SLAC-351
UC-414
(T)

DISCRETIZED LIGHT-CONE QUANTIZATION: APPLICATION TO QUANTUM ELECTRODYNAMICS*

Andrew C. Tang

Stanford Linear Accelerator Center
Stanford University
Stanford, California 94309

June 1990

Prepared for the Department of Energy
under contract number DE-AC03-76SF00515

Printed in the United States of America. Available from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, Virginia 22161. Price: Printed Copy A11, Microfiche A01.

* Ph.D. thesis

DISCRETIZED LIGHT-CONE QUANTIZATION: APPLICATION TO QUANTUM ELECTRODYNAMICS

Andrew C. Tang, Ph.D.
Stanford University, 1990

In this work, a general method for solving quantum field theories, Discretized Light-Cone Quantization (DLCQ), is presented. The method is very straightforward and essentially consists of diagonalizing the light-cone Hamiltonian matrix for the mass spectrum and wavefunctions. This method has been applied successfully in the past to various one space, one time dimensional theories. In each of these past applications, the mass spectrum and wave functions were successfully obtained, and all results agree with previous analytical and numerical work.

The success of DLCQ in 1+1 dimensions provides the hope of solving theories in three space and one time dimensions. The application to higher dimensions is much more involved than in 1+1 dimensions due to the need to introduce ultraviolet and infrared regulators, and invoke a renormalization scheme consistent with gauge invariance and Lorentz invariance. This is in addition to the extra work involved implementing two extra dimensions with their added degrees of freedom. In this paper, I will present the application of DLCQ to 3+1 dimensional Quantum Electrodynamics.

The theoretical framework of DLCQ in the context of 3+1 QED is shown in the first 8 sections. Issues addressed include the question of self-induced inertias and normal ordering, the agreement of Feynman rule and light-cone answers for one-loop radiative corrections, and ultraviolet and infrared regulation. Many of the results presented here are applicable to quantum field theory in general. Unfortunately, solving 3+1 QED in this general framework has so far proven elusive due to

a number of difficulties. These problems and a way around them using a truncated Fock space are presented in Section 7, with renormalization in this truncated space presented in Section 8. The next 5 sections show attempts to numerically solve 3+1 QED in a truncated Fock space by diagonalization of the Hamiltonian and by a variational calculation for the positronium system.

The numerical results shown are not competitive with state of the art calculations for positronium, but do demonstrate that the theoretical underpinnings of DLCQ are sound and that applications to other field theories such as Quantum Chromodynamics should be achievable. Further improvements in numerical technology may provide competitive answers.

ACKNOWLEDGEMENTS

Before beginning, I would like to recognize some important people that helped make this work possible. I wish to acknowledge the constant support and devotion given to me by my parents, Y.C. and Helen Tang; the generous input and time shown by my advisor, Professor Stanley Brodsky, without whose inspiration and tireless dedication this work would never have been possible; Professor H.C. Pauli and Monika Pauli, whose hospitality provided me a home away from home during my 7-month stay at the Max-Planck Institute for Nuclear Physics in Heidelberg, West Germany, and with whom I spent many wonderful hours discussing the intricacies and details of light-cone quantization; the members of my reading committee, Stanley Brodsky, Frederick Gilman, and Steven Chu; Professor Chris Hamer, who was the source of a very enlightening series of letters; my fellow graduate students who provided a constant sound-board for new ideas and questions, in particular Matjaz Kaluza, Michael Krautgaertner, Alexander Langnau, and Kent Hornbostel; the computer staff at the Stanford Linear Accelerator Center, especially Mark Barnett, who showed never ending willingness to answer my ceaseless computing questions; the Pittsburgh Supercomputing Center staff, who provided much of the computer resources; and last but not least, the theory secretarial staff at SLAC, Sharon Jensen and Robbin Nixon.

TABLE OF CONTENTS

1. INTRODUCTION

Discretized Light-Cone Quantization (DLCQ) is a general method for solving quantum field theories for their mass spectrum and wave functions. This method was developed for and applied successfully to Yukawa theory, $\overline{\psi}\psi\phi$, in one space and one time dimensions by Pauli and Brodsky.[1] Other successful applications in 1+1 dimensions include Eller, Pauli and Brodsky[2] to 1+1 QED and the Schwinger model, Harindranath and Vary[3] to ϕ^4 theory in 1+1 dimensions, and Hornbostel, Brodsky and Pauli[4] to 1+1 QCD for N_C = 2,3,4. In each of these applications, the mass spectrum and wave functions are successfully obtained, and all results agree with previous analytical and numerical work. For details, please refer to the original papers.

The success of DLCQ in 1+1 dimensions provides the hope of solving 3+1 theories. The application to higher dimensions is much more involved than in 1+1 dimensions due to the need to introduce ultraviolet and infrared regulators, and invoke a renormalization scheme consistent with gauge invariance and Lorentz invariance. This is in addition to the extra work involved implementing two extra dimensions with their added degrees of freedom. In this paper, I will present the application of DLCQ to 3+1 dimensional QED.

Sections 2 and 3 outline the general methodology of DLCQ. The results here are applicable to all field theories. Section 4 applies the general method to 3+1 QED. A variety of interesting problems are exposed and (hopefully) solved, including the inversion of the operators $i\partial^+$ and $(i\partial^+)^2$, the question of self-induced inertias and normal-ordering, and the derivation of the fermion mass renormalization insertion. With the introduction of ultraviolet and infrared regulators in Section 5 and 6, the general framework is outlined for 3+1 field theories. Unfortunately, solving 3+1 QED in this general framework has so far proven elusive due to a number of difficulties. These problems and a way around them using a truncated Fock space are presented in Section 7, with renormalization in this truncated space presented in Section 8. The next five sections describe attempts to numerically solve 3+1

QED in this truncated Fock space by diagonalizing the Hamiltonian and by a variational calculation for the ground state.

Those familiar with the language of DLCQ may wish to skip Sections 2 and 3 and continue to Sections 4 through 8 where the groundwork for 3+1 QED is laid. For those mainly interested in numerical results, these are in Sections 9, 10, 12, and 13. A variational method for finding the ground-state is described in Section 11. Most of the mathematical details are relegated to various appendices.

2. OVERVIEW OF LIGHT-CONE QUANTIZATION

An age old question of modern physics has been how to combine relativity (covariance) with a Hamiltonian formulation of dynamics. Typically, one might do this by specifying a particle's dynamical coordinates, say its position and momentum, at various time slices $t = t_0$. The system is quantized by specifying commutation relations between the various dynamical quantities on these equal-time surfaces. The particle is then propagated forward in time by the Hamiltonian, H. However, as Dirac[5] points out, one is not confined to specifying conditions on a surface of equal-time; in fact, any space-like hypersurface can be used. This is shown in Figure 1. The dynamics should be independent of the hypersurface chosen. Three specific forms are detailed by Dirac. The instant form is the traditional formulation given by hypersurfaces of equal time, $t = t_0$. The point form is described by surfaces with $x_\mu x^\mu = \kappa^2$, where κ is a constant. The form that the rest of this paper focuses on is the front form, now commonly referred to as light-cone quantization. In this form, dynamical quantities are specified on surfaces of equal light-cone time, $\tau = t + z/c$. A comparison of these various forms is given in Figure 2. One advantage of the front form is immediately evident. Whereas the Hamiltonian in the instant form, $H_{instant} = \sqrt{\vec{p}^2 + m^2}$, involves the square root operator, the light-cone Hamiltonian, $H_{front} = (\vec{p}_\perp^2 + m^2)/p^+$, does not. In fact, all other forms other than the front form involve the square root operator. This turns out to provide numerous simplifications, including only positive light-cone momenta and a simple vacuum structure. A comparison of light-cone quantization with equal-time quantization is shown in Table 1 and various definitions are given in Table 2.

The general method of quantizing a field theory proceeds as follows. The stress-energy tensor is derived from the Lagrangian in the usual fashion,

$$T^{\mu\nu} = \frac{\partial \mathcal{L}}{\partial(\partial_\mu \phi)} \partial^\nu \phi - g^{\mu\nu} \mathcal{L}. \tag{2.1}$$

The four plus components are conserved quantities,

$$\frac{\partial}{\partial\tau}P^{\mu}=0\ ,\qquad P^{\mu}=\frac{1}{2}\int dx^{-}d^{2}\vec{x}_{\perp}\ T^{+\mu}\ , \tag{2.2}$$

and correspond to the light-cone energy ($\mu=-$), light-cone momentum ($\mu=+$), and transverse momentum ($\mu=1,2$). One can question whether the boundary at $x^{-}=\infty$ is handled correctly. A good discussion of this matter can be found in McCartor.[6] He shows for massive theories that the above four quantities are not only conserved, but also are equivalent to the conserved quantities one would normally write down in an equal-time theory. That is, for massive theories,

$$P^{\mu}=\frac{1}{2}\int dx^{-}d^{2}\vec{x}_{\perp}\ T^{+\mu}=\int d^{3}\vec{x}\ T^{0\mu}\ . \tag{2.3}$$

The theory is then quantized by imposing equal light-cone time commutation relations between the various independent degrees of freedom, ϕ, and their momenta, $\pi=\frac{\partial\mathcal{L}}{\partial(\partial_{\tau}\phi)}$,

$$\left[\phi(x^{+},\underline{x}),\pi(x^{+},\underline{x}')\right]=i\delta^{(3)}(\underline{x}-\underline{x}')\ . \tag{2.4}$$

The above procedure is shown in more detail for 3+1 QED in Section 4.

A mathematically similar but conceptually different approach to light-cone quantization is infinite momentum frame physics. This method involves observing the system in a frame moving past the laboratory at the speed of light and was first uncovered by Weinberg.[7] It should be pointed out that though light-cone quantization is similar to infinite momentum frame quantization, it differs since no reference frame is chosen for calculations and is thus manifestly Lorentz covariant. The only aspect that "moves at the speed of light" is the quantization surface. Other works in infinite momentum frame physics include Drell, Levy and Yan,[8] Susskind and Frye,[9] Bjorken, Kogut and Soper,[10] and Brodsky, Roskies and Suaya.[11] This last reference presents the infinite momentum frame perturbation theory rules for QED in Feynman gauge, calculates one-loop radiative corrections, and demonstrates renormalizability.

Important papers in light-cone quantization include Casher,[12] Chang, Root and Yan,[13] Lepage and Brodsky,[14] Brodsky and Ji,[15] and Lepage, Brodsky, Huang and Mackenzie.[16] Casher gives the first construction of the light-cone Hamiltonian for non-Abelian gauge theory and gives an overview of important considerations in light-cone quantization. Chang, Root and Yan demonstrate the equivalence of light-cone quantization with standard covariant Feynman analysis. Detailed rules for QCD and applications to exclusive processes are provided by Lepage and Brodsky. They also present a table of light-cone spinor properties in Appendix A. A summary of the light-cone perturbation theory rules for QED in light-cone gauge and their derivation is given in Appendix B of Ref. 15 and Appendix A of Ref. 16. The notation used in this paper will follow that used in these two references. A recent summary of QCD in light-cone quantization can be found in Brodsky[17] and Brodsky and Lepage.[18]

3. DISCRETIZED LIGHT-CONE QUANTIZATION

An outstanding problem of modern quantum field theory is solving for bound states. The best available method at present is the Bethe-Salpeter formalism. However, calculations using this method are extremely complex and may be intractable beyond the ladder approximation. It may also not be practically possible to extend the method to systems with more than a few constituent particles.

A more intuitive approach would be to solve the equation

$$H\,|\psi\rangle = \sqrt{\vec{P}^2 + M^2}\,|\psi\rangle \tag{3.1}$$

for the particle's mass, M, and wavefunction, $|\psi\rangle$. Here, one imagines that $|\psi\rangle$ is an expansion in multi-particle occupation number Fock states and that the operators H and $\vec{P}$ are second quantized Heisenberg picture operators. Unfortunately, this method is severely complicated by the presence of the square root. This first of all introduces the mathematical difficulty of interpreting the square root as an operator. But more importantly, it leads to a very complicated vacuum structure involving spontaneous particle creation from the vacuum. This problem persists even if the above equation is replaced by a Dirac or Klein Gordon type of equation. Fortunately, light-cone quantization offers an avenue of escape. As already mentioned in Section 2, the square root operator does not appear in light-cone formalism, and as we will see in Section 4, there is no spontaneous particle creation in this theory.

The method of Discretized Light-Cone Quantization (DLCQ) was first developed by Pauli and Brodsky (see Refs. 1 and 2) as a general method for solving field theories. They applied the method to 1+1 Yukawa theory and 1+1 QED. The method was later extended to numerous other field theories (see Introduction). The derivation of the method follows.

The mass shell condition for a particle such as a pion with 4-momentum P^μ

written in light-cone variables is

$$P^- = \frac{\vec{P}_\perp^2 + M_\pi^2}{P^+} \tag{3.2}$$

where P^- and P^+ are the light-cone energy and momentum, respectively. This condition is equivalent to the usual space-time expression,

$$P^0 = \sqrt{\vec{P}^2 + M_\pi^2} \; . \tag{3.3}$$

Because the pion is an eigenstate of the system, it must satisfy this equation. In the spirit of second quantization, one imagines that $P^-, P^+, \vec{P}_\perp$ are Heisenberg picture operators and that the pion wavefunction is expanded in a complete set of multi-particle occupation number Fock states,

$$|\pi\rangle = \sum_n \psi_{n/\pi} \, |n\rangle \; . \tag{3.4}$$

$|\psi_{n/\pi}|^2$ is the probability of finding the Fock state $|n\rangle$ inside the pion. For example, the pion can be expanded into a quark pair Fock state, a Fock state with a quark pair and a gluon, and so forth,

$$|\pi\rangle = |q\bar{q}\rangle \, \psi_{q\bar{q}/\pi} + |q\bar{q}g\rangle \, \psi_{q\bar{q}g/\pi} + \ldots \; . \tag{3.5}$$

A single-particle state such as $|q\rangle$ is defined to be a quark creation operator acting on the Fock state vacuum,

$$|q\rangle = a_q^\dagger \, |0\rangle \; , \tag{3.6}$$

and the many-body Fock state with n_q quarks, $n_{\bar{q}}$ anti-quarks and n_g gluons is described by

$$|n\rangle = \sqrt{\mathcal{N}} \, \left| n_q : q_1, q_2, \ldots q_{n_q}; \; n_{\bar{q}} : \bar{q}_1, \bar{q}_2, \ldots \bar{q}_{n_{\bar{q}}}; \; n_g : g_1, g_2, \ldots g_{n_g} \right\rangle \tag{3.7}$$

where $\sqrt{\mathcal{N}}$ is a normalization factor that keeps $\langle n|n\rangle = 1$ and q_i, $\bar{q}_i$, and g_i are generic labels that describe all the quantum numbers of the ith component. The

decomposition of $|\pi\rangle$ into its Fock basis is shown schematically in Figure 3. The variables are explained below.

If one is working with a Lorentz invariant theory (i.e.: $\mathcal{L}$ transforms as a Lorentz scalar), the infinitesimal generators of Lorentz boosts, P^μ and $M^{\mu\nu}$ (energy-momentum and angular momentum), must satisfy the Poincare algebra,

$$\begin{aligned} [P^\mu, P^\nu] &= 0 \ , \\ i\left[M^{\mu\nu}, P^\lambda\right] &= -g^{\mu\lambda}P^\nu + g^{\nu\lambda}P^\mu \ , \\ i\left[M^{\mu\nu}, M^{\rho\sigma}\right] &= -g^{\mu\rho}M^{\nu\sigma} + g^{\nu\rho}M^{\mu\sigma} - g^{\mu\sigma}M^{\rho\nu} + g^{\nu\sigma}M^{\rho\mu} \ . \end{aligned} \tag{3.8}$$

In particular, P^+, P^-, and $\vec{P}_\perp$ form a commuting set. One should verify this for the specific Lagrangian one is interested in. If the basis set $|n\rangle$ is chosen such that P^+ and $\vec{P}_\perp$ are diagonal – this can be done by choosing the set of plane waves – then one can replace the operators P^+ and $\vec{P}_\perp$ by their respective eigenvalues. One then has

$$P^- \left|\pi : P^+, \vec{P}_\perp\right\rangle = \frac{P_\perp^2 + M_\pi^2}{P^+} \left|\pi : P^+, \vec{P}_\perp\right\rangle \ . \tag{3.9}$$

Assuming that the basis set $|n\rangle$ is complete, one can project out the nth component,

$$\sum_m \langle n| P^- |m\rangle \, \psi_{m/\pi} = \frac{P_\perp^2 + M_\pi^2}{P^+} \, \psi_{n/\pi} \ . \tag{3.10}$$

$\psi_{n/\pi}$ is the amplitude for finding the state $|n\rangle$ in $|\pi\rangle$ and is therefore a Lorentz invariant quantity. As a result, it can only be a function of the Lorentz invariant quantities x_i and $\vec{k}_{\perp i}$. x_i is the light-cone momentum fraction of the ith constituent of $|n\rangle$ and $\vec{k}_{\perp i}$ is the momentum of the ith constituent perpendicular to the total momentum of the pion. One can show by Lorentz transformation that the constituent's plus momentum and momentum perpendicular to the z direction, k_i^+ and $\tilde{\vec{k}}_{\perp i}$, are related to the pion's plus and perpendicular momenta, P^+, $\vec{P}_\perp$, by

$$k_i^+ = x_i P^+ \ , \qquad \tilde{\vec{k}}_{\perp i} = x_i \vec{P}_\perp + \vec{k}_{\perp i} \ . \tag{3.11}$$

x_i is also the antilog of the ith constituent particle's rapidity. Momentum conser-

vation requires

$$\sum_i x_i = 1 \ , \qquad \sum_i \vec{k}_{\perp i} = \vec{0}_\perp \ . \tag{3.12}$$

The Fock state expansion for the pion now reads,

$$\left|\pi : P^+, \vec{P}_\perp\right\rangle = \sum_n \psi_{n/\pi}(x_i, \vec{k}_{\perp i}, \lambda_i) \left|n : x_i P^+, x_i \vec{P}_\perp + \vec{k}_{\perp i}\right\rangle \tag{3.13}$$

with normalization

$$\langle \pi | \pi \rangle = 1 \ . \tag{3.14}$$

The sum is over all Fock states, momenta, and spins, and $\psi_{n/\pi}(x_i, \vec{k}_{\perp i}, \lambda_i)$ is the amplitude for finding the Fock state $|n\rangle$ with constituents with momenta $(x_i P^+, x_i \vec{P}_\perp + \vec{k}_{\perp i})$. Note that if the coefficients $\psi_{n/\pi}$ are determined for some P^+ and $\vec{P}_\perp$, then $\psi_{n/\pi}$ are known for all P^+ and $\vec{P}_\perp$ since $\psi_{n/\pi}$ is independent of these quantities. In particular, one may as well choose $\vec{P}_\perp = \vec{0}_\perp$ and $P^+ = M_\pi$. These are the values of the pion's plus and perpendicular momenta at rest. It should be emphasized that this choice does not imply that further calculations are being done in the pion's rest frame. The choice $\vec{P}_\perp = \vec{0}_\perp$ and $P^+ = M_\pi$ is only made for convenience because the coefficients $\psi_{n/\pi}$ are independent of P^+ and $\vec{P}_\perp$. Any other convenient choice is also acceptable.

