (JFLAG.NE.0)     GO TO 29
      KA = 1
      MAK = 2*LM2+2
      KB = MAK
      KQ = 1
      IF ( L .GT. 0 )             GO TO 3
      LQQ = 0
      GO TO 5
    3 LQQ = LVEC ( L, MAK )
    4 IF ( L1 )    5, 5, 6
    5 LQQ1 = 0
      GO TO 28
    6 LQQ1 = LVEC ( L1, MAK )
   28 IF ( LQQ .EQ. LVEC(L+1,1) )       KQ = 2
      IF ( LQQ1 .EQ. LVEC(L1+1,1) )     KQ = 2
      LONE  = LQQ  + 1
      LONE1 = LQQ1 + 1
      GO TO 7
   29 LONE = LVEC(L+1,JFLAG)+1
      LONE1 = LVEC(L1+1,JFLAG)+1
      KA = JFLAG+1
      KB = MAK-JFLAG
    7 DO 27 K=KA,KB
      LAST = LVEC ( L + 1, K )
      LAST1 = LVEC ( L1 + 1, K )
      IF ( K .EQ. 1 )                   GO TO 22
      IF ( LVEC(L+1,K) .EQ. LVEC(L+1,K-1) )      GO TO 30
      IF ( LVEC(L1+1,K) .EQ. LVEC(L1+1,K-1) )    GO TO 30
      GO TO 23
   22 IF ( KQ .EQ. 2 )             GO TO 30
   23 MG = KG ( LONE )
      JK2 = LMK ( MG ) + LPK ( MG )
      XJK2 = JK2+1
      DO 101 NLA = 1, 3
      DO 101 NLB = 1, 3
      DO 101 NE = 1, NEM
      P( NLA, NLB, NE) = 0.
  101 CONTINUE
      DO 25 N = LONE, LAST
      LB = LBES ( N )
      GO TO ( 8, 9, 10, 11, 12, 13 ) ,    LB
    8 NA = 1
      NB = 3
      GO TO 14
    9 NA = 2
      NB = 3
```

```
      GO TO 14
   10 NA = 3
      NB = 3
      GO TO 14
   11 NA = 1
      NB = 1
      GO TO 14
   12 NA = 1
      NB = 2
      GO TO 14
   13 NA = 2
      NB = 2
   14 MF = KF ( N )
      JK = LPK ( MF ) + LMK ( MF )
      DO 25 NLA = NA, NB
      LA = 2 * ( L + NLA - 2 )
      DO 25 N1 = LONE1, LAST1
      LB1 = LBES ( N1 )
      GO TO ( 15, 16, 17, 18, 19, 20 ) ,     LB1
   15 NA1 = 1
      NB1 = 3
      GO TO 21
   16 NA1 = 2
      NB1 = 3
      GO TO 21
   17 NA1 = 3
      NB1 = 3
      GO TO 21
   18 NA1 = 1
      NB1 = 1
      GO TO 21
   19 NA1 = 1
      NB1 = 2
      GO TO 21
   20 NA1 = 2
      NB1 = 2
   21 MF1 = KF ( N1 )
      JK1 = LPK ( MF1 ) + LMK ( MF1 )
      DO 25 NLA1 = NA1, NB1
      LA1 = 2 * ( L1 + NLA1 - 2 )
      MIL = MIN0 (LA,LA1,JK2+1)
      NIM = MIL / 2 + 1
      DO 24 NE = 1, NIM
      LE = 2 * ( NE - 1 )
      IF ( JK2 .LT. LE )                    GO TO 32
      CALL CCOEFS ( LA  , JK2, JK  , LE , -LE-1, CA )
      CALL CCOEFS ( LA1 , JK2, JK1 , LE , -LE-1, CB )
      SUMEM = - CA * CB
      IF ( ( LPK(MF)-LMK(MF)+LPK(MF1)-LMK(MF1) ) .EQ. 0 )  SUMEM=-SUMEM
      GO TO 33
   32 SUMEM = 0.0
   33 CALL CCOEFS ( LA  , JK2, JK  , LE , -LE+1, CA )
      CALL CCOEFS ( LA1 , JK2, JK1 , LE , -LE+1, CB )
      SUMEM = SUMEM - CA * CB
      LMOD = MOD( (LPK(MF)-LPK(MF1)+L-L1), 4 )
      IF (LMOD.EQ.0)        SUMEM = -SUMEM
      SUMEM = SUMEM*( CR(MF)*CR(MF1) + SI(MF)*SI(MF1) ) * 0.5
      P(NLA,NLA1,NE) = P(NLA,NLA1,NE) + SUMEM*HF(N,NLA)*HF(N1,NLA1)
   24 CONTINUE
   25 CONTINUE
```

```
      DO 26 NLA  = 1, 3
      DO 26 NLA1 = 1, 3
      DO 26 NE   = 1, NEM
      PH(NLA,NLA1,NE) = PH(NLA,NLA1,NE) + P(NLA,NLA1,NE)*XJK2
   26 CONTINUE
   30 LONE = LAST + 1
      LONE1 = LAST1 + 1
   27 CONTINUE
      RETURN
      END
```

```
$IBFTC RATER   NOLIST,NODECK
      FUNCTION RATERP(X)
C     OBTAIN VALUES FROM SCF TAPE AND INTERPOLATE
      COMMON /SCF/NTAPE,NTOT,NSKIP,NCOUNT
      DIMENSION TABR(101),TABRAT(101)
      IF (NCOUNT)                                              99,1,2
    1 CONTINUE
C ***              NCOUNT.EQ.0 -- FIRST TIME INTO RATERP FOR THIS ELEMENT
      NCOUNT = NCOUNT + 1
      IF (NCOUNT-NSKIP)                                        11,10,99
   11 CONTINUE
C ***                DIFFERENCE.GT.0 -- HAVE NOT REACHED LAST BINARY
C         RECORD OF TABLE.
      ISTOP = 101
      GO TO                                          12
   10 CONTINUE
C ***                DIFFERENCE.EQ.0 -- HAVE REACHED LAST BINARY RECORD
C         OF TABLE.
      ISTOP =  MOD (NTOT,100) + 1
   12 CONTINUE
C ***                READ IN NEW BINARY RECORD.
      READ (NTAPE)  (TABR(I),TABRAT(I),I=2,ISTOP)
      ISTART = 2
    2 CONTINUE
C ***                NCOUNT.GT.0 -- RE-ENTRY INTO RATERP FOR ELEMENT.
      DO  3   I=ISTART,ISTOP
C ***                LOOK FOR TABLE INTERVAL FOR X.
      IF (X-TABR(I))                                           30,4,3
   30     IF (ABS (X-TABR(I)) - .0001)      4,4,5
    3 CONTINUE
C ***                X TOO LARGE -- CARRY OVER LAST VALUES AND READ
C         IN NEW BINARY RECORD.
      TABR(1) = TABR(101)
      TABRAT(1) = TABRAT(101)
      GO TO                                          1
    4 CONTINUE
C ***                EXACT MATCH -- NO INTERPOLATION NECESSARY.
      VAL = TABRAT(I)
      ISTART = ISTART + 1
      GO TO                                          6
    5 CONTINUE
C ***                COMPUTE RATIO NEEDED FOR INTERPOLATION.
      TERM = (X-TABR(I-1)) / (TABR(I)-TABR(I-1))
      IF (NTOT-100)                                            7,7,8
    8 CONTINUE
C ***                LINEAR INTERPOLATION.
      VAL = TABRAT(I-1) + (TABRAT(I)-TABRAT(I-1)) * TERM
      GO TO                                          9
    7 CONTINUE
C ***                SEMI-LOG INTERPOLATION.
      VAL = TABRAT(I-1) *  (TABRAT(I)/TABRAT(I-1)) ** TERM
    9 CONTINUE
      ISTART = I
    6 CONTINUE
C ***                STORE VALUE IN FUNCTION.
      RATERP = VAL
      RETURN
   99 PRINT 1000,NCOUNT
      WRITE (6,1000) NCOUNT
 1000 FORMAT (10X,9HNCOUNT = I6,10X,28HREADING SCF TAPE INCORRECTLY )
```

```
      CALL EXIT
      RETURN
      END
```

```
$IBFTC RKUTT   NOLIST,NODECK
       SUBROUTINE RKUTTA
C      RUNGE-KUTTA INTEGRATION
       COMMON /FUNCT/S(1500),DY(1500),Y(1500),NEQ,X,H
       COMMON /KUT/D(4),E(4),F(4),G(4)
       DO  130                    J=1,4
       CALL DIRAC
       DO  120                    I=1,NEQ
       Z = D(J)*(DY(I)-E(J)*S(I))
       Y(I) = Y(I) + H*Z
  120  S(I) = S(I) + 3.0*Z - F(J)*DY(I)
  130  X = X + G(J)*H
       RETURN
       END
```