We now define the light-cone Hamiltonian, H_{LC}, to be P^+P^-. Henceforth, the light-cone Hamiltonian is taken to mean the product of P^+ and P^-. The light-cone bound state equation reads,

$$\sum_m \langle n | \, H_{LC} \, | m \rangle \, \psi_{m/\pi}(x_i, \vec{k}_{\perp i}, \lambda_i) = M_\pi^2 \ \psi_{n/\pi}(x_i, \vec{k}_{\perp i}, \lambda_i) \ . \tag{3.15}$$

If one discretizes the Fock basis by requiring periodicity or anti-periodicity of the quark and gluon fields along the $x^+ = ct - z$ and $\vec{x}_\perp$ directions, one sees that the

bound state equation is a discrete matrix equation for the eigenvalues, M_π^2, and eigenvectors, $\psi_{n/\pi}$,

$$\left(M_\pi^2 - \sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i}\right) \begin{bmatrix} \psi_{q\bar{q}/\pi} \\ \psi_{q\bar{q}g/\pi} \\ \vdots \end{bmatrix} = \begin{bmatrix} \langle q\bar{q}|V|q\bar{q}\rangle & \langle q\bar{q}|V|q\bar{q}g\rangle & \cdots \\ \langle q\bar{q}g|V|q\bar{q}\rangle & \langle q\bar{q}g|V|q\bar{q}g\rangle & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} \psi_{q\bar{q}/\pi} \\ \psi_{q\bar{q}g/\pi} \\ \vdots \end{bmatrix} . \tag{3.16}$$

Here, H_{LC} has been split into an interacting piece, V, and a non-interacting piece, $H_0 = \sum_i (k_{\perp i}^2 + m_i^2/x_i)$. m_i is the mass of the ith constituent particle. For the case of the pion, it is either the quark mass or the gluon mass. Diagonalization of this equation can now be done on a computer (after implementing ultraviolet and infrared regulators) to reveal the complete spectrum of pion states and multi-particle scattering states with the same quantum numbers, along with their corresponding wavefunction expansion coefficients, $\psi_{n/\pi}$. Solving field theory has now been reduced to obtaining the solution to this fairly simple equation. In the next section, a specific application to 3+1 QED is detailed.

4. DLCQ: APPLICATION TO 3+1 QED

In this section, the light-cone Hamiltonian, $H_{LC} = P^+P^-$, is derived from the 3+1 dimensional QED Lagrangian. The approach given here is a heuristic, loosely structured approach based on canonical quantization similar to that used in Refs. 1 and 2. A more rigorous treatment can be made using Dirac's methods for handling constrained Hamiltonians. This is explored in Appendix A. One issue that will arise is how to invert the operators $i\partial^+$ and $(i\partial^+)^2$. The method used here was suggested by Hamer[19] and involves using the symmetrized form of the Lagrangian and making some simple arguments based on momentum conservation.

We begin with the familiar form for the 3+1 QED Lagrangian,

$$\mathcal{L} = \frac{i}{2}\left[\overline{\psi}\gamma^\mu\partial_\mu - (\partial_\mu\overline{\psi})\gamma^\mu\right]\psi - m_e\overline{\psi}\psi - \frac{1}{4}F_{\mu\nu}F^{\mu\nu} - g\overline{\psi}\gamma^\mu\psi A_\mu \,. \tag{4.1}$$

$F^{\mu\nu}$ is the electromagnetic field tensor and is equal to $\partial^\mu A^\nu - \partial^\nu A^\mu$. We will assume that the system obeys periodic boundary conditions in the $x^i, i = 1,2$ directions with period $2L_\perp$ and periodic or anti-periodic boundary conditions in the x^- direction with period $2L$. We also choose to work in light-cone gauge, $A^+ = A^0 + A^3 = 0$. It turns out, as will be shown shortly, that this gauge has the advantage of having only two, physical photons. This is very useful in the context of DLCQ since it makes the Fock state expansion easy to interpret. It also turns out that the development of light-cone gauge in light-cone quantization is very similar to axial gauge ($A^3 = 0$) in usual equal-time quantization.[20] It will be useful to define the projection operators

$$\begin{aligned} \Lambda_\pm &= \frac{1}{2}\gamma^0\gamma^\pm = \frac{1}{2}\gamma^0(\gamma^0 \pm \gamma^3) \,, \\ \psi_\pm &= \Lambda_\pm\psi \,. \end{aligned} \tag{4.2}$$

The operators Λ_+ and Λ_- have the following properties,

$$\Lambda_\pm^\dagger = \Lambda_\pm \,, \quad \Lambda_+ + \Lambda_- = 1 \,, \quad \Lambda_\pm^2 = \Lambda_\pm \,, \quad \Lambda_+\Lambda_- = 0 \,. \tag{4.3}$$

Using the expression for the canonical momenta for the various fields,

$$\pi_\phi = \frac{\partial \mathcal{L}}{\partial(\partial_\tau \phi)} \ , \tag{4.4}$$

results in

$$\begin{aligned} \pi_{\psi_+} = i\psi_+^\dagger \ , \quad \pi_{\psi_+^\dagger} = -i\psi_+ \ , \quad \pi_{\psi_-} = 0 \ , \quad \pi_{\psi_-^\dagger} = 0 \ , \\ \pi_{A_i} = -\partial^+ A^i \ , \quad \pi_{A_+} = 0 \end{aligned} \tag{4.5}$$

where we recall that $\tau = x^- = x^0 + x^3$ is the light-cone time. Since the fields ψ_-, $\psi_-^\dagger$ and A^- do not have canonical momenta, they must be eliminated. This can be done by solving the classical equations of motion for these fields,

$$\begin{aligned} i\partial^+ \psi_- &= \left[-i\partial_i \alpha^i + g A_i \alpha^i + \beta m_e\right] \psi_+ \ , \\ i\partial^+ \psi_-^\dagger &= \psi_+^\dagger \left[i\overleftarrow{\partial_i}\alpha^i + g A_i \alpha^i + \beta m_e\right] \ , \\ (i\partial^+)^2 A^- &= 2\partial^+ \partial_i A^i + 4g\psi_+^\dagger \psi_+ \ . \end{aligned} \tag{4.6}$$

$\overleftarrow{\partial}$ means the derivative acts to the left. These classical equations of motion are obtained from the usual Euler-Lagrange equation, $\partial_\mu \frac{\partial \mathcal{L}}{\partial(\partial_\mu \phi)} = \frac{\partial \mathcal{L}}{\partial \phi}$, and can be inverted by defining $\phi = \frac{1}{\partial_x} g$ to be the solution, including homogeneous terms, of $\partial_x \phi = g$. This gives

$$\begin{aligned} \psi_- &= \frac{1}{i\partial^+}\left[-i\partial_i \alpha^i + \beta m_e\right]\psi_+ - \frac{g}{i\partial^+} A^i \alpha^i \psi_+ \ , \\ \psi_-^\dagger &= \frac{-1}{i\partial^+}\psi_+^\dagger \left[i\overleftarrow{\partial_i}\alpha^i + \beta m_e\right] + \frac{g}{i\partial^+}\psi_+^\dagger A^i \alpha^i \ , \\ A^- &= \frac{2}{(i\partial^+)^2}\partial^+ \partial_i A^i + \frac{4g}{(i\partial^+)^2}\psi_+^\dagger \psi_+ \ . \end{aligned} \tag{4.7}$$

More details concerning $1/i\partial^+$ and $1/(i\partial^+)^2$ will be provided later. Observe that the only remaining independent degrees of freedom are ψ_+ and the two transverse, physical photons, $\vec{A}_\perp$. Upon substituting the expressions for the dependent fields,

writing everything in light-cone variables, and partial integrating ∂_i (this is permissible due to periodic boundary conditions in x^i), the light-cone energy density (density of P^-), $\mathcal{P}^- = 2\sum_\phi \pi_\phi \partial_+ \phi - 2\mathcal{L}$, becomes

$$\mathcal{P}^- = \mathcal{P}_0^- + g\mathcal{P}_1^- + g^2\mathcal{P}_2^- \ ,$$

$$\begin{aligned}
\mathcal{P}_0^- &= \partial^i A^j \partial^i A^j - \partial^i A^i \partial^j A^j + \left\{ i\partial^+ i\partial_i A^i \frac{1}{(i\partial^+)^2} i\partial^+ i\partial_j A^j \right\}_{\text{sym}} \\
&\quad + 2\left\{ \psi_+^\dagger \left[-i\partial_i \alpha^i + \beta m_e \right] \frac{1}{i\partial^+} \left[-i\partial_j \alpha^j + \beta m_e \right] \psi_+ \right\}_{\text{sym}} \ , \\
\mathcal{P}_1^- &= -2\left\{ \psi_+^\dagger A^i \alpha^i \frac{1}{i\partial^+} \left[-i\partial_i \alpha^i + \beta m_e \right] \psi_+ \right\}_{\text{sym}} + h.c. \\
&\quad - 4\left\{ \psi_+^\dagger \psi_+ \frac{1}{(i\partial^+)^2} i\partial^+ i\partial_i A^i \right\}_{\text{sym}} \ , \\
\mathcal{P}_2^- &= 2\left\{ \psi_+^\dagger A^i \alpha^i \frac{1}{i\partial^+} A^j \alpha^j \psi_+ \right\}_{\text{sym}} + 4\left\{ \psi_+^\dagger \psi_+ \frac{1}{(i\partial^+)^2} \psi_+^\dagger \psi_+ \right\}_{\text{sym}} \ .
\end{aligned} \tag{4.8}$$

$\{\ldots\}_{\text{sym}}$ are defined to be

$$\begin{aligned}
\left\{ A \frac{1}{i\partial^+} B \right\}_{\text{sym}} &= \frac{1}{2}\left[A \frac{1}{i\partial^+} B - \left(\frac{1}{i\partial^+} A \right) B \right] \ , \\
\left\{ A \frac{1}{(i\partial^+)^2} B \right\}_{\text{sym}} &= A \frac{1}{(i\partial^+)^2} B + \left(\frac{1}{i\partial^+} A \right)\left(\frac{1}{i\partial^+} B \right) + \left(\frac{1}{(i\partial^+)^2} A \right) B \ .
\end{aligned} \tag{4.9}$$

The system is now quantized by imposing canonical commutation relations on the independent fields,

$$\begin{aligned}
\{ \psi_{+\alpha}(x^+, \underline{x}), \psi_{+\beta}^\dagger(x^+, \underline{y}) \} &= \Lambda_{+\alpha\beta} \delta^{(3)}(\underline{x} - \underline{y}) \ , \\
\left[A^i(x^+, \underline{x}), \partial^+ A^j(x^+, \underline{y}) \right] &= i\delta^{ij} \delta^{(3)}(\underline{x} - \underline{y}) \ .
\end{aligned} \tag{4.10}$$

Once again, the above can also be derived rigorously using Dirac's method for constrained Hamiltonians. Recall also that the light-cone Hamiltonian, H_{LC}, is

related to $\mathcal{P}^-$ by

$$H_{LC} = P^+P^- = \left[\frac{1}{2}\int dx^+ d^2\vec{x}_\perp \; \mathcal{P}^+\right]\left[\frac{1}{2}\int dx^+ d^2\vec{x}_\perp \; \mathcal{P}^-\right] \; . \tag{4.11}$$

The extra factors of $1/2$ arise from the Jacobian transformation between $\vec{x}$ and $(x^+, \vec{x}_\perp)$. Also note that in the above development, no use is ever made of the relation

$$\int_{-L}^{L} dx \; A\frac{1}{\partial_x}B = -\int_{-L}^{L} dx \; \left(\frac{1}{\partial_x}A\right)B \; . \tag{4.12}$$

The system is discretized by expanding the fields in terms of solutions to the free equations of motion (plane waves), and requiring periodic boundary conditions for the photon field in the x^- and $x^i, i = 1,2$ directions, periodic boundary conditions for the fermion field in the $x^i, i = 1,2$ directions, and periodic or anti-periodic boundary conditions in the x^- direction. The numerical results depend very little on this last choice, and in the remainder of this paper anti-periodic boundary conditions will be used.

$$\psi_+(\underline{x}) = \frac{1}{\sqrt{\Omega}}\sum_{s,\underline{n}} \chi(s)\left[b_{s,\underline{n}}e^{-i\underline{k}\cdot\underline{x}} + d^\dagger_{-s,\underline{n}}e^{i\underline{k}\cdot\underline{x}}\right] \; ,$$

$$A^i(\underline{x}) = \frac{1}{\sqrt{\Omega}}\sum_{\lambda,\underline{p}} \frac{1}{\sqrt{k^+}}\left[a_{\lambda,\underline{p}}\epsilon^i(\lambda)e^{-i\underline{k}\cdot\underline{x}} + a^\dagger_{\lambda,\underline{p}}\epsilon^i(\lambda)^* e^{-i\underline{k}\cdot\underline{x}}\right] \; ,$$

$$\begin{aligned}
\text{fermions}: \quad k^i &= \frac{n^i\pi}{L_\perp} \quad n^i = 0, \pm 1, \pm 2, \ldots \\
k^+ &= \frac{n\pi}{L} \quad n = \begin{cases} 2,4,6,\ldots & \text{(periodic b.c.)} \\ 1,3,5,\ldots & \text{(anti-periodic b.c.)} \end{cases} \; , \\
\text{photons}: \quad k^i &= \frac{p^i\pi}{L_\perp} \quad p^i = 0, \pm 1, \pm 2, \ldots \\
k^+ &= \frac{p\pi}{L} \quad p = 2,4,6,\ldots \; ,
\end{aligned} \tag{4.13}$$

$$\chi(\uparrow) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} , \quad \chi(\downarrow) = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 1 \\ 0 \\ -1 \end{pmatrix} ,$$

$$\vec{\epsilon}_\perp(\uparrow) = \frac{-1}{\sqrt{2}}(1, i) , \quad \vec{\epsilon}_\perp(\downarrow) = \frac{1}{\sqrt{2}}(1, -i) .$$

Note that only positive k^+ are allowed. This is because the mass shell condition,

$$k^- = \frac{k_\perp^2 + m^2}{k^+} , \tag{4.14}$$

only allows for k^+ and k^- both positive or both negative. As one does in equal-time considerations, the modes with negative energy (in our case, negative k^-) are re-defined to be anti-fermions. The result is that in light-cone quantization, one only has states with both positive k^+ and positive k^-. The $k^+ = 0$ mode is eliminated because it turns out to be outside the range permitted by the ultraviolet cut-off (see Section 5). Besides which, the zero mode is not a solution to the free equation of motion for the fermion anyway. The canonical commutation relations are preserved by choosing

$$\left\{ b_{s,\underline{n}}, b^\dagger_{s',\underline{n}'} \right\} = \left\{ d_{s,\underline{n}}, d^\dagger_{s',\underline{n}'} \right\} = \delta_{s,s'} \delta^{(3)}_{\underline{n},\underline{n}'} , \quad \left[a_{\lambda,\underline{p}}, a^\dagger_{\lambda',\underline{p}'} \right] = \delta_{\lambda,\lambda'} \delta^{(3)}_{\underline{p},\underline{p}'} ,$$

$$\left\{ b_{s,\underline{n}}, d^\dagger_{s',\underline{n}'} \right\} = 0 . \tag{4.15}$$

Before continuing, the expressions for some of the other conserved quantities should be written down:

$$\begin{aligned} Q &= \frac{1}{2} \int dx^- d^2\vec{x}_\perp \, j^+ = g \int dx^- d^2\vec{x}_\perp \, \psi_+^\dagger \psi_+ , \\ P^+ &= \frac{1}{2} \int dx^- d^2\vec{x}_\perp \, T^{++} = \int dx^- d^2\vec{x}_\perp \left[\psi_+^\dagger (i\partial^+) \psi_+ + \frac{1}{2} (\partial^+ A^i)^2 \right] , \\ P^i &= \frac{1}{2} \int dx^- d^2\vec{x}_\perp \, T^{+i} = \int dx^- d^2\vec{x}_\perp \left[\psi_+^\dagger (i\partial^i) \psi_+ + \frac{1}{2} \partial^+ A^j \partial^i A^j \right] . \end{aligned} \tag{4.16}$$

The x^- integrals run from $-L$ to L and the $x^i, i = 1, 2$ integrals run from $-L_\perp$ to

$L_\perp$. Inserting the expansions for ψ_+ and $\vec{A}_\perp$ into these expressions is straightforward if one remembers that

$$\begin{aligned} \frac{1}{\Omega}\int dx^- d^2\vec{x}_\perp \; e^{-i(\underline{k}-\underline{k}')\cdot\underline{x}} &= \delta^{(3)}_{\underline{k},\underline{k}'} \;, \\ \chi^\dagger(s)\chi(s') &= \delta_{s,s'} \;, \\ \vec{\epsilon}_\perp(\lambda)^* \cdot \vec{\epsilon}_\perp(\lambda') &= \delta_{\lambda,\lambda'} \;. \end{aligned} \tag{4.17}$$

Doing so and normal-ordering to remove vacuum values results in

$$\begin{aligned} Q &= g \sum_{s,\underline{n}} \left[b^\dagger_{s,\underline{n}} b_{s,\underline{n}} - d^\dagger_{s,\underline{n}} d_{s,\underline{n}} \right] \;, \\ P^+ &= \sum_{\lambda,\underline{p}} k^+ a^\dagger_{\lambda,\underline{p}} a_{\lambda,\underline{p}} + \sum_{s,\underline{n}} k^+ \left[b^\dagger_{s,\underline{n}} b_{s,\underline{n}} + d^\dagger_{s,\underline{n}} d_{s,\underline{n}} \right] \;, \\ P^i &= \sum_{\lambda,\underline{p}} k^i a^\dagger_{\lambda,\underline{p}} a_{\lambda,\underline{p}} + \sum_{s,\underline{n}} k^i \left[b^\dagger_{s,\underline{n}} b_{s,\underline{n}} + d^\dagger_{s,\underline{n}} d_{s,\underline{n}} \right] \;. \end{aligned} \tag{4.18}$$

The last two equations are just statements of k^+ and $\vec{k}_\perp$ momentum conservation: P^+ is just the sum of the individual k^+s and $\vec{P}_\perp$ is just the sum of the individual $\vec{k}_\perp$s. These expressions are especially simple, and since they are already diagonal, the wavefunction, $|\psi\rangle$, can immediately be chosen to be an eigenstate of them. Choosing the eigenvalues to be $P^+ = 2m_e$ and $\vec{P}_\perp = \vec{0}_\perp$ (recall that this choice is not necessary, only convenient – see Section 3) gives

$$\begin{aligned} \left\{ \sum_{\lambda,\underline{p}} p \; a^\dagger_{\lambda,\underline{p}} a_{\lambda,\underline{p}} + \sum_{s,\underline{n}} n \left[b^\dagger_{s,\underline{n}} b_{s,\underline{n}} + d^\dagger_{s,\underline{n}} d_{s,\underline{n}} \right] \right\} |\psi\rangle \;&=\; \frac{2m_e L}{\pi} |\psi\rangle \;=\; K\,|\psi\rangle \;, \\ \left\{ \sum_{\lambda,\underline{p}} p^i a^\dagger_{\lambda,\underline{p}} a_{\lambda,\underline{p}} + \sum_{s,\underline{n}} n^i \left[b^\dagger_{s,\underline{n}} b_{s,\underline{n}} + d^\dagger_{s,\underline{n}} d_{s,\underline{n}} \right] \right\} |\psi\rangle \;&=\; 0\,|\psi\rangle \;, \\ p = 2,4,6,\ldots \;, \qquad & n = 1,3,5,\ldots \text{or } 2,4,6,\ldots \;, \\ & p^i, n^i = 0, \pm 1, \pm 2, \ldots \;. \end{aligned} \tag{4.19}$$

From now on, only those expansion states that satisfy these equations need be considered. In the first expression, the integer K is defined to be the eigenvalue

P^+ times L/π,

$$P^+ = \frac{K\pi}{L} \ . \tag{4.20}$$

In the past, K has been referred to as the harmonic resolution (Refs. 1 and 2).

Now turn to the expansion of the light-cone Hamiltonian $H_{LC} = P^+P^-$. The expressions

$$\begin{aligned} [n|m] &= \left(\frac{\pi}{L}\right)^2 \frac{1}{2L} \int_{-L}^{L} dx^- \ \left\{ e^{\frac{i}{2}k_n^+ x^-} \frac{1}{(i\partial^+)^2} \ e^{-\frac{i}{2}k_m^+ x^-} \right\}_{\rm sym} \ , \\ \{n|m\} &= \left(\frac{\pi}{L}\right) \frac{1}{2L} \int_{-L}^{L} dx^- \ \left\{ e^{\frac{i}{2}k_n^+ x^-} \frac{1}{i\partial^+} \ e^{-\frac{i}{2}k_m^+ x^-} \right\}_{\rm sym} \end{aligned} \tag{4.21}$$

occur frequently and are taken to be

$$\begin{aligned} [n|m] &= \begin{cases} \frac{1}{n^2}\delta_{n,m} & n,m \neq 0 \\ \kappa & n \text{ and } m = 0 \\ 0 & \text{otherwise} \end{cases} \ , \\ \{n|m\} &= \begin{cases} \frac{1}{n}\delta_{n,m} & n,m \neq 0 \\ 0 & n \text{ or } m = 0 \end{cases} \ . \end{aligned} \tag{4.22}$$

The derivation of these results is shown in Appendix B and makes use of k^+ and $\vec{k}_\perp$ momentum conservation to eliminate various homogeneous solutions that arise from inverting $i\partial^+$ and $(i\partial^+)^2$. The value of the constant κ will turn out to be irrelevant as long as it is not infinity because the quantity $[0|0]$ never occurs with the inclusion of the gauge invariant ultraviolet cut-off (see Section 5). The following spinor identities will also be necessary to complete the expansion of H_{LC}:

$$\begin{aligned} v^j \chi^\dagger(s) \epsilon_\lambda^i \alpha^i \alpha^j \chi(t) &= 2\delta_{s,t}\delta_{2s,\lambda} \epsilon_\lambda^i v^i \ , \\ v^j \chi^\dagger(s) \epsilon_\lambda^{i*} \alpha^i \alpha^j \chi(t) &= 2\delta_{s,t}\delta_{2s,-\lambda} \epsilon_\lambda^{i*} v^i \ , \\ \chi^\dagger(s) \epsilon_\lambda^i \alpha^i \beta \chi(t) &= -\sqrt{2}\delta_{s,-t}\delta_{2s,\lambda} \ , \\ \chi^\dagger(s) \epsilon_\lambda^{i*} \alpha^i \beta \chi(t) &= \sqrt{2}\delta_{s,-t}\delta_{2s,-\lambda} \ . \end{aligned} \tag{4.23}$$

Using all of the above relations and normal-ordering the Hamiltonian to remove

vacuum values and the self-induced inertias (more on these in Section 8), one finally obtains

$$
\begin{aligned}
&: H_{LC} : \; = \; H_0 + H_1 + H_2 \; , \\
&H_1 = V_{flip} + V_{noflip} \; , \quad H_2 = V_{instphot} + V_{instferm} \; , \\
&H_0 = \sum_{\lambda,\underline{p}} \frac{K}{p} \left[\left(\frac{p_\perp \pi}{L_\perp} \right)^2 + \lambda^2 \right] a^\dagger_{\lambda,\underline{p}} a_{\lambda,\underline{p}} \\
&\quad + \sum_{s,\underline{n}} \frac{K}{n} \left[\left(\frac{n_\perp \pi}{L_\perp} \right)^2 + m_e^2 \right] \left[b^\dagger_{s,\underline{n}} b_{s,\underline{n}} + d^\dagger_{s,\underline{n}} d_{s,\underline{n}} \right] \; , \\
&V_{flip} = g \frac{K m_e}{2\sqrt{\pi} L_\perp} \sum_s \sum_{\underline{p},\underline{m},\underline{n}} \frac{1}{\sqrt{p}} \\
&\quad \Big\{ + a_{2s,\underline{p}} b^\dagger_{s,\underline{m}} b_{-s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{p},\underline{m}} \left(\frac{1}{n} - \frac{1}{m} \right) + \text{h.c.} \\
&\qquad - a_{2s,\underline{p}} d^\dagger_{s,\underline{m}} d_{-s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{p},\underline{m}} \left(\frac{1}{n} - \frac{1}{m} \right) + \text{h.c.} \\
&\qquad + a^\dagger_{2s,\underline{p}} b_{s,\underline{m}} d_{s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{m},\underline{p}} \left(\frac{1}{n} + \frac{1}{m} \right) + \text{h.c.} \Big\} \; , \\
&V_{noflip} = g \sqrt{\frac{\pi}{2}} \frac{K}{L_\perp^2} \sum_s \sum_{\underline{p},\underline{m},\underline{n}} \frac{1}{\sqrt{p}} \\
&\quad \Big\{ + a_{2s,\underline{p}} b^\dagger_{s,\underline{m}} b_{s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{p},\underline{m}} \, \vec{\epsilon}_{\perp 2s} \cdot \left(\frac{\vec{p}_\perp}{p} - \frac{\vec{n}_\perp}{n} \right) + \text{h.c.} \\
&\qquad + a_{-2s,\underline{p}} b^\dagger_{s,\underline{m}} b_{s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{p},\underline{m}} \, \vec{\epsilon}_{\perp -2s} \cdot \left(\frac{\vec{p}_\perp}{p} - \frac{\vec{m}_\perp}{m} \right) + \text{h.c.} \\
&\qquad - a_{2s,\underline{p}} d^\dagger_{s,\underline{m}} d_{s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{p},\underline{m}} \, \vec{\epsilon}_{\perp 2s} \cdot \left(\frac{\vec{p}_\perp}{p} - \frac{\vec{n}_\perp}{n} \right) + \text{h.c.} \\
&\qquad - a_{-2s,\underline{p}} d^\dagger_{s,\underline{m}} d_{s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{p},\underline{m}} \, \vec{\epsilon}_{\perp -2s} \cdot \left(\frac{\vec{p}_\perp}{p} - \frac{\vec{m}_\perp}{m} \right) + \text{h.c.} \\
&\qquad - a^\dagger_{2s,\underline{p}} b_{s,\underline{m}} d_{-s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{m},\underline{p}} \, \vec{\epsilon}^{\,*}_{\perp 2s} \cdot \left(\frac{\vec{p}_\perp}{p} - \frac{\vec{n}_\perp}{n} \right) + \text{h.c.} \\
&\qquad - a^\dagger_{-2s,\underline{p}} b_{s,\underline{m}} d_{-s,\underline{n}} \, \delta^{(3)}_{\underline{n}+\underline{m},\underline{p}} \, \vec{\epsilon}^{\,*}_{\perp -2s} \cdot \left(\frac{\vec{p}_\perp}{p} - \frac{\vec{m}_\perp}{m} \right) + \text{h.c.} \Big\} \; ,
\end{aligned}
\tag{4.24}
$$

$$V_{instphot} = g^2 \frac{K}{2\pi L_\perp^2} \sum_{s,t} \sum_{\underline{k},\underline{l},\underline{m},\underline{n}}$$

$$\Bigg\{ - b^\dagger_{s,\underline{k}} b^\dagger_{t,\underline{l}} b_{s,\underline{m}} b_{t,\underline{n}} \; \delta^{(3)}_{\underline{k}+\underline{l},\underline{m}+\underline{n}} \; \frac{1}{2}[k-m|-l+m]$$

$$- d^\dagger_{s,\underline{k}} d^\dagger_{t,\underline{l}} d_{s,\underline{m}} d_{t,\underline{n}} \; \delta^{(3)}_{\underline{k}+\underline{l},\underline{m}+\underline{n}} \; \frac{1}{2}[k-m|-l+m]$$

$$- b^\dagger_{s,\underline{k}} d^\dagger_{-s,\underline{l}} b_{t,\underline{m}} d_{-t,\underline{n}} \; \delta^{(3)}_{\underline{k}+\underline{l},\underline{m}+\underline{n}} \; [k+l|\tilde{m}+n]$$

$$+ b^\dagger_{s,\underline{k}} d^\dagger_{-t,\underline{l}} b_{s,\underline{m}} d_{-t,\underline{n}} \; \delta^{(3)}_{\underline{k}+\underline{l},\underline{m}+\underline{n}} \; [k-m|-l+n]$$

$$+ d^\dagger_{s,\underline{k}} d_{t,\underline{l}} d_{s,\underline{m}} b_{-t,\underline{n}} \; \delta^{(3)}_{\underline{k},\underline{l}+\underline{m}+\underline{n}} \; [k-m|l+n] + \text{h.c.}$$

$$+ b^\dagger_{s,\underline{k}} b_{t,\underline{l}} b_{s,\underline{m}} d_{-t,\underline{n}} \; \delta^{(3)}_{\underline{k},\underline{l}+\underline{m}+\underline{n}} \; [k-m|l+n] + \text{h.c.} \Bigg\} \,,$$

$$V_{instferm} = g^2 \frac{K}{4\pi L_\perp^2} \sum_{s} \sum_{\underline{p},\underline{q},\underline{m},\underline{n}} \frac{1}{\sqrt{pq}}$$

$$\Bigg\{ + a^\dagger_{-2s,\underline{p}} a_{-2s,\underline{q}} b^\dagger_{s,\underline{m}} b_{s,\underline{n}} \; \delta^{(3)}_{\underline{p}+\underline{m},\underline{q}+\underline{n}} \{p+m|q+n\}$$

$$- a^\dagger_{2s,\underline{p}} a_{2s,\underline{q}} b^\dagger_{s,\underline{m}} b_{s,\underline{n}} \; \delta^{(3)}_{\underline{p}+\underline{m},\underline{q}+\underline{n}} \{p-n|q-m\}$$

$$+ a^\dagger_{-2s,\underline{p}} a_{-2s,\underline{q}} d^\dagger_{s,\underline{m}} d_{s,\underline{n}} \; \delta^{(3)}_{\underline{p}+\underline{m},\underline{q}+\underline{n}} \{p+m|q+n\}$$

$$- a^\dagger_{2s,\underline{p}} a_{2s,\underline{q}} d^\dagger_{s,\underline{m}} d_{s,\underline{n}} \; \delta^{(3)}_{\underline{p}+\underline{m},\underline{q}+\underline{n}} \{p-n|q-m\}$$

$$- a^\dagger_{2s,\underline{p}} a^\dagger_{-2s,\underline{q}} b_{s,\underline{m}} d_{-s,\underline{n}} \; \delta^{(3)}_{\underline{p}+\underline{q},\underline{m}+\underline{n}} \{p-m|-q+n\} + \text{h.c.}$$

$$- a^\dagger_{2s,\underline{p}} a_{2s,\underline{q}} b_{-s,\underline{m}} d_{s,\underline{n}} \; \delta^{(3)}_{\underline{p},\underline{q}+\underline{m}+\underline{n}} \{p-n|q+m\} + \text{h.c.}$$

$$+ a^\dagger_{2s,\underline{p}} a_{2s,\underline{q}} b_{s,\underline{m}} d_{-s,\underline{n}} \; \delta^{(3)}_{\underline{p},\underline{q}+\underline{m}+\underline{n}} \{p-m|q+n\} + \text{h.c.}$$

$$- a_{-2s,\underline{p}} a_{2s,\underline{q}} b^\dagger_{s,\underline{m}} b_{s,\underline{n}} \; \delta^{(3)}_{\underline{m},\underline{p}+\underline{q}+\underline{n}} \{p+n|-q+m\} + \text{h.c.}$$

$$- a_{-2s,\underline{p}} a_{2s,\underline{q}} d^\dagger_{s,\underline{m}} d_{s,\underline{n}} \; \delta^{(3)}_{\underline{m},\underline{p}+\underline{q}+\underline{n}} \{p+n|-q+m\} + \text{h.c.} \Bigg\} \,.$$

As explained before, $p, q, m, n, \ldots$ are allowed to take on the integer values

$$
\begin{aligned}
p^i, q^i, k^i, l^i, m^i, n^i &= 0, \pm 1, \pm 2, \ldots \ , \qquad i = 1, 2 \ , \\
p, q &= 2, 4, 6, \ldots \ , \\
k, l, m, n &= \begin{cases} 2, 4, 6, \ldots & \text{(periodic b.c.)} \\ 1, 3, 5, \ldots & \text{(anti-periodic b.c.)} \end{cases} \ ,
\end{aligned}
\tag{4.25}
$$

and λ is a small, fake photon mass. In the above, all terms involving [0|0] have been removed since these are eliminated by the gauge invariant ultraviolet cut-off. Also, as described above, one effectively has

$$
\begin{aligned}
[n|m] &= \begin{cases} \frac{1}{n^2}\delta_{n,m} & n, m \neq 0 \\ 0 & n \text{ or } m = 0 \end{cases} \ , \\
\{n|m\} &= \begin{cases} \frac{1}{n}\delta_{n,m} & n, m \neq 0 \\ 0 & n \text{ or } m = 0 \end{cases} \ .
\end{aligned}
\tag{4.26}
$$

One still needs to include fermion mass renormalization terms in H_{LC} (see Section 8). V_{flip} is the spin-flip amplitude for a (anti-) fermion to emit (absorb) a photon and V_{noflip} is the no spin-flip amplitude for this process. The familiar three-point Dirac QED vertex is just the sum of these two amplitudes. Two other types of vertices appear in light-cone quantization: a four-point instantaneous photon exchange, $V_{instphot}$, and a four-point instantaneous fermion exchange, $V_{instferm}$. These are just the graphs needed to reproduce the usual covariant Feynman S-matrix result for scattering amplitudes. An example of this for Møller scattering ($e^-e^- \to e^-e^-$) is shown in Appendix C. One can think of the instantaneous photon exchange graph in light-cone gauge as being analagous to the Coulomb exchange graph in Coulomb gauge. All the interactions conserve k^+ and $\vec{k}_\perp$, as they must, and are shown schematically in Figs. 4, 5, 6 and 7.

One very interesting feature of H_{LC} is that *it does not involve the longitudinal box size, L!* This is because $P^+ = \pi K/L$ is proportional to $1/L$, whereas it can be shown easily that P^- is proportional to L. Recall that H_{LC} has been defined to be P^+P^-.