```
$IBFTC SELEC   NOLIST,NODECK
       SUBROUTINE SELECT
C      MATRIX ELEMENT SELECTION RULES AND INDEXING
       COMMON /INDEX/LBES(1500),KF(1500),KG(1500),NK,NFG,LVEC(11,22)
       COMMON /MAXL/JM,LM,LM1,LM2,IEND,JFLAG,ITEST,IREP
       SQL = LM * ( LM + 1 ) + LM1 * ( LM1 + 1 ) + LM2 * ( LM2 + 1 )
       RTL = SQRT ( SQL )
       LTP = RTL + 2.0
       K1  =  2*LM1 + 2
       NK = 0
       L = 0
       K = 0
   20  KAP2  =  LM2 + 1
       KSN2  =  1
       N2  =  LM2 + 1
   30  LKAP2  =  IABS (KAP2) + (KSN2-1)/2
       LMKAP2  =  LKAP2 - KSN2
   40  LQ  =  IABS (L - LMKAP2)
       LR  =  IABS (L - LKAP2)
       LMKAP1  =   MIN0 (LQ,LR)
   41  IF  (LMKAP1-LQ)          50,42,50
   42  IF  (LMKAP1)             43,50,43
   43  LMKAP1  =  LMKAP1 - 1
   50  IF ( MOD (L+LMKAP1+LKAP2, 2))       60,70,60
   60  LMKAP1  =  LMKAP1 + 1
   70  LK  =   MIN0 (LKAP2+L,LM1)
       LP  =   MIN0 (LMKAP2+L,LM1)
   80  KAP1  =  LMKAP1 + 1
       KSN1  =  1
       N1  =  LM1 + 1
   90  IF  (LKAP2 - LM2)                  100,100,130
  100  IF  (LR-LMKAP1)                    110,110,130
  110  IF ( LMKAP1 - LK )                     115, 115, 130
  115  IF ( LMKAP1 + LKAP2 + L - LTP )        120, 120, 130
  120  NK = NK + 1
       LBES(NK)  =  L + 1
       KF(NK)  =  KAP1 + N1
       KG(NK)  =  KAP2 + N2 + K1
  130  IF  (LMKAP2 -LM2)                  140,140,180
  140  LKAP1  =  LMKAP1 + KSN1
  150  IF  (LQ - LKAP1)                  160,160,180
  160  IF ( LKAP1 - LP )                      165, 165, 180


  165  IF ( LKAP1 + LMKAP2 + L - LTP )        170, 170, 180
  170  NK  =  NK + 1
       LBES(NK)  =  L + 1
       KF(NK)  =  KAP2 + N2 + K1
       KG(NK)  =  KAP1 + N1
  180  IF  (KSN1)              200,200,190
  190  KAP1 = -LMKAP1
       KSN1  =  -1
       N1  =  LM1 + 2
  191  IF  (KAP1)              90,200,90
  200  LMKAP1  =  LMKAP1 +2
  210  IF  (LMKAP1 - LK)                  80,80,220
  220  IF  (LMKAP1-1-LP)                  80,80,230
  230  IF  (LMKAP2 - LM2)                 240,240,260
  240  KAP2  =  KAP2 - 1
       K = K +1
```

```
      LVEC(L+1,K) = NK
      IF  (KAP2)                    30,250,30
  250 KAP2  =  -1
      KSN2  =  -1
      N2  =  LM2 + 2
      GO  TO                        30
  260 K = K +1
      LVEC(L+1,K) = NK
      L = L + 1
      K = 0
  270 IF  (L - LM)                              20,20,300
  300 MAK = 2 * LM2 + 2
  310 IF ( LVEC ( LM+1, MAK ) .GT. LVEC ( LM, MAK ) )      GO TO 320
      LM = LM - 1
      GO TO 310
  320 RETURN
      END
```

```
$IBFTC SETAP   NOLIST,NODECK
       SUBROUTINE SETAPE(Z)
C *** POSITION SCF TABLE TAPE
       COMMON /SCF/NTAPE,NTOT,NSKIP,NCOUNT
       REWIND NTAPE
  402  CONTINUE
       READ (NTAPE) ZTRY,NTOT
       NSKIP = NTOT/100 + 1
C ***                  MATCH CURRENT Z TO TAPE RECORD.
       IF (Z-ZTRY)                                          403,405,403
  403  CONTINUE
C ***                  NO MATCH -- SKIP UNWANTED TABLES.
       DO 404   I=1,NSKIP
  404  READ (NTAPE) DUMMY
       GO TO                                          402
  405  CONTINUE
C ***                  MATCH -- TAPE POSITIONED AT START OF DESIRED TABLES.
       NCOUNT = 0
       WRITE (6,5003) Z,NTOT
 5003  FORMAT (49HOUSING SCF POTENTIAL.  LIBRARY TABLE FOR ELEMENT , F4.0
      A, 5H HAS , I5, 9H ENTRIES. )
       RETURN
       END
```