Finally observe that because of k^+ momentum conservation and positivity of k^+, there are no interactions involving spontaneous creation or annihilation of a fermion pair and a photon from the vacuum. Because of this fact, the Fock state vacuum (the state with no particles) is an eigenstate of the light-cone Hamiltonian with mass zero,

$$H_{LC}\,|0\rangle = 0\,|0\rangle \ . \tag{4.27}$$

This immensely simplifies solving for bound states because it removes the need to constantly recalculate the vacuum.

We now proceed to solve the bound state equation Eq. (3.15) by implementing a Fock state expansion for the system we are interested in. In these considerations, we will focus mainly on positronium:

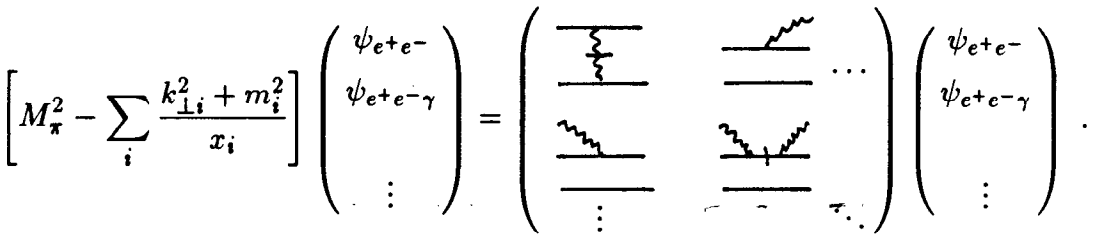
$$\begin{aligned} |\pi\rangle &= \sum_n \psi_{n/\pi}(x, \vec{k}_\perp)\,|n\rangle \\ &= \sum \psi_{e^+e^-}\,|e^+e^-\rangle + \psi_{e^+e^-\gamma}\,|e^+e^-\gamma\rangle + \dots \ . \end{aligned} \tag{4.28}$$

The sum is over all Fock states $|n\rangle$ with constituent momenta x_i and $\vec{k}_{\perp i}$. The overlap of positronium with the $e^+e^-\gamma$ Fock state is shown in Figure 8. The Fock states are eigenstates of P^+, $\vec{P}_\perp$, and H_0,

$$\begin{aligned} P^+ \left|n : k_i^+, \vec{k}_{\perp i}\right\rangle &= \frac{K\pi}{L} \left|n : k_i^+, \vec{k}_{\perp i}\right\rangle \ , \\ \vec{P}_\perp \left|n : k_i^+, \vec{k}_{\perp i}\right\rangle &= \vec{0}_\perp \left|n : k_i^+, \vec{k}_{\perp i}\right\rangle \ , \\ H_0 \left|n : k_i^+, \vec{k}_{\perp i}\right\rangle &= \sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} \left|n : k_i^+, \vec{k}_{\perp i}\right\rangle \ , \end{aligned} \tag{4.29}$$

but not of the interactions, V. After implementing ultraviolet and infrared cut-offs,

the matrix equation that must be solved for the positronium mass spectrum is



(4.30)

In summary, the DLCQ procedure is fairly straightforward. One derives the light-cone Hamiltonian from the Lagrangian by a procedure very similar to standard canonical quantization. The commuting operators, the light-cone momentum $P^+ = K\pi/L$, transverse momentum $\vec{P}_\perp$, and light-cone Hamiltonian $H_{LC} = P^+P^-$ are constructed by expanding in Fock states and are simultaneously diagonalized. The expressions for P^+ and $\vec{P}_\perp$ are already diagonal if one expands in plane waves. The system is discretized by requiring periodic or anti-periodic boundary conditions in the light-cone spatial dimensions and the system is quantized by imposing canonical commutation relations between the independent fields and their canonical momenta. The bound state equation $H_{LC}\,|\pi\rangle = M_\pi^2\,|\pi\rangle$ is diagonalized to obtain the invariant mass spectrum and wavefunctions. All of these quantities are independent of L. To recover the continuum theory, one lets K and $L_\perp$ approach infinity (this is equivalent to letting $L, L_\perp \to \infty$).

5. COVARIANT ULTRAVIOLET REGULATOR

Before continuing, a method of regulating the $\vec{k}_\perp$ Fock space and other ultraviolet divergences is necessary. Recall that the Fock space for QED is constructed by choosing the set of all states with n_{e^-} electrons, n_{e^+} positrons and n_γ photons with appropriate quantum numbers (charge, total k^+, total $\vec{k}_\perp$,...). The longitudinal momenta k^+ are taken to be odd or even multiples of π/L (depending on boundary conditions – see Section 4), the transverse momenta $\vec{k}_\perp$ are taken to be integer multiples of $\pi/L_\perp$, the total longitudinal momentum is taken to be $K\pi/L$, and the total transverse momentum is taken to be $\vec{0}_\perp$.

The Fock space is naturally finite in k^+ because the total k^+ is just the sum of the individual k^+s. Combining the fact that all the individual k^+s are positive, non-zero integers with the fact that there are only a finite number of ways of summing a set of positive, non-zero integers to form a given positive number demonstrates finiteness of the k^+ space. As an example, a Fock state with one electron and two photons with $K = 9$ can have the following quantum numbers (anti-periodic boundary conditions),

Fock State	1	2	3	4	5	6
Electron	1	1	1	3	3	5
Photon 1	2	4	6	2	4	2
Photon 2	6	4	2	4	2	2

.

In contrast to k^+, the Fock space is naturally infinite in $\vec{k}_\perp$ because $\vec{k}_\perp$ can take values that are positive or negative. An ultraviolet regulator must therefore be introduced. We will choose one such that the sum of the $(k_\perp^2 + m^2)/x$ of each Fock state is less than a cut-off value, Λ^2 (see Ref. 14),[⋆]

$$\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} \leq \Lambda^2 \,. \tag{5.1}$$

⋆ A local ultraviolet regulator has been shown to give better numerical convergence in higher-loop perturbation theory calculations. Applying such a regulator to bound state calculations is under investigation.

The sum is over all the individual particles in the Fock state under consideration. The left hand side of this equation is just the invariant mass (for a single particle state, the invariant mass is the rest mass) squared of the Fock state, $M^2 = P^+P^- + P_\perp^2$. It is also the value of the light-cone Hamiltonian at zero coupling. So, the ultraviolet regulator can be stated simply as requiring the invariant mass squared of the individual Fock states be less than Λ^2. Since the invariant mass is frame independent, this regulator is Lorentz invariant. It also turns out, with a modification described below, to be tree-level gauge invariant.

The off-shellness of a certain Fock state is given by

$$\begin{aligned} \sum_i k_i^- - P^- &= \sum_i \left[\frac{(\vec{k}_{\perp i} + x_i \vec{P}_\perp)^2 + m_i^2}{x_i P^+} \right] - \frac{P_\perp^2 + M^2}{P^+} \\ &= \frac{1}{P^+} \left[\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} - M^2 \right] . \end{aligned} \tag{5.2}$$

One sees immediately that the ultraviolet cut-off given in Eq. (5.1) removes Fock states that are far off-shell; this is a reasonable procedure because far off-shell states give only a small contribution to a physical wavefunction. In fact, one sees from Eqs. (3.15) and (3.16) that a typical wavefunction in QED will have the form

$$\psi_n(x_i, \vec{k}_{\perp i}, \lambda_i) = \frac{1}{M^2 - \sum_i (k_{\perp i}^2 + m_i^2)/x_i} \, (V\Psi) \tag{5.3}$$

which tends to vanish as

$$\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} - M^2 \to \infty \, . \tag{5.4}$$

In principle, one lets Λ go to infinity to recover the full theory. Practically, we will make all further deliberations at finite values of Λ and include all effects from Λ in the bare quantities $\alpha(\Lambda)$ and $m_e(\Lambda)$. Furthermore, since we are mainly interested in bound state properties, one would imagine that Fock states with large invariant mass squared (i.e: are far off-shell) have little effect. As a result, one can even calculate with fairly small values of Λ.

Cutting off the photon's momentum $\vec{k}_\perp$ can lead to problems with gauge invariance because the various graphs involved in photon exchange are cut-off in a different way. That is, one can imagine a situation in Møller scattering ($e^-e^- \to e^-e^-$), for example, in which the exchange of a real, physical photon is cut-off (the relevant Fock state is the e^-, e^-, γ intermediate state) but the exchange of an instantaneous photon is not (there is no intermediate state in this graph).

We now introduce a new method to restore gauge invariance by considering the instantaneous photon in the instantaneous photon exchange graph to have quantum numbers as if it were a real photon. One then cuts it off in a manner similar to the Fock state cut-off for a real intermediate state. That is, one requires

$$\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} \leq \Lambda^2 \tag{5.5}$$

where the sum is over the individual particles in the Fock state *plus* the instantaneous photon. Though it does not affect gauge invariance, a similar procedure is taken for the instantaneous fermion interaction so the correct Feynman S-matrix amplitudes are restored in this sector also. As a concrete example, consider the graphs involved in Møller scattering shown in Figure 9. Assume that k_1^+ is larger than k_3^+. In the first graph, the photon's momenta are fixed by momentum conservation, and the three particle intermediate state is cut-off by

$$\frac{k_{3\perp}^2 + m_e^2}{x_3} + \frac{k_{2\perp}^2 + m_e^2}{x_2} + \frac{q_\perp^2}{x_q} \leq \Lambda^2 \, . \tag{5.6}$$

In the second graph, one *assigns* momenta to the instantaneous photon, $q^+ = k_1^+ - k_3^+$, $\vec{q}_\perp = \vec{k}_{1\perp} - \vec{k}_{3\perp}$, and then requires

$$\frac{k_{3\perp}^2 + m_e^2}{x_3} + \frac{k_{2\perp}^2 + m_e^2}{x_2} + \frac{q_\perp^2}{x_q} \leq \Lambda^2 \, . \tag{5.7}$$

With this requirement, whenever the instantaneous photon exchange graph occurs, a corresponding graph with the exchange of a real, intermediate photon occurs because both graphs are now cut-off in exactly the same way! As shown in Appendix

C, the sum of the graphs is simply the gauge invariant Feynman rule answer, $1/q_F^2$. Thus, we see that this method maintains gauge invariance of the ultraviolet cut-off for tree-level diagrams. It is not clear if this conclusion can be carried over to loop diagrams.

We have now completed the ultraviolet regulation of light-cone theory. All Fock states are cut-off by requiring the invariant mass squared to be less than Λ^2,

$$\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} \leq \Lambda^2 \,. \tag{5.8}$$

Graphs involving an instantaneous photon or instantaneous fermion are treated as if they were real particles and cut-off in the same fashion. With this inclusion, the ultraviolet regulator is both Lorentz invariant and (tree-level) gauge invariant. We also note that this regulator is a continuum regulator: the cut-off condition is not changed by discretization.

6. COVARIANT INFRARED REGULATOR

There are a number of potential sources of infrared singularities and divergences in light-cone quantized QED. These are

1. Singularities in H_0, V_L and V_T from fermions at $x = 0$ ($k^+ = 0$),
2. Singularities and divergences in H_0, V_L and V_T from photons at and near $x = 0$,
3. Singularity in $V_{instferm}$ from the exchange of an instantaneous fermion at $x = 0$, and
4. Singularity and divergence in $V_{instphot}$ from the exchange of an instantaneous photon at and near $x = 0$.

A singularity is defined to be an expression of the form $1/0$; whereas a divergence is taken to be an expression that approaches infinity as x approaches 0. The definitions and equations for H_0, V_L, V_T, $V_{instferm}$ and $V_{instphot}$ were given in Section 4.

The singularity described in item 1 can be removed by requiring anti-periodic boundary conditions for the fermions in the x^- direction. Similarly, the singularity in item 3 is removed if the fermions obey anti-periodic boundary conditions and the photons periodic boundary conditions because the momentum exchange will never be zero. Recall that the instantaneous fermion interaction is proportional to $1/q^+$ where $q^+ = k^+_{outgoing\ photon} - k^+_{incoming\ fermion}$.

The singularity arising from photons with $x = 0$ (point 2) is eliminated by the cut-off described in the previous section if $\vec{q}_\perp \neq \vec{0}_\perp$ because the invariant mass squared of such a photon would be greater than any finite Λ^2. That is,

$$\frac{q_\perp^2}{x} > \Lambda^2 \tag{6.1}$$

for $q^+ = 0$. The case of $\vec{q}_\perp = \vec{0}_\perp$ is dealt with below. The singularity from instantaneous photons at $x = 0$ (point 4) and $\vec{q}_\perp \neq \vec{0}_\perp$ is eliminated because

instantaneous photons are treated for purposes of the cut-off as if they were real photons. As a result, they are also eliminated because

$$\frac{q_\perp^2}{x} > \Lambda^2 \tag{6.2}$$

where q^+ and $\vec{q}_\perp$ are assigned to the instantaneous photon according to momentum conservation as explained in Section 5. Again, the situation for $\vec{q}_\perp = \vec{0}_\perp$ is described below.

If periodic boundary conditions had been chosen for the fermions instead of anti-periodic conditions, the singularities at $x = 0$ for real and instantaneous fermions would be eliminated by the same reasoning as for real and instantaneous photons.

The divergence as x approaches 0 for real and instantaneous photons is removed by invoking an infrared cut-off,

$$\frac{q_\perp^2}{x} \geq \epsilon \, . \tag{6.3}$$

All Fock states with real photons not satisfying this condition and all instantaneous photon interactions not meeting this criterion are removed. Once again, q^+ and $\vec{q}_\perp$ for a real Fock state photon are taken to be their actual values; q^+ and $\vec{q}_\perp$ for an instantaneous photon are assigned according to momentum conservation as if it were a real photon. Similarly to the ultraviolet cut-off of Section 5, this cut-off is Lorentz invariant because it only involves a Lorentz invariant quantity, the photon's invariant mass, and is gauge invariant (at least for tree-level diagrams) because it treats all photons, real and instantaneous, alike.

Note that if ϵ is chosen to any number smaller than $(\pi/L_\perp)^2$ but greater than 0, then the only effect of the infrared cut-off is to remove photons with $\vec{q}_\perp = \vec{0}_\perp$. Since the effect of the cut-off is identical for all ϵ less than $(\pi/L_\perp)^2$, one may as well take the limit $\epsilon \to 0$ right away. Since the point $\vec{q}_\perp = \vec{0}_\perp$ has now been removed, the problem of the $x = 0$ singularity for real and instantaneous photons with zero

$\vec{q}_\perp$ described above has been taken care of. Another way of removing the point $x = 0$ when $\vec{q}_\perp = \vec{0}_\perp$ is to imagine that the photon has a small mass λ. Then $x = 0$ would be eliminated for all $\vec{q}_\perp$ by the ultraviolet cut-off, Eq. (5.1).

One might ask why an extra infrared cut-off is necessary. After all, the main problem we are interested in, the spectrum and wavefunction of positronium, has no infrared divergence. As compared to scattering processes that involve exposed charges, the positronium system's charges are "hidden" in the bound state. Therefore, there should be no infrared divergence from emitting an infinite number of long wavelength photons. As long as any infrared singularities are removed and any infrared divergences are properly regulated, the various infrared divergences arising from the fermion self-energy diagram and exchanges of real and instantaneous photons should cancel.

Unfortunately, matters are not so clean numerically. Without the infrared photon cut-off, one would expect that the infrared behavior is controlled by the parameter K. Once the point $x = 0$ is removed, the closest point to $x = 0$ is $x = 1/K$, which approaches zero as K approaches infinity. Figure 10 shows the behavior of the lowest energy level in a variational calculation as K is increased. Details of this calculation are described in Section 13. For now, the point is that the energy level diverges as $K \to \infty$ if an infrared cut-off is not included. This divergence is apparently removed by the inclusion of the cut-off. An explanation for this behavior is that the integral that must be reproduced to obtain the ground state energy level,

$$\langle \psi_0 | \, H_{LC} \, | \psi_0 \rangle = M_0^2 \ , \tag{6.4}$$

has an integrand that diverges like

$$\frac{1}{x(q_\perp^2 + m_e^2) - q_\perp^2} \tag{6.5}$$

for small $x, \vec{q}_\perp$. Of course, the integral itself is still finite. In the continuum, the points near $x = 0$, $\vec{q}_\perp = \vec{0}_\perp$ are a set of measure zero and give a finite contribution

to the integral. Unfortunately, in the discrete case, any one Fock state has a finite measure since there are only a finite number of Fock states. Each $(e^+e^-\gamma)$ Fock state contributes one point to the sum, Eq. (6.4). As a result, the Fock states with photon x near zero and $\vec{q}_\perp = \vec{0}_\perp$ give a contribution proportional to $1/x \sim K$. In the absence of a better solution, photons with $\vec{q}_\perp = \vec{0}_\perp$ must be removed by an infrared cut-off such as Eq. (6.3) to keep the sum Eq. (6.4) finite as $K \to \infty$. An approximate form for the ground state integral is given in Eq. (F.29), Appendix F.

In summary, an infrared regulator is included by requiring that all photons, real and instantaneous, have invariant mass squared greater that ϵ,

$$\frac{q_\perp^2}{x} \geq \epsilon \, . \tag{6.6}$$

This Lorentz invariant, (tree-level) gauge invariant regulator ensures that all infrared divergences are well defined and cancel in a charge neutral system such as positronium. The numerical proof for this last statement is shown in Section 13. Since the only effect of the cut-off is to remove photons with $\vec{q}_\perp = \vec{0}_\perp$ for any $0 < \epsilon < (\pi/L_\perp)^2$, the limit $\epsilon \to 0$ can be taken immediately. Also note that this infrared regulator is a continuum condition: the cut-off requirement is unaffected by discretization.

7. TRUNCATED FOCK SPACE

The basic layout for solving 3+1 QED has been set: The light-cone Hamiltonian and bound state equation are given in Section 4, ultraviolet regulation is described in Section 5, and infrared regulation in Section 6. Unfortunately, there are still a number of outstanding problems that prevent the solving of Eq. (3.15) for the spectrum of 3+1 QED. These include, but may not be limited to the following.

1. As of yet, no non-perturbative prescription is available for renormalization to all orders.

2. It is not clear if one needs to keep track of Fermi statistics in instantaneous interactions and renormalization counter-terms. Note that these may in fact be points of measure zero in the continuum.

   (a) Figure 11 shows two graphs that occur in the analysis of QED. It is clear that the intermediate state in the first graph requires Fermi statistics because it involves real fermions. Unfortunately it is not so clear whether Fermi statistics is required in the second graph. Furthermore, if statistics is necessary in the second graph, it is not clear how one would assign quantum numbers to the instantaneous fermion. In particular, the "spin" of the instantaneous fermion would be ambiguous.

   (b) A similar situation occurs in the consideration of renormalization pieces. The first graph in Figure 12 shows an interaction in QED. The intermediate state obeys Fermi statistics because it is constructed by a Fock state expansion in photon and fermion fields that necessarily abides by Fermi statistics. There is some question whether this first graph should even obey statistics. Given that it does, the problem is whether the second graphs should also. This graph is needed to renormalize the photon's mass. As a parenthetical comment, photon mass renormalization is not necessary in standard Pauli-Villars regulation because it is a subtractive regulation scheme. It turns out that the subtracted piece with

massive Pauli-Villars particles exactly cancels the self-mass of the original diagram. On the other hand, our regulator is not subtractive, so an explicit photon mass counter-term must be constructed. More on this in Appendix E.

3. Full QED has light by light scattering graphs as shown in Figure 13 that need to be regularized. As of yet, a method of regularizing these diagrams compatible with DLCQ has not been found.

The above problems, and possibly others, need to be answered before the full 3+1 QED light-cone Hamiltonian can be diagonalized. We will circumvent these difficulties by considering a truncated Fock space that allows only one extra photon. To be specific, for the case of $Q = 0$, the Fock space will be limited to just (e^+, e^-) and (e^+, e^-, γ). For $Q = -1$, the only Fock states will be (e^-) and (e^-, γ). The number of interactions effectively allowed in this truncated Fock space is very much reduced from the full set shown in Figs. 4, 5, 6, and 7. All graphs involving pair creation are effectively removed because the truncated Fock space does not allow for extra fermion pairs (diagrams 3, 6, 9, 11, 12, 17, 18, and 19). Diagrams 14, 16, 20, and 21 are effectively removed because they involve two photons in flight. Finally, diagram 10 is eliminated when it occurs in the presence of a spectator photon because such a situation also has two photons in flight. Taking all these removals into account, the only diagrams that need to be considered are 1, 2, 4, 5, 10, 13, and 15.

Limiting the Fock space may bring gauge invariance into question. However, we have carefully made sure that everytime an intermediate state with real photons is removed, the corresponding intermediate state with instantaneous photons is also removed. This restores gauge invariance because photons are thus removed from the theory in gauge invariant sets. For example the interaction $e^+e^- \to \gamma \to e^+e^-$ is removed from consideration because the intermediate state with one real photon has been eliminated. To restore gauge invariance, we have been careful to drop

diagram 9 which involves the same process, but through an instantaneous photon.*

It should be emphasized that though the Fock space is limited, the analysis remains non-perturbative because the allowed Fock states can be iterated as many times as one wishes. In particular, keeping only $(e^+e^-, e^+e^-\gamma)$ is similar to the ladder approximation in Bethe-Salpeter methods, which is an all orders calculation. Since this approximation has been solved in Bethe-Salpeter formalism for the spectrum of positronium, we have reasonable hope that diagonalizing the light-cone QED Hamiltonian in this truncated Fock space will also reproduce the positronium spectrum. Recall that the full Coulomb potential is completely contained in the exchange of a single photon; therefore, we should be able to reproduce the Bohr spectrum (non-relativistic Coulomb spectrum),

$$\epsilon_n \;=\; -\frac{1}{2} m_{\rm red} \left(\frac{Z\alpha}{n}\right)^2 \;, \tag{7.1}$$

in our truncated Fock space. Also included are $L \cdot S$ coupling, the hyperfine interaction, and the part of the Lamb shift coming from the fermion's self-energy diagram.

We expect that diagonalizing H_{LC} in the space $(e^+e^-, e^+e^-\gamma)$ gives back the positronium Bohr spectrum (actually, the muonium Bohr spectrum since the annihilation channel has been removed), plus continuum states. It should also contain the hyperfine splitting since this comes from the spin-spin interaction between the positron and the electron, and the first bit of the Lamb shift coming from Figure 14. Obtaining the true spectrum of positronium would require putting back the annihilation channel (fermion pair creation diagrams).

Diagonalizing the space $(e^-, e^-\gamma)$ gives back the "spectrum" of the electron in the cloud of a single photon, plus continuum states. Since the electron is an

---

⋆ Recent investigations have revealed a spurious $1/q^+$ singularity in light-cone gauge in a truncated Fock space which vanishes when the full Fock space is restored. This singularity is eliminated in this work by the choice of wavefunction used in the variational calculation presented in Section 11.

elementary particle, there is only one state in its spectrum: diagonalization is actually a check that mass renormalization is being done correctly.

All further work in this paper will be limited to solving 3+1 QED in the space $(e^-, e^-\gamma)$ or $(e^+e^-, e^+e^-\gamma)$. Solving the first case checks fermion mass renormalization; solving the second reproduces the first-order positronium spectrum plus some second-order corrections.

8. RENORMALIZATION: SELF-INDUCED INERTIAS AND MASS COUNTERTERMS

Two issues are of concern regarding renormalization. First is the question of the self-induced inertias that appear in the theory if one does not normal-order the light-cone Hamiltonian. The second is whether the light-cone perturbation theory results for the one-loop radiative corrections agree with the usual Feynman S-matrix answers. Let us investigate the first question.

If one begins with a Hamiltonian that is not normal-ordered and proceeds to normal-order, one finds extra terms arising from interchanging operators in the instantaneous photon and instantaneous fermion interactions. These terms have been referred to in the past as "self-induced inertias" (Refs. 1 and 2) and have been the source of much discussion concerning their role in light-cone physics. In 3+1 QED, these extra terms would take the form

$$\frac{2\alpha}{L_\perp^2}\sum_{\lambda,\underline{p}} a^\dagger_{\lambda,\underline{p}} a_{\lambda,\underline{p}} J_p \ , \quad J_p \ = \ \frac{1}{2p}\sum_{\underline{m}} \left[\{p-m \mid p-m\} - \{p+m \mid p+m\}\right] \tag{8.1}$$

for the photon and

$$\begin{aligned} &\frac{2\alpha}{L_\perp^2}\left[\sum_{s,\underline{n}} b^\dagger_{s,\underline{n}} b_{s,\underline{n}} (I_n + K_n) + d^\dagger_{s,\underline{n}} d_{s,\underline{n}} (I_n + M_n)\right] \ , \\ I_n \ &= \ \frac{1}{2}\sum_{\underline{m}} \{[n-m \mid n-m] - [n+m \mid n+m]\} \ , \\ K_n \ &= \ \frac{1}{2}\sum_{\underline{q}} \frac{1}{q}\{n-q \mid n-q\} \ , \\ M_n \ &= \ \frac{1}{2}\sum_{\underline{q}} \frac{1}{q}\{n+q \mid n+q\} \end{aligned} \tag{8.2}$$

for the fermion. Remember that for fermion anti-periodic boundary conditions and

photon periodic conditions,

$$p,q \;=\; 2,4,6,\ldots\,, \qquad m,n \;=\; 1,3,5,\ldots\,. \tag{8.3}$$

The photon's self-induced inertia J_p comes from interchanging fermion operators in the instantaneous fermion interaction $V_{instferm}$. I_n comes from interchanging fermion operators in the instantaneous photon interaction $V_{instphot}$ and K_n, M_n come from interchanging photon operators in the instantaneous fermion interaction. Note that the fermion self-induced inertias are not charge conjugate invariant because $K_n \neq M_n$.

The question then arises: should the self-induced inertias remain in the theory or should they be removed? Simply starting with a normal-ordered Hamiltonian eliminates these inertias. A satisfactory answer for the truncated Fock space we are considering is that they are *not* needed. In fact, a procedure that properly renormalizes the fermion mass in the truncated Fock space requires the addition of mass counterterms that are equal to the one-loop light-cone perturbation theory mass counterterms. This will be covered shortly. It even turns out that one may keep the self-induced inertias in the theory if one wishes, but they just get cancelled by an appropriate mass counterterm. Therefore, the self-induced inertias can be kept, but are unnecessary. Because they will be dropped or cancelled anyway, the problem of the self-induced inertias being not charge conjugate invariant is moot. Before continuing, it should be noted that this result, which will be detailed below, only holds in the truncated space $(e^+e^-, e^+e^-\gamma)$ or $(e^-, e^-\gamma)$. A more general procedure that includes higher Fock states may in fact require the presence of the self-induced inertias.

In our truncated Fock space, the full set of proper one-loop radiative corrections is shown in Figure 15 (improper graphs do not need to be renormalized). Again, there is no vacuum polarization because the Fock space does not allow an extra fermion pair to be created. Mass counterterms are needed to cancel these self-mass

diagrams. The discretized counterterms are

$$\delta H_{LC}^{(1)} = -$$

$$= K\,\frac{2\alpha}{L_\perp^2}\sum_{q,\vec{q}_\perp}\frac{\frac{1}{2n(n-q)}\left[n^2\left(\vec{q}_\perp-\frac{q}{n}\,\vec{n}_\perp\right)^2+q^2\beta_f\right]+\frac{n^2}{q^2}\left(\vec{q}_\perp-\frac{q}{n}\,\vec{n}_\perp\right)^2}{n^2\left(\vec{q}_\perp-\frac{q}{n}\,\vec{n}_\perp\right)^2+q^2\beta_f+n(n-q)\,\beta_\gamma} \tag{8.4}$$

and

$$\delta H_{LC}^{(2)} = -\sum_{N=2}^{\infty}$$

$$= -K\,\frac{\beta_f\,\pi^2}{n\,L_\perp^2}\;\frac{\left[\frac{\alpha}{\pi^2}\sum_{q,\vec{q}_\perp}\frac{q}{n^2\left(\vec{q}_\perp-\frac{q}{n}\,\vec{n}_\perp\right)^2+q^2\beta_f+n(n-q)\,\beta_\gamma}\right]^2}{1+\frac{\alpha}{\pi}\sum_{q,\vec{q}_\perp}\frac{n-q}{n^2\left(\vec{q}_\perp-\frac{q}{n}\,\vec{n}_\perp\right)^2+q^2\beta_f+n(n-q)\,\beta_\gamma}} \tag{8.5}$$

where

$$\begin{aligned}\beta_f=\left(\frac{m_e\,L_\perp}{\pi}\right)^2,\qquad \beta_\gamma=\left(\frac{\lambda\,L_\perp}{\pi}\right)^2,\qquad \alpha=\frac{e^2}{4\pi}\,,\\ n\;&=1,\,3,\,5,\,\ldots\qquad\text{(anti-periodic b.c.)}\,,\\ q\;&=2,\,4,\,6,\,\ldots\,,\\ n^i,q^i\;&=0,\,\pm1,\,\pm2,\,\ldots\,.\end{aligned} \tag{8.6}$$

$(n,\vec{n}_\perp)$ are the quantum numbers for the incoming fermion and λ is a small, fake photon mass. The sum is over $2\le q\le n-1$ and must satisfy both the ultraviolet and infrared cut-offs,

$$\begin{aligned}\frac{q_\perp^2+\beta_\gamma}{q}+\frac{(\vec{n}_\perp-\vec{q}_\perp)^2+\beta_f}{n-q}&\le\frac{1}{K}\left(\frac{\Lambda L_\perp}{\pi}\right)^2-\sum_{\text{spec}}\frac{m_\perp^2+\beta}{m}\,,\\ \frac{q_\perp^2+\beta_\gamma}{q}&\ge\frac{1}{K}\left(\frac{L_\perp}{\pi}\right)^2\epsilon\,.\end{aligned} \tag{8.7}$$

The sum in the last equation is over the quantum numbers $(m,\vec{m}_\perp)$ of all the spectator particles (i.e.: particles that go from the initial to final state without an interaction). The derivation of these results is given in Appendix D.

Inclusion of these mass counterterms and diagonalizing the space $(e^-, e^-\gamma)$ reproduces the real electron mass to be one to 12 significant figures on an IBM 3090 running 64-bit (double precision) real variables and thus verifies that this is indeed the correct fermion mass renormalization prescription. The numerical results are shown in Section 9. If self-induced inertias are retained, the mass counterterm is modified to include -(self-induced inertias). This just cancels the original inertias and diagonalizing again reproduces the real electron mass $= 1.000\ldots m_e$.

Now turn to the second question posed above, the equivalence of the mass counterterms derived from Feynman S-matrix theory and light-cone perturbation theory. A caveat must be made here: the comparison will actually be made between time-ordered perturbation theory in the infinite momentum frame (Ref. 11) (TOPTh_∞) and S-matrix theory rather than between light-cone perturbation theory (LCPTh) and S-matrix theory because the mathematics is easier to extract in TOPTh_∞. It is believed that LCPTh and TOPTh_∞ are mathematically equivalent, though conceptually different; therefore, it is likely that the following statements also hold in light-cone formulation. If not, one can for argument's sake consider that all considerations up to this point have actually been made in the infinite momentum frame and that we are attempting to diagonalize the infinite momentum frame 3+1 QED Hamiltonian. The comparison will also be made in Feynman gauge rather than light-cone gauge; again, it is believed that the basic result carries over.

The actual derivations for the mass counterterms are given in Appendix E; only the results are mentioned here. The fermion self-energy diagram shown in Figure 16 has the value

$$T_{fi} = -\frac{ig^2}{(2\pi)^4}\int d^4k\ \frac{\bar{u}(p)\gamma^\mu(\not{p}-\not{k}+m_e)\gamma_\mu u(p)}{[(p-k)^2-m_e^2+i\epsilon]\ (k^2-\lambda^2+i\epsilon)} \tag{8.8}$$

according to the Feynman rules in Feynman gauge. Doing the numerator algebra, combining denominators, shifting variables to $q = k - xp$, doing the q^0 integral by

contour integration, and finally doing the q^3 integral gives the result

$$T_{fi} = \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \frac{2m_e^2(1+x)}{k_\perp^2 + m_e^2 x^2 + \lambda^2(1-x) - i\epsilon} \quad . \tag{8.9}$$

Note that this result diverges like $\ln k_\perp^2$ for large $k_\perp^2$. A regulation scheme such as including a heavy, negative metric Pauli-Villars particle is needed to perform the remaining integrations.

The two graphs contributing to the fermion self-energy in time-ordered perturbation theory are shown in Figure 17. The expression for the first graph in regular time-ordered perturbation theory is

$$T_{fi}^{(1)} = \frac{g^2}{4(2\pi)^3} P \int_{-\infty}^{\infty} dx \int d^2\vec{k}_\perp \frac{1}{E_1 E_2} \frac{\bar{u}(p)\not{\epsilon} u(k_2)\ \bar{u}(k_2)\not{\epsilon}^* u(p)}{E - E_1 - E_2 + i\epsilon} \tag{8.10}$$

where momenta have been assigned as follows:

$$p = \left(E, \vec{0}_\perp, P\right) , \qquad k_1 = \left(E_1, \vec{k}_\perp, xP\right) , \qquad k_2 = \left(E_2, -\vec{k}_\perp, (1-x)P\right) ,$$

$$E = \sqrt{P^2 + m_e^2} , \qquad E_1 = \sqrt{x^2P^2 + \lambda_\perp^2} , \qquad E_2 = \sqrt{(1-x)^2P^2 + m_\perp^2} ,$$

$$\lambda_\perp^2 = k_\perp^2 + \lambda^2 , \qquad m_\perp^2 = k_\perp^2 + m_e^2 . \tag{8.11}$$

A heavy, negative metric Pauli-Villars particle must be subtracted to facilitate ultraviolet regulation. The correct method of evaluating this expression in the infinite momentum frame would be to do the x integral first and then let the observer's momentum go to infinity in the $-z$ direction by letting $P \to \infty$. In contrast, what is normally done is to take the limit $P \to \infty$ first and then do the x integral. The two are the same only if interchanging the limit and the integral

is allowed. Using the usual method (take limit first) gives the result

$$T_{fi}^{(1)} = \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \frac{1}{1-x} \quad \frac{(2-2x-2x^2)\,m_e^2 - k_\perp^2}{k_\perp^2 + x^2 m_e^2 + (1-x)\lambda^2 - i\epsilon} - (\lambda \to \Lambda) \quad (8.12)$$

which diverges like Λ^2 for large Λ. It turns out not to matter if one interchanges the limit and the x integral in this expression, so this answer is indeed the correct answer for this graph.