```
$IBFTC XFUNP   NOLIST,NODECK
       SUBROUTINE XFUN(L,K1,K2,NLA,GEL)
C      REDUCED X-COEFFICIENT
       GO TO (1,2,3) ,NLA
     1 FNUM = K1+K2+L
       DEN = L*(2*L+1)
       GO TO 4
     2 FNUM=K2-K1
       DEN = L*(L+1)
       GO TO 4
     3 FNUM=K1+K2-L-1
       DEN = (L+1)*(2*L+1)
     4 GEL = FNUM/SQRT(DEN)
       RETURN
       END
```

ACKNOWLEDGMENT

The authors take this opportunity to acknowledge the help received on this project from colleagues at Union Carbide: Dr. J. Agresta contributed the angular distribution routine. Programming assistance was provided by Mr. R. A. Fronapfel in the first year and Mr. A. Glick thereafter, with smaller scale contributions by Miss C. Currie and Mr. J. W. Nostrand, Jr. Dr. S. K. Penny checked major portions of both the analysis and the program.

We are indebted to Dr. J. T. Waber of Los Alamos National Laboratory for supplying us his self-consistent field atomic wavefunctions and potentials on tape.

The cooperation of Mr. Q. Peasley at the Marshall Space Flight Center through many computer runs and frequent program changes is gratefully acknowledged.

REFERENCES

1. H. A. Bethe and W. Heitler, Proc. Royal Soc. (London) A146, 83 (1934).

2. H. W. Koch and J. W. Motz, Rev. of Mod. Phys. 31, 920 (1959).

3. J. C. Jaeger and H. R. Hulme, Proc. Royal Soc. (London) A153, 443 (1936).

4. J. C. Jaeger, Nature 137, 781 (1936).

5. J. C. Jaeger, Nature 148, 86 (1941).

6. Rosenblum, Shroder, and Warner, Phys. Rev. 88, 612 (1952); I. E. Dayton, Phys. Rev. 89, 544 (1953); H. I. West, Jr., Phys. Rev. 101, 915 (1956); C. D. Zerby and H. S. Moran, "A Semi-Empirical Equation for Pair Production Below 5 Mev", ORNL-CF-56-12-4 (1956).

7. A. I. Akhiezer and V. B. Berestetsky, Quantum Electrodynamics (State Technico-Theoretical Literature Press, Moscow, 1953) translated version: Technical Information Service Extension, Oak Ridge, Tenn., AEC-tr-2876.

8. M. E. Rose, Elementary Theory of Angular Momentum, (John Wiley and Sons, New York, 1957).

9. A. Erdelyi, et al., Higher Transcendental Functions (McGraw-Hill Book Company, New York, 1953) Vol. I and II.

10. W. Heitler, The Quantum Theory of Radiation, (Oxford University Press, London, 1954).

11. M. E. Rose and L. C. Biedenharn, "Theory of the Photoelectric Effect. I. Formal Aspects," Oak Ridge National Laboratory Report ORNL 1779 (1954).

12. M. E. Rose, "On the Relation Between Certain 9-j and 6-j Symbols," (unpublished).

13. M. Abramowitz and I. A. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, National Bureau of Standards (Washington, D. C., 1964).

14. D. Liberman, J. T. Waber, and D. T. Cromer, Phys. Rev. 137, A27 (1965).

15. S. Gill, Proc. Camb. Phil. Soc. 47, 96 (1951).

16. Miriam S. Shapiro and Max Goldstein, "A Collection of Mathematical Computer Routines," NYO-1480-14 (1965).

17. E. U. Condon and G. H. Shortley, Theory of Atomic Spectra, (Cambridge University Press, 1935).

18. G. Racah, Phys. Rev. 62, 438 (1942).

19. L. C. Biedenharn and M. E. Rose, Rev. Mod. Phys. 25, 729 (1953).

20. J. W. Motz and R. C. Placious, Phys, Rev. 109, 235 (1958).

21. F. Sauter, Ann. Physik (5), 20, 404 (1934).