Now turn to the Z-graph contribution, which is

$$T_{fi}^{(2)} = \frac{g^2}{4(2\pi)^3} P \int_{-\infty}^{\infty} dx \int d^2\vec{k}_\perp \frac{1}{E_1 E_2} \frac{-\bar{u}(p)\not{\epsilon}^* v(k_2)\; \bar{v}(k_2)\not{\epsilon} u(p)}{-E - E_1 - E_2 + i\epsilon} ,$$

$$p = \Big(E, \vec{0}_\perp, P\Big) , \quad k_1 = \Big(E_1, \vec{k}_\perp, -xP\Big) , \quad k_2 = \Big(E_2, -\vec{k}_\perp, -(1-x)P\Big) ,$$

$$E = \sqrt{P^2 + m_e^2} , \quad E_1 = \sqrt{x^2 P^2 + \lambda_\perp^2} , \quad E_2 = \sqrt{(1-x)^2 P^2 + m_\perp^2} ,$$

$$\lambda_\perp^2 = k_\perp^2 + \lambda^2 , \quad m_\perp^2 = k_\perp^2 + m_e^2 \quad (8.13)$$

in time-ordered perturbation theory. Taking the limit $P \to \infty$ first gives an answer of zero. The correct method is to do the x integral first and interchange the limit and the integral only where allowed. Doing so, one finds a non-zero contribution to the Z-graph from the region x near zero leading to an answer of

$$\begin{aligned} T_{fi}^{(2)} &= \frac{g^2}{8\pi^3} \int d^2\vec{k}_\perp \log \frac{k_\perp^2 + \Lambda^2}{k_\perp^2 + \lambda^2} \\ &= \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \frac{-\lambda^2 + 2m_e^2\, x}{k_\perp^2 + m_e^2 x^2 + (1-x)\,\lambda^2 - i\epsilon} - (\lambda \to \Lambda) . \end{aligned} \quad (8.14)$$

Brodsky, Roskies and Suaya (Ref. 11) and Lepage and Brodsky (Ref. 14) present a rule for including backward moving particles in tree graphs. Naively applying

this rule to calculate the Z-graph gives an incorrect answer, showing that this rule should not be extended in this simple manner. Of course, their rule still holds for tree graphs.

If this result for the Z-graph is added to the result for the usual time ordering, Eq. (8.12), and one is added and subtracted from the integrand, one obtains

$$T_{fi}^{(1)} + T_{fi}^{(2)} = \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \, \frac{2m_e^2(1+x)}{k_\perp^2 + m_e^2 x^2 + (1-x)\lambda^2 - i\epsilon} - (\lambda \to \Lambda) \, . \quad (8.15)$$

which agrees with the Feynman answer, Eq. (8.9)! This demonstrates that the TOPTh$_\infty$ and Feynman rules results for the one-loop fermion self-energy in Feynman gauge are identical if one is careful to do the x integral first and interchange limit and integral only when allowed in the TOPTh$_\infty$ calculation. If one takes the limit first, one obtains Eq. (8.12) as the complete answer, which agrees with the usual LCPTh answer for the one-loop fermion self-energy, but disagrees with the Feynman answer. The discrepancy is found in a non-zero contribution from the Z-graph in TOPTh$_\infty$ near $x = 0$. In order to reconcile the LCPTh and Feynman rules answers for the one-loop fermion self-energy, an extra piece equal to the TOPTh$_\infty$ Z-graph must be added to the light-cone Hamiltonian and the LCPTh rules. However, since this piece is a self-mass, it is cancelled by including the appropriate mass counterterm. As a result, in practice the extra piece from the Z-graph can be ignored.

Notice that the usual time-ordering contribution in TOPTh$_\infty$, Eq. (8.12), diverges like Λ^2, but the full Feynman rules answer, Eq. (8.9), diverges like $\ln \Lambda$. Apparently, the leading Λ^2 divergence in the usual time-ordering graph is exactly cancelled by a similar divergence in the Z-graph.

A consideration much like the above can also be made for the vacuum polarization graph. Details are in Appendix E.

This completes the discussion of electron mass renormalization. Due to the absence of pair creation, there is no renormalization arising from vacuum polarization

in the truncated Fock space consideration. This leaves just electron wavefunction renormalization, which is equivalent to simply stating that the real electron's wavefunction is normalized. The probability of finding the bare Fock electron inside the real electron is given by the expansion coefficient ψ_{e^-} for the single electron Fock state shown in Eq. (4.28). This coefficient is just the wavefunction renormalization constant $\sqrt{Z_2}$.

To summarize, there is no photon wavefunction renormalization (charge renormalization) in the truncated Fock space, $(e^+e^-, e^+e^-\gamma)$ or $(e^-, e^-\gamma)$, because there is no allowance for pair creation. Electron wavefunction renormalization is automatic because the real electron's wavefunction is normalized. If one is careful about the behavior near the endpoints, $x = 0, 1$, the one-loop self-mass corrections in TOPTh$_\infty$ and probably LCPTH agree with the answer from S-matrix analysis. Mass renormalization is then done by inserting mass counterterms into H_{LC} that exactly cancel the one-loop self-mass contributions. If one decides to keep the "self-induced inertias", these are also cancelled by mass counterterms. Since the self-mass endpoint corrections and self-induced inertias are just cancelled anyway, what one effectively does is start with a normal-ordered Hamiltonian (i.e.: without self-induced inertias) and inserts the mass counterterms given in Eqs. (8.4) and (8.5) . Once again, this prescription is valid only in the truncated Fock space of one additional photon. If higher Fock states are included, a more general method is necessary which may in fact include the self-induced inertias in a crucial way.

Since only elementary particles require renormalization, no further renormalization needs to be done. That is, there is no positronium mass or wavefunction renormalization. The full light-cone Hamiltonian given by Eq. (4.24) plus mass counterterms given by Eqs. (8.4) and (8.5) is now ready to be diagonalized.

9. DIAGONALIZATION: CHARGE -1 SPACE

The prescription for diagonalizing the QED light-cone bound state equation Eq. (3.15) is then the following. H_{LC} is equal to $H_0 + H_1 + H_2 + H_{self}$ where H_0, H_1, and H_2 were given in Eq. (4.24) and H_{self} is the mass counterterms given in Eqs. (8.4) and (8.5). The Fock space is generated by keeping all Fock states that satisfy

$$\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} \leq \Lambda^2 \tag{9.1}$$

and have photons that satisfy

$$\frac{q_\perp^2}{x} \geq \epsilon \, . \tag{9.2}$$

These two cut-off conditions are also applied to the instantaneous fermion and photon interactions with the instantaneous particles treated as if they were real particles. Diagonalizing gives the full mass spectrum of bound states and scattering states and their corresponding wavefunctions as a Fock state expansion,

$$|\pi\rangle = \sum_n \psi_{n/\pi}(x, \vec{k}_\perp) \, |n\rangle \ . \tag{9.3}$$

In principle, the true continuum theory is reproduced by taking the limits $K, L_\perp, \Lambda \to \infty$ and $\epsilon \to 0$. Recall from Section 6 that the results are identical for any choice of ϵ less than $(\pi/L_\perp)^2$; therefore, one is allowed to take the limit $\epsilon \to 0$ immediately. In this paper, the Fock space is limited for various reasons discussed in Section 7 to just $(e^-, e^-\gamma)$ for charge -1 and $(e^+e^-, e^+e^-\gamma)$ for charge zero. Doing so checks fermion mass renormalization to one-loop and reproduces the Bethe-Salpeter ladder approximation for positronium.

Diagonalizing the light-cone Hamiltonian in the charge -1 space of $(e^-, e^-\gamma)$ for any value of α, K, $L_\perp$, Λ and ϵ reproduces

$$M^2 = 1.000\ldots m_e^2 \tag{9.4}$$

for the ground state. Remember that as pointed out in Section 7, in this truncated

Fock space consideration, diagram 14 must be dropped from the full set of light-cone diagrams in Fig. 7 . The accuracy of this result is only limited by machine precision. On an IBM 3090 running 64-bit real variables, this is 12 places behind the decimal point. This result numerically proves that fermion mass renormalization is being done correctly in the truncated space $(e^-, e^-\gamma)$ because the physical mass of the fermion (i.e.: the ground state mass, M) is equal to the bare fermion mass, m_e.

One also obtains the fermion's structure function by summing the ground state wavefunction over all modes with a fixed x,

$$f(x)dx = \sum_{n,\ \text{fixed}\ x} |\psi_{n/\pi}(x, \vec{k}_\perp)|^2 \ . \tag{9.5}$$

Typical structure functions for $\alpha = .3$ and .6 are shown in Figure 18 and Figure 19. As expected, the structure function is peaked at $x = 1$ and has a characteristic long radiative tail.

10. DIAGONALIZATION: CHARGE ZERO SPACE

A summary of the diagonalization prescription was given at the beginning of Section 9. Diagonalizing the truncated Fock space $(e^+e^-, e^+e^-\gamma)$ omitting diagrams 9, 14, 16 from Figs. 6 and 7 should return the Bohr answer

$$M = 2m_e - \frac{1}{4}m_e\left(\frac{Z\alpha}{n}\right)^2 \quad n = 1,2,3,\ldots \tag{10.1}$$

for the positronium bound state spectrum plus $L \cdot S$ coupling, the hyperfine splitting, and the part of the Lamb shift from the one-loop fermion self-energy diagram (actually, the muonium spectrum is returned since the annihilation channel has been removed in this truncated Fock space). We should also obtain the full spectrum and wavefunctions of scattering states, along with the bound states. The four lowest wavefunctions should be the one parapositronium and three orthopositronium states.

To give an example of the potential power of the method, a typical spectrum obtained from diagonalizing is shown in Figure 20. A 420 by 420 matrix was diagonalized on an IBM 3090 in 6 minutes to obtain this spectrum. Unfortunately, the number of Fock states is very limited by computer space ($\approx$ 500 states, maximum). The typical ground state wavefunction extends outward one point in the $\vec{k}_\perp$ direction and one or two points in the x direction. One hardly expects such a course ground state wavefunction to reproduce the correct mass eigenvalue. This is borne out by examining the convergence in K and $L_\perp$. Figure 21 and Figure 22 show no convergence in these parameters, or at most convergence to the free value $M^2 = 4m_e^2$.

One can estimate how large K, $L_\perp$, and Λ need to be for a given α by considering the expected ground state wavefunction,

$$\psi_{e^+e^-} = \frac{A}{\left[\frac{k_\perp^2+m_e^2}{x(1-x)} - M_{\text{Bohr}}^2\right]^2} \quad , \qquad M_{\text{Bohr}} = 2m_e - \frac{1}{4}m_e\alpha^2 \ . \tag{10.2}$$

This wavefunction has its peak at $\vec{k}_\perp = \vec{0}_\perp$, $x = \frac{1}{2}$ and falls to $\frac{1}{25}$ of its peak value

at

$$x = \frac{1}{2} , \quad k_x \text{ or } k_y = \pm m_e \alpha ; \qquad x = \frac{1}{2} \pm \frac{\alpha}{2} , \quad \vec{k}_\perp = \vec{0}_\perp . \tag{10.3}$$

If one wants the $n_\perp$th point in $\vec{k}_\perp$ and the nth point in x to lie at these points, then

$$\frac{n_\perp \pi}{L_\perp} = m_e \alpha , \qquad \frac{2n}{K} = \frac{\alpha}{2} . \tag{10.4}$$

One must still choose Λ large enough so that these points are actually included in the Fock space. One can estimate this Λ to occur when the electron and positron have $x = \frac{1}{2} - \frac{\alpha}{2}$, k_x or $k_y = \frac{1}{2} m_e \alpha$ and the photon $x = \alpha$, k_x or $k_y = -m_e \alpha$. This gives

$$\begin{aligned} \Lambda^2 &= \frac{(\frac{1}{2} m_e \alpha)^2 + m_e^2}{\frac{1}{2} - \frac{\alpha}{2}} + \frac{(\frac{1}{2} m_e \alpha)^2 + m_e^2}{\frac{1}{2} - \frac{\alpha}{2}} + \frac{(m_e \alpha)^2}{\alpha} \\ &= 4 m_e^2 \left(\frac{1 + \alpha/4}{1 - \alpha} \right) . \end{aligned} \tag{10.5}$$

Note that the largest contribution is from the last term, the photon's invariant mass squared. If one wants at least five points in each of the directions x, k_x, and k_y ($n, n_\perp = 5$ in Eq. (10.4)), one must choose K, $L_\perp$, and Λ to be at least

$$K = \frac{20}{\alpha} , \qquad L_\perp = \frac{5\pi}{m_e \alpha} , \qquad \Lambda = 2 m_e \sqrt{\frac{1 + \alpha/4}{1 - \alpha}} . \tag{10.6}$$

For $\alpha = .3$ and .6, the numerical values are

$$\begin{aligned} \alpha = .3 : \quad & K = 68 , \quad L_\perp = \frac{52}{m_e} , \quad \Lambda = 2.5 m_e , \\ \alpha = .6 : \quad & K = 34 , \quad L_\perp = \frac{26}{m_e} , \quad \Lambda = 3.4 m_e . \end{aligned} \tag{10.7}$$

These are far larger than the typical $K, L_\perp, \Lambda$ used above. The total number of Fock states for these values is $10,773,680$ for $K = 68$, $L_\perp = 52 \frac{1}{m_e}$, $\Lambda = 2.5 m_e$ and $8,362,468$ for $K = 34$, $L_\perp = 26 \frac{1}{m_e}$, $\Lambda = 3.4 m_e$.

That the method of diagonalizing the light-cone Hamiltonian to obtain bound state mass eigenvalues and wavefunctions works in principle is demonstrated in Sections 12 and 13. If one were able to diagonalize the large matrices necessary by either better numerical techniques, making the matrices smaller by theoretical considerations, or using a larger computer, one should be able to obtain the spectrum and wavefunctions for positronium.

11. VARIATIONAL METHOD

As seen in Section 10, diagonalizing the light-cone Hamiltonian directly to find mass eigenvalues and wavefunctions is limited numerically. The computer time and storage needed to handle the large matrices necessary to produce reasonable results are unyieldingly large using present techniques. A possible solution is to reduce the size of the matrices by implementing various symmetries that have been so far ignored. Possibilities include reflection symmetries, charge conjugation, and angular momentum. Another solution is to make use of the sparseness of the Hamiltonian matrix by implementing a Lanczos method or other algorithm to diagonalize.

In lieu of diagonalization, a third alternative is to do a variational calculation to find an upper bound on the mass eigenvalues and approximate the wavefunctions. A variational ansatz is made for the ground state wavefunction in the charge zero sector. For example, if one restricts to the truncated Fock space $(e^+e^-, e^+e^-\gamma)$, the expectation value of the light-cone Hamiltonian is calculated to find an upper limit on the ground state mass squared,

$$\langle\psi|\, H_{LC}\,|\psi\rangle = M^2 \geq M_0^2 \ , \tag{11.1}$$

with the variational wavefunction

$$|\psi\rangle = \psi_{e^+e^-}\left|e^+e^-\right\rangle + \psi_{e^+e^-\gamma}\left|e^+e^-\gamma\right\rangle \ . \tag{11.2}$$

$\psi_{e^+e^-}$ and $\psi_{e^+e^-\gamma}$ are functions of x and $\vec{k}_\perp$ as explained in the paragraphs before Eq. (3.11). H_{LC} is the expression derived in Section 4 (Eq. (4.24)) plus fermion mass renormalization terms (Eqs. (8.4) and (8.5)).

Here, we choose $\psi_{e^+e^-}$ to be

$$\psi_{e^+e^-} = \frac{A}{\left[\frac{k_\perp^2+v_1 m_e^2}{x(1-x)} - v_2 M_{\rm Bohr}^2\right]^{2v_3}} \tag{11.3}$$

for $s, s' = +1, +1$. $x = k^+/P^+$ and $\vec{k}_\perp$ are the electron's plus momentum fraction

and perpendicular momentum and s, s' are the electron and positron spins. v_1, v_2, and v_3 are variational parameters. This choice of $\psi_{e^+e^-}$ is from Brodsky and Ji[21] and is the relativistic extension of the Bohr result for the ground state (with v_1, v_2, and v_3 set equal to one),

$$\psi_{e^+e^-} = \frac{A'}{\left[\vec{k}^2 - \frac{1}{4}m_e^2\alpha^2\right]^2} \quad , \tag{11.4}$$

written in light-cone coordinates. $\psi_{e^+e^-\gamma}$ is chosen to be

$$\psi_{e^+e^-\gamma} = \sum_{(e^+e^-)} \frac{v_4}{v_5 M_{\rm Bohr}^2 - M_{e^+e^-\gamma}^2} \left\langle e^+e^-\gamma \right| H_{LC} \left| e^+e^- \right\rangle \psi_{e^+e^-} \ ,$$
$$M_{e^+e^-\gamma}^2 = \sum_{i=e^+,e^-,\gamma} \frac{k_{\perp i}^2 + m_i^2}{x_i} \ . \tag{11.5}$$

The meaning of this equation will be explained shortly. $M_{e^+e^-\gamma}^2$ is the invariant mass squared of the $(e^+e^-\gamma)$ state, $M_{\rm Bohr}$ is defined to be $2m_e - \frac{1}{4}m_e\alpha^2$, which is the non-relativistic Bohr answer for the ground state mass. A is a normalization constant chosen such that

$$\langle\psi|\psi\rangle = \sum_{x_i, \vec{k}_{\perp i}} |\psi_{e^+e^-}|^2 + |\psi_{e^+e^-\gamma}|^2 = 1 \ . \tag{11.6}$$

Such a choice of wavefunction should put a bound on the mass of the triplet $S = 1$ positronium state, orthopositronium (or more precisely, orthomuonium since the annihilation channel has been eliminated by restricting the Fock space to exclude pair creation). If the e^+e^- wavefunction had been chosen to be $+\psi_{e^+e^-}$ for $s, s' = +1, -1$ and $-\psi_{e^+e^-}$ for $s, s' = -1, +1$, a mass bound for the singlet $S = 0$ positronium state, parapositronium, would be found. We choose to work with orthopositronium for the numerical reason that it requires only the storage of spin up fermions, which reduces the computer storage requirement by $\frac{1}{4}$.

$\psi_{e^+e^-\gamma}$ for an electron, positron, photon Fock state with given quantum numbers is determined by looping over all (e^+e^-) states and summing all non-zero light-cone Hamiltonian matrix elements with a factor $v_4/(v_5 M^2_{\rm Bohr} - M^2_{e^+e^-\gamma})$. The first-order perturbation theory answer for $\psi_{e^+e^-\gamma}$ would be to multiply $\psi_{e^+e^-}$ by a factor of $\langle e^+e^-\gamma|P^-|e^+e^-\rangle/D$ where D is the light-cone energy denominator (i.e.: difference of P^-s). Since $H_{LC} = P^+P^-$ and $M^2_{\rm Bohr} - M^2_{e^+e^-\gamma} = P^+D$, we see that our choice of $\psi_{e^+e^-\gamma}$ is just what one would write down from perturbation theory (with v_4 and v_5 set to one).

Excited states could in principle be calculated by doing a variational calculation with variational wavefunctions chosen orthogonal to the ground state and any other lower states.

An analagous construction can be made in the charge -1 sector. A variational wavefunction is chosen,

$$|\psi\rangle = \psi_e^- \left|e^-\right\rangle + \psi_{e^-\gamma} \left|e^-\gamma\right\rangle \ , \tag{11.7}$$

with

$$\psi_e^- = A\ \delta(x-1)\delta^{(2)}(\vec{k}_\perp) \tag{11.8}$$

for $s = +1$ and

$$\begin{aligned} \psi_{e^-\gamma} =& \frac{v_1}{v_2 m_e^2 - M^2_{e^-\gamma}} \left\langle e^-\gamma\right| H_{LC} \left|e^-\right\rangle \psi_e^- \ , \\ M^2_{e^-\gamma} =& \sum_{i=e^-,\gamma} \frac{k^2_{\perp i} + m_i^2}{x_i} \ . \end{aligned} \tag{11.9}$$

Again, v_1 and v_2 are variational parameters and A is the normalization constant. Calculating the expectation value H_{LC} returns an upper bound on the real electron's mass, M, in terms of the bare electron's mass, m_e:

$$\langle\psi| H_{LC} |\psi\rangle = M^2 \geq m_e^2 \ . \tag{11.10}$$

If fermion mass renormalization has been done properly, M should equal m_e.

That the variational equation,

$$\langle\psi|\, H_{LC}\, |\psi\rangle = M^2 \ , \tag{11.11}$$

is equivalent the momentum space Schroedinger equation for positronium (muonium),

$$\int d^3\vec{k}\ |\psi(\vec{k})|^2\ \frac{\vec{k}^2}{m_e}\ -\ \frac{\alpha}{2\pi^2}\int d^3\vec{k}_i\ d^3\vec{k}_f\ \psi^*(\vec{k}_f)\psi(\vec{k}_i)\ \frac{1}{(\vec{k}_f-\vec{k}_i)^2}\ =\ -\frac{1}{4}m_e\alpha^2\ , \tag{11.12}$$

in the charge zero sector and

$$M = m_e \tag{11.13}$$

in the charge -1 sector in the non-relativistic limit is demonstrated in Appendix F and Appendix G.

12. VARIATIONAL METHOD: CHARGE -1 SPACE

The variational calculation described in Section 11 is applied here to the truncated Fock space $(e^-, e^-\gamma)$. Recall that the wavefunction is chosen to be

$$|\psi\rangle = \psi_e^- |e^-\rangle + \psi_{e^-\gamma} |e^-\gamma\rangle \ ,$$
$$\psi_e^- = A\ \delta(x-1)\delta^{(2)}(\vec{k}_\perp) \ , \qquad \psi_{e^-\gamma} = \frac{v_1}{v_2 m_e^2 - M_{e^-\gamma}^2} \langle e^-\gamma | H_{LC} |e^-\rangle \psi_e^- \ . \tag{12.1}$$

v_1 and v_2 are variational parameters. The expectation value of the light-cone Hamiltonian

$$\langle\psi| H_{LC} |\psi\rangle = M^2 \geq m_e^2 \tag{12.2}$$

is calculated to obtain an upper limit on the physical mass of the electron. H_{LC} is the sum of Eq. (4.24) and the mass counterterms Eqs. (8.4) and (8.5), but excludes diagram 14 (see Fig. 7) since we are working with a truncated Fock space.

Note from Appendix G that if one drops the instantaneous fermion interaction $V_{instferm}$ and its associated mass counterterm, Eq. (8.5), M^2 is equal to m_e^2 for all α, K, $L_\perp$, and Λ with $v_1, v_2 = 1$. This is borne out numerically: for every set of α, K, $L_\perp$, and Λ investigated, we obtained

$$M^2 = 1.000\ldots m_e^2 \tag{12.3}$$

to 10 places behind the decimal point.

If $V_{instferm}$ is retained, the expectation value is not minimum at $v_1, v_2 = 1$. Varying the parameters for $\alpha = .6$, $K = 25$, $L_\perp = 12\frac{1}{m_e}$, $\Lambda = 3.5 m_e$, one finds the minimum at $v_1 = .99934$, $v_2 = .99834$, at which point the expectation value is

$$M^2 = 1.0002431502\ m_e^2 \ . \tag{12.4}$$

The wavefunctions for this case is shown in Figure 23. It shows the expected photon cloud around the bare electron at $x = 1$, $\vec{k}_\perp = \vec{0}_\perp$.

Thus, the result from diagonalizing the charge -1 space, $M = m_e$, has been reproduced as a variational calculation, once again demonstrating that choosing the mass counterterms given in Section 8, Eqs. (8.4) and (8.5), is the correct renormalization prescription to keep the fermion's bare mass equal to its physical mass. Since there is no further renormalization arising from composite objects such as positronium (i.e.: no positronium "mass renormalization" or "wavefunction renormalization"), we may carry these mass counterterms over to the charge zero sector and proceed to calculate for positronium.

13. VARIATIONAL METHOD: CHARGE ZERO SPACE

As described in Section 11, the expectation value

$$\langle\psi|\, H_{LC}\, |\psi\rangle = M^2 \geq M_0^2 \tag{13.1}$$

of the light-cone Hamiltonian given in Eq. (4.24) plus mass counterterms Eqs. (8.4) and (8.5) is calculated in this section in the truncated Fock space $(e^+e^-, e^+e^-\gamma)$ with wavefunction

$$\begin{aligned} |\psi\rangle &= \psi_{e^+e^-}\,|e^+e^-\rangle + \psi_{e^+e^-\gamma}\,|e^+e^-\gamma\rangle\ , \\ \psi_{e^+e^-} &= \frac{A}{\left[\frac{k_\perp^2+v_1 m_e^2}{x(1-x)} - v_2 M_{\rm Bohr}^2\right]^{2v_3}}\ , \\ \psi_{e^+e^-\gamma} &= \sum_{(e^+e^-)} \frac{v_4}{v_5 M_{\rm Bohr}^2 - M_{e^+e^-\gamma}^2}\,\langle e^+e^-\gamma|\, H_{LC}\, |e^+e^-\rangle\,\psi_{e^+e^-} \end{aligned} \tag{13.2}$$

to obtain an upper limit on the ground state of positronium. In this section, the variational parameters v_1 through v_5 are set equal to one. As explained in Section 7, since the Fock space is truncated, diagrams 9, 14, and 16 from Figs. 6 and 7 must be dropped from H_{LC}. Diagram 10 must also be dropped when another photon is present. All Fock states are required to have invariant mass squared less than Λ^2,

$$\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} \leq \Lambda^2\ , \tag{13.3}$$

and photon invariant mass squared greater than ϵ,

$$\frac{q_\perp^2}{x} \geq \epsilon\ . \tag{13.4}$$

In these equations, instantaneous particles are treated as if they were real particles. Also, as explained in Section 6, the limit $\epsilon \to 0$ can be taken immediately. Numerically, this means ϵ can be chosen equal to the computer's machine precision. For

computational reasons, the spins of the fermions and anti-fermions are all chosen to be up; therefore, the bound will be placed on the $1s^{3/2}$ level, orthopositronium. Extending to include all spin states is a simple matter given enough computer capabilities.

First, we demonstrate numerically the need for an infrared cut-off on the photon states. This contention was made in Section 6 and requires all photons to satisfy Eq. (13.4). Figure 24 and Figure 25 show the behavior in K for $\alpha = .6$, $L_\perp = 20\frac{1}{m_e}$, $\Lambda = 2.4m_e$ with and without an infrared cut-off, respectively. KE and PE are defined in Appendix F, Eq. (F.23). Quite clearly, the latter case does not converge. As mentioned in Section 6, this lack of convergence is due to the discrete sum placing too much weight on the points near $x_\gamma = 0$, $\vec{k}_{\perp\gamma} = \vec{0}_\perp$. Barring a better solution, the points $\vec{k}_{\perp\gamma} = \vec{0}_\perp$ must be removed by a cut-off such as Eq. (13.4) to give convergence as $K \to \infty$.

Before continuing, we summarize some results first shown in Section 10. Note that most of the (e^+e^-) wavefunction given in Eq. (13.2) occurs for electrons with x between $\frac{1}{2}-\frac{\alpha}{2}$ and $\frac{1}{2}+\frac{\alpha}{2}$ and k_x, k_y between $-m_e\alpha$ and $+m_e\alpha$. The wavefunction has fallen to 1/25 of its peak value at these points. The Coulomb binding is mainly due to electrons inside this binding region. If one wants $2n$ points in x and $2n_\perp$ points in each of k_x and k_y to lie inside this region, one must choose K and $L_\perp$ to be at least

$$K = \frac{4n}{\alpha}\,, \qquad L_\perp = \frac{n_\perp \pi}{m_e \alpha}\,. \tag{13.5}$$

To ensure that all electrons and positrons inside the binding region are indeed kept, one needs Λ at least

$$\Lambda = 2m_e\sqrt{\frac{1+\alpha/4}{1-\alpha}}\,, \tag{13.6}$$

which for $\alpha = .6$ is $\Lambda = 3.4m_e$.

Also note that the instantaneous fermion interaction, $V_{instferm}$, contributes only a small amount to the expectation value $\langle\psi|\, H_{LC}\,|\psi\rangle$. For example, at $\alpha = .6$,

$K = 26$, $L_\perp = 20\frac{1}{m_e}$, $\Lambda = 2.8 m_e$,

$$\begin{aligned} \langle\psi|\, V_{instferm} \,|\psi\rangle &= 0.001196\ m_e^2\ , \\ M^2 = \langle\psi|\, H_{LC} \,|\psi\rangle &= 4.076754\ m_e^2\ , \\ PE &= -.129765\ m_e^2\ . \end{aligned} \tag{13.7}$$

PE is defined in Appendix F, Eq. (F.23) and is approximately the contribution of the potential energy to M^2. The instantaneous fermion contribution is only .03 percent of M^2 and .92 percent of PE. We therefore choose to ignore $V_{instferm}$ in all further deliberations since the computer time needed to calculate just this contribution turns out to be far greater than the time needed to calculate everything else. For consistency, the mass counterterm Eq. (8.5) must also then be removed.

So, dropping the instantaneous fermion interaction and counterterm Eq. (8.5) and calculating $M^2 = \langle\psi|\, H_{LC} \,|\psi\rangle$ for $\alpha = .6$ gives the results shown in Figure 26, Figure 27, and Figure 28. The graphs plot the quantities M^2, KE, PE, PE_{flip}, PE_{noflip}, and $PE_{instphot}$ in units of m_e^2. M^2 is equal to $4m_e^2 + KE + PE$ and $PE = PE_{flip} + PE_{noflip} + PE_{instphot}$. KE, PE, PE_{flip}, PE_{noflip} and $PE_{instphot}$ are defined in Appendix F, Eq. (F.23) and are approximately the contribution of the kinetic energy to M^2, the potential energy to M^2, the contribution of V_{flip} to PE, the contribution of V_{noflip} to PE and the contribution of $V_{instphot}$ to PE. PE_{flip} is actually zero in our case because we have chosen to keep only fermions and anti-fermions with spin up.

Note that M^2, KE, and PE have the following numerical values for a pure Bohr spectrum at $\alpha = .6$:

$$E_{\text{Bohr}} = -\frac{1}{4} m_e \alpha^2\ , \qquad KE_{\text{Bohr}} = \frac{1}{4} m_e \alpha^2\ , \qquad PE_{\text{Bohr}} = -\frac{1}{2} m_e \alpha^2\ ,$$

$$\begin{aligned} M^2 &= (2m_e + E_{\text{Bohr}})^2 = 3.648\ m_e^2\ , \\ KE &\approx (2m_e + KE_{\text{Bohr}})^2 - 4m_e^2 = .368\ m_e^2\ , \\ PE &\approx (2m_e + PE_{\text{Bohr}})^2 - 4m_e^2 = -.688\ m_e^2\ . \end{aligned} \tag{13.8}$$

The cross term between the kinetic and potential energies of order $\frac{1}{4}\alpha^4 = .03$

has been dropped to obtain the last two numbers. We see from the figures that KE converges to the Bohr answer rapidly, whereas PE converges rather slowly. Also note that PE_{noflip} and $PE_{instphot}$ are individually infrared and ultraviolet divergent – they each diverge as $K \to \infty$ and $\Lambda \to \infty$ – but their sum, PE, is convergent. This cancellation is extremely sensitive, and is destroyed by increasing or decreasing the strength of $V_{instphot}$ relative to V_{noflip} by the smallest amount. This delicate cancellation increases our confidence that the correct form of the potential has been used in H_{LC}. The cancellation also fails if the infrared regulator, Eq. (13.4), is removed. A typical wavefunction for $\alpha = .6$, $K = 42$, $L_\perp = 32\frac{1}{m_e}$, $\Lambda = 2.5 m_e$ is exhibited in Figure 29.

Unfortunately, we find that our computer resources are exhausted before reasonable answers are realized. For example, the rightmost point in Fig. 30 is barely bound,

$$M^2 = 3.9984843235\ m_e^2\ , \tag{13.9}$$

but requires 1,621,435 Fock states and approximately 11 Cray YMP service units* to calculate.

However, we show in Appendix F that $\langle\psi|\, H_{LC}\,|\psi\rangle$ is approximately equal to

$$\sum_{x,\vec{k}_\perp} |\psi_{e^+e^-}|^2\, \frac{k_\perp^2 + m_e^2}{x(1-x)} + \sum_{x_i,x_f,\vec{k}_{\perp i},\vec{k}_{\perp f}} \frac{4K\pi}{L\Omega} \psi^{f\,*}_{e^+e^-} \psi^{i}_{e^+e^-} \frac{e^2}{q^2_{FR}} \tag{13.10}$$

where

$$\psi_{e^+e^-} = \frac{A}{\left[\frac{k_\perp^2+m_e^2}{x(1-x)} - M^2_{\rm Bohr}\right]^2} \tag{13.11}$$

and the normalization A is determined by

$$\sum_{x,\vec{k}_\perp} |\psi_{e^+e^-}|^2 = 1\ . \tag{13.12}$$

---

* One service unit at the Pittsburgh Supercomputing Center is approximately .75 CPU hours or .375 MWordhours of CPU memory. One Word is the storage needed for one double precision word.

q^2_{FR} is given in Eq. (F.25) in Appendix F. The set of points in Fig. 30 labelled "C" show the value of this quantity as a function of Λ. These data points will be referred to from now on as "Coulomb data". The points labelled "V" are obtained by calculating the expectation value $\langle\psi|\, H_{LC}\,|\psi\rangle$ and will be referred to as "light-cone data". The closeness of the two sets of data points in Fig. 30 demonstrates numerically that the approximation Eq. (13.10) for $\langle\psi|\, H_{LC}\,|\psi\rangle$ is justifiable. One can therefore reasonably believe that the light-cone data points, if ever calculated, will follow the Coulomb data points as $K, L_\perp, \Lambda$ are increased.

Now turn to a consideration of Coulomb data. From Figs. 31 and 32 we see that M^2 converges like $1/L_\perp^2$ for reasonably large $L_\perp$ and like $1/K$ for reasonably large K (points at smaller values of K, $L_\perp$ have been omitted from the plots and are more erratic due to the smallness of K and $L_\perp$). Fitting the data to the form $M^2 = A(1 + B/L_\perp^2)(1 + C/K)$ gives

$$M^2 \;=\; 4.000\; m_e^2 \;-\; .242\left(1 - \frac{453}{L_\perp^2}\right)\left(1 - \frac{31.1}{K}\right) m_e^2\,. \tag{13.13}$$

The upper bound placed on orthopositronium from this fit is

$$M^2 = 3.758\; m_e^2\,, \tag{13.14}$$

which should be compared to the pure Bohr answer

$$M^2 = 3.648\; m_e^2\,. \tag{13.15}$$

The data used to produce these fits is in Appendix I. Due to the smallness of Λ, we have been unable to fit the data to this parameter. One would expect M^2 to fall off as $A + B/\Lambda^6$ because PE is proportional to $\int d^2k_{\perp i} d^2k_{\perp f} \frac{1}{k_{\perp i}^4 k_{\perp f}^4 (\vec{k}_{\perp i} - \vec{k}_{\perp f})^2} \to \frac{1}{\Lambda^6}$ for large Λ (see Eq. (F.29) in Appendix F).

We see that the M^2 value obtained from fitting Coulomb data in K and $L_\perp$ with $V_{instferm}$ ignored gives a bound on orthopositronium of $M^2 = 3.758 m_e^2$. This result

is at least within the ballpark from the true answer at $\alpha = .6$ which is estimated from the Bohr formula to be $M^2 = 3.648 m_e^2$. This answer can be improved by running at larger Λ and then fitting in this parameter. There is also room for improvement from varying the variational parameters v_1 through v_5. Restoring $V_{instferm}$ should change the answer by only a small amount.

One can fairly confidently say that the light-cone variational method does a reasonable job of reproducing the orthopositronium state. If one had the computer resources necessary, the correct answer can most likely be obtained without approximating $\langle\psi| H_{LC} |\psi\rangle$ by Eq. (13.10) and dropping $V_{instferm}$.

14. SUMMARY

Discretized Light-Cone Quantization (DLCQ) has been presented as a fully relativistic discrete representation of quantum field theories and has been demonstrated to work in principle for Quantum Electrodynamics in three space and one time dimensions. Covariant, (tree-level) gauge invariant ultraviolet and infrared regulation were presented in Sections 5 and 6 and a complete renormalization scheme in the truncated Fock space of $(e^-, e^-\gamma)$ or $(e^+e^-, e^+e^-\gamma)$ was outlined in Section 8. The numerical proof of the renormalization method is the demonstration that the electron's bare mass is equal to its physical mass using diagonalization or a variational calculation. These were shown in Sections 9 and 12. Most of the positronium spectrum is contained in this truncated Fock space: the Bohr levels, $L \cdot S$ coupling, the hyperfine interaction, and the part of the Lamb shift from the fermion self-energy diagram are all included (the results obtained in this truncated Fock space will actually be for muonium because the annihilation potential is not present).

The best numerical result to date for the $1s^{3/2}$ state of positronium, orthopositronium, is an upper bound for $\alpha = .6$ of

$$M_0^2 \leq 3.758\ m_e^2\ . \tag{14.1}$$

This result is from fitting what was described as "Coulomb data" in Section 13 and extrapolating in the parameters K and $L_\perp$. It compares with an estimate of the true value using the Bohr formula of $M_0^2 = (2m_e - \frac{1}{4}m_e\alpha^2)^2 = 3.648m_e^2$. Two approximations are made in this result: $\langle\psi|\, H_{LC}\, |\psi\rangle$ is approximated by Eq. (13.10) and the instantaneous fermion interaction is dropped. The validity of the first approximation is demonstrated mathematically in Appendix F and numerically in Section 13. The second approximation is shown numerically in Section 13 to have only a very small effect on the answers. Both approximations are not fundamental and are done only to reduce the amount of computer resources needed to do the calculations. Given enough computer time and memory, the bound of $3.758m_e^2$ can

be reproduced without these approximations. Calculating the $1s^{1/2}$ level, para-positronium, and finding the hyperfine splitting can be done by including all spin states for the electron and positron. This can be done with no modifications to the method described, only more computer resources or better numerical technology to store the added electron and positron spin states is needed.

The success of the variational calculation has tested the foundations of DLCQ and shown them to be sound. There should be no fundamental reason why the light-cone bound state equation

$$H_{LC} |\psi\rangle = M^2 |\psi\rangle \tag{14.2}$$

can not be diagonalized to obtain the mass spectrum and wavefunctions in the Fock space of $(e^+e^-, e^+e^-\gamma)$. A series of numerical approximations were made to actually solve Eq. (14.2) . These were to replace (14.2) by a variational calculation,

$$\langle\psi| H_{LC} |\psi\rangle = M^2 \geq M_0^2 \ , \tag{14.3}$$

drop $V_{instferm}$, and to replace $\langle\psi| H_{LC} |\psi\rangle$ by Eq. (13.10). Again, the only reason these approximations were made is lack of numerical technology or computer facilities. There is no theoretical barrier to directly solving (14.2). An estimate of the number of Fock states needed for reasonable answers using the methods described in this paper is 9,444,569 for $\alpha = .6$, $K = 42$, $L_\perp = 32\frac{1}{m_e}$, $\Lambda = 3.5m_e$.

A possible method of extending to include the Fock state with two photons, $(e^+e^-\gamma\gamma)$, is to include mass counterterms for the fermion self-mass diagrams with two photons in flight. A subset of these are shown in Figure 33. Including this Fock state with two photons should reproduce the full Lamb shift excluding the Uehling term from vacuum polarization. The Uehling term can be included by further extending the Fock state to include $(e^+e^-e^+e^-)$. This extension can be implemented by introducing photon mass counterterms for the graphs in Figure 34. As explained in Appendix E, photon mass counterterms are necessary because we

are using a non-subtractive ultraviolet regulation scheme. A test of whether this is done correctly is to check that the ground state has $M^2 = 0$. This would verify that the bare photon mass remains equal to the physical photon mass. Including this extra Fock state also puts back the annihilation potential needed to calculate true positronium levels.

Possible methods of improving the numerical technology that deserve further consideration include:

1. Implementing symmetries that have been so far ignored. These might include angular momentum, charge conjugation, and so forth. Choosing the Fock states to be eigenstates of these operators would dramatically reduce their number.

2. Using a Monte Carlo method to calculate the expectation value $\langle\psi| H_{LC} |\psi\rangle$ in the framework of a variational calculation. Doing so may eliminate the need to store all the Fock states in computer memory simultaneously, thus allowing the consideration of many more states than is now permitted.

3. Using a numerical method such as the Lanczos method to take advantage of the sparseness of the light-cone Hamiltonian matrix. Such a diagonalization routine would reduce the amount of computer time and memory needed since the whole matrix would not have to be stored at once as is done now.

One or more of these improvements may yet provide the numerical accuracy needed to make calculations of positronium to higher precision.

A new ultraviolet cut-off method proposed by Pauli[22] has shown much promise in this area. His suggestion is to only keep Fock states that satisfy the condition

$$\begin{aligned} \sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} - M_{\text{min}}^2 &\leq \Lambda^2 \,, \\ M_{\text{min}}^2 &= \min\left[\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i}\right] . \end{aligned} \tag{14.4}$$

The first term on the left-hand side is a sum over all the constituent particles in the

state. The second term is the minimum value that this sum can take for that type of Fock state. That is, there is one value of $M^2_{\rm min}$ for all Fock states containing an electron and a positron, one value for all Fock states containing an electron, a positron, and a photon, and so on. For example, $M^2_{\rm min}$ is equal to $4m_e^2$ for (e^+e^-) Fock states (this occurs at $\vec{k}_\perp = \vec{0}_\perp$, $x = \frac{1}{2}$) and $16m_e^2$ for $(e^+e^-e^+e^-)$ Fock states (at $\vec{k}_\perp = \vec{0}_\perp$, $x = \frac{1}{4}$). In order to maintain gauge invariance at the tree-level, this cut-off is also applied to states with instantaneous particles in the same manner as explained in Section 5. Work is now in progress with this new cut-off and is showing signs of much improved convergence properties.

The method of DLCQ has a number of important positive attributes:

1. The technique is straightforward, non-perturbative, and fully relativistic, and can be applied to quantum field theories in general, the most obvious candidate being Quantum Chromodynamics. Even the truncated Fock space analysis is non-perturbative since the Fock states that are allowed are iterated an infinite number of times.

2. Due to the positivity of P^+, there are no interactions in the theory that create particles out of the vacuum. As a result, the vacuum structure is simple: the perturbative vacuum is the Fock state vacuum is the true vacuum, and they are all eigenstates of H_{LC} with $M^2 = 0$.

3. Diagonalization has the potential of giving the full spectrum of bound states and scattering states along with their respective wavefunctions. The structure functions needed in calculations of high-energy scattering processes are obtained from the wavefunctions simply,

$$f(x)dx = \sum_{n,\ \text{fixed}\ x} |\psi_{n/\pi}(x, \vec{k}_\perp)|^2 \ . \tag{14.5}$$

4. The fermions are treated in a natural way. There are no fermion determinants or fermion doubling.

5. In $A^+ = 0$ gauge, there are only two physical photons.

6. As shown in Appendix F, DLCQ is equivalent to the momentum space Schroedinger equation in the non-relativistic limit.

To emphasize once again, the framework for DLCQ has been established for quantum field theories in three space and one time dimensions in a truncated Fock space. No further theoretical considerations need to be made; better results are a matter of improved numerical technology and computer resources. Pauli's new ultraviolet cut-off is a promising avenue of hope in this regard. Extensions to other field theories such as Quantum Chromodynamics should now be possible. Though the numerical results presented here are not as good as one might like, hopefully the appetite has been whetted.

APPENDIX A

In this appendix, Dirac's method for quantizing constrained Hamiltonian systems such as QED is briefly described and then applied to space-time QED in temporal gauge ($A^0 = 0$) and light-cone QED in light-cone gauge ($A^+ = 0$). A more complete discussion can be found in Dirac[23] and Hanson, Regge and Teitelboim (Ref. 20). Hanson et al also present specific applications of the method to space-time QED in a variety of gauges. Steinhardt[24] discusses the application of Dirac's method to light-cone QED. Much of the application below to QED is from Hanson et al and Steinhardt.

The general method is as follows:

1. One finds the canonical momenta $p^n = \frac{\partial L}{\partial \dot{q}_n}$ from the Lagrangian $L(q_n, \dot{q}_n)$. This may lead to a number of constraint equations relating q_n and p^n (i.e.: equations that are independent of $\dot{q}_n$). These equations are referred to as primary constraints,

$$\phi_m(p,q) \approx 0 \ , \qquad m = 1, \ldots M \ . \tag{A.1}$$

M is the number of primary constraint equations. The wiggly equal sign means that the equation is a weak equality because the Poisson bracket of ϕ_m with some of the canonical variables may not equal zero. The normal equal sign will be used to denoted strong equalities which have zero Poisson brackets with all the canonical quantities. Assume Poisson bracket relations between the p^n and q_n,

$$\{p^n, q_m\} = -\delta^n_m \ . \tag{A.2}$$

The Poisson bracket is defined to be

$$\{A, B\} = \sum_i \frac{\partial A}{\partial q_i}\frac{\partial B}{\partial p^i} - \frac{\partial A}{\partial p^i}\frac{\partial B}{\partial q_i} \ . \tag{A.3}$$

2. The canonical Hamiltonian H_c is derived from the Lagrangian as

$$H_c = p^n \dot{q}_n - L(q_n, \dot{q}_n) \ . \tag{A.4}$$

This Hamiltonian is not unique because any multiple of one of the primary constraints can be added to give

$$\tilde{H} = H_c + u_m \phi_m(p, q) \ . \tag{A.5}$$

The equations of motion can be generated by taking the Poisson bracket with $\tilde{H}$,

$$\frac{\partial g}{\partial t} = \{g, \tilde{H}\} \approx \{g, H_c\} + u_m \{g, \phi_m\} \ . \tag{A.6}$$

To have a consistent system, the primary constraints must stay zero. Therefore, $\dot{\phi} = \frac{\partial \phi}{\partial t}$ must equal zero. That is,

$$\dot{\phi}_n \approx \{\phi_n, H_c\} + u_m \{\phi_n, \phi_m\} \approx 0 \ . \tag{A.7}$$

This new set of equations can lead to one of four outcomes.

(a) The result may be an inconsistency. If this is so, the Lagrangian is no good.

(b) These equations may provide no new information.

(c) They may result in conditions on the coefficients u_m.

(d) The equations may cause a new condition (independent of the u_m's) on the p^n, q_n. These are secondary constraints and are collectively denoted

$$\phi_a(p, q) \approx 0 \ , \qquad a = M + 1, \ldots T \ . \tag{A.8}$$

One then requires $\dot{\phi}_a \approx 0$. This may lead to further secondary constraints. This procedure is continued until all secondary constraints are found. T is the total number of constraints, primary and secondary.

3. Define first and second class constraints. If

$$\{\phi_a, \phi_b\} \approx 0 \tag{A.9}$$

for all $b = 1, \ldots, T$, then ϕ_a is first class. Otherwise ϕ_a is second class. The first class constraints are collectively denoted

$$\psi_i(p,q) \approx 0 \ , \qquad i = 1, \ldots I \ . \tag{A.10}$$

I is the number of first class constraints.

4. The number of first class constraints is equal to the number of gauge degrees of freedom, which are eliminated by imposing gauge conditions as secondary constraints. These gauge secondary conditions may generate more secondary constraints as in point 2. The total number of gauge conditions,

$$\gamma_i(p,q) \approx 0 \ , \qquad i = 1, \ldots I \ , \tag{A.11}$$

should equal the number of first class constraints. Upon imposing these gauge conditions, all constraints become second class.

5. Now form the matrix

$$C_{\alpha\beta} = \{\phi_\alpha, \phi_\beta\} \tag{A.12}$$

where ϕ_a, ϕ_b run over all the second class constraints. This matrix is inverted to give $C^{-1}_{\alpha\beta}$. Dirac has shown that if $\{\gamma_i, \psi_i\}$ is well defined and non-singular, then $C^{-1}_{\alpha\beta}$ exists.

6. Replace all Poisson brackets by the Dirac bracket,

$$\{A, B\}^* = \{A, B\} - \{A, \phi_\alpha\} C^{-1}_{\alpha\beta} \{\phi_\beta, B\} \ . \tag{A.13}$$

One can show that $\{\phi_\alpha, A\}^* = 0$ for all second class constraints. As a result, all second class constraint equations (or all constraints if a full set of gauge constraints has been imposed) can now be set strongly equal to zero.

7. The total Hamiltonian is taken to be

$$H = H_c + v_i \psi_i(p,q) \tag{A.14}$$

where the sum is over any remaining first class constraints. Since the second class conditions $\phi_\alpha = 0$ are now strong equalities, they can be used in the right hand side of Eq. (A.14). If I gauge constraints had been imposed, the total Hamiltonian would just be H_c. The new equations of motion are

$$\frac{\partial g}{\partial t} = \{g, H\}^* \tag{A.15}$$

and the Poisson bracket relations are given by $\{A, B\}^*$.

8. The system is quantized by replacing $\{A, B\}^*$ by $-i\,[A, B]$.

As a first example of Dirac's method, consider ordinary space-time QED in temporal gauge ($A^0 = 0$). In just this example, the metric tensor $g^{\mu\nu}$ will be chosen to have diagonal elements $(-1, 1, 1, 1)$. In the rest of this paper, $g^{\mu\nu}$ has diagonal elements $(1, -1, -1, -1)$. The canonical momenta are derived from the Lagrangian to be

$$\pi^0 = 0\ , \qquad \vec{\pi} = \dot{\vec{A}} - \vec{\nabla} A^0\ , \tag{A.16}$$

from which one obtains the canonical Hamiltonian

$$H_c = \int d^3\vec{x} \left[\frac{1}{2}\vec{\pi}^2 + \frac{1}{2}\vec{B}^2 - \vec{\pi} \cdot \vec{\nabla} A^0 \right]\ . \tag{A.17}$$

The relationship for π^0 is a primary constraint since it does not involve the velocity fields $\dot{A}^\mu$. This primary constraint turns out to give one secondary constraint, $\vec{\nabla} \cdot \vec{\pi} \approx 0$. These two constraints are both first class, which means that a total of two gauge conditions may be chosen. The temporal gauge condition $A^0 \approx 0$ leads

to a secondary gauge constraint $\vec{\nabla} \cdot \vec{A} \approx 0$. Collecting the constraints, which are now all second class,

$$\begin{aligned} \textit{primary constraint}: &\quad \pi^0 \approx 0 \\ \textit{secondary constraint}: &\quad \vec{\nabla} \cdot \vec{\pi} \approx 0 \\ \textit{gauge constraint}: &\quad A^0 \approx 0 \\ \textit{secondary gauge constraint}: &\quad \vec{\nabla} \cdot \vec{A} \approx 0 \ , \end{aligned} \tag{A.18}$$

forming the matrix $C_{\alpha\beta}$ between these four second class constraints, and inverting leads to the Dirac bracket conditions

$$\begin{aligned} \{\pi^\mu(t,\vec{x}), A^\nu(t,\vec{x}')\}^* &= \left(-g^{\mu\nu} - g^{\mu 0} g^{0\nu}\right) \delta^{(3)}(\vec{x} - \vec{x}') + \frac{\partial^2}{\partial x_\mu \partial x'_\nu} \frac{1}{4\pi |\vec{x} - \vec{x}'|} \ , \\ \{\pi^\mu, \pi^\nu\}^* &= \{A^\mu, A^\nu\}^* = 0 \ , \end{aligned} \tag{A.19}$$

total Hamiltonian

$$H = \int d^3\vec{x} \left[\frac{1}{2}\vec{\pi}^2 + \frac{1}{2}\vec{B}^2\right] \ , \tag{A.20}$$

and equations of motion

$$\begin{aligned} \dot{\vec{A}} &= \{\vec{A}, H\}^* = \vec{\pi} \ , \\ \dot{\vec{\pi}} &= \{\vec{\pi}, H\}^* = -\vec{\nabla} \times \vec{B} = \nabla^2 \vec{A} \ . \end{aligned} \tag{A.21}$$

These two equations of motion can be combined into one equation to reproduce the familiar result

$$\Box \vec{A} = 0 \ . \tag{A.22}$$

Since the second class constraints are now strong equations, one also has $\vec{\nabla} \cdot \vec{A} = 0$, which is the condition one normally writes for the Coulomb gauge. We find in Dirac's formalism that Coulomb gauge and temporal gauge are equivalent!

Light-cone QED in light-cone gauge ($A^+ = 0$) is considered as a second example. In this example only, $g^{\mu\nu}$, x^+ and x^- are defined to be

$$g^{\mu\nu} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & +1 & 0 \\ 0 & 0 & 0 & +1 \end{pmatrix}, \quad x^+ = \frac{1}{\sqrt{2}}(x^3 + x^0) , \quad x^- = \frac{1}{\sqrt{2}}(x^3 - x^0) , \tag{A.23}$$

which differs from the notation used in the rest of this paper given in Table 2. The canonical momenta and Hamiltonian are found from the Lagrangian to be

$$\begin{aligned} \pi^\mu &= \partial^\mu A^+ - \partial^+ A^\mu , \\ H_c &= \int dx^- d^2\vec{x}_\perp \left[\frac{1}{2}(\pi^-)^2 + \frac{1}{2}(F^{12})^2 + (\pi^- \partial_- + \pi^i \partial_i) A^- \right] . \end{aligned} \tag{A.24}$$

This gives two primary constraints and one resulting secondary constraint,

$$\begin{aligned} \textit{primary constraint}: \quad & \pi^+ \approx 0 \\ \textit{primary constraint}: \quad & \pi^i - \partial_i A^+ + \partial_- A^i \approx 0 \\ \textit{secondary constraint}: \quad & \partial_- \pi^- + \partial_i \pi^i \approx 0 . \end{aligned} \tag{A.25}$$

Recall that the velocity fields in light-cone formalism are $\partial_+ A^\mu$. The first and third of these constraint equations are first class and the second second class. We are thereby accorded two gauge conditions. Choosing $A^+ \approx 0$ leads to one secondary gauge constraint. The full set of constraints is

$$\begin{aligned} \textit{primary constraint}: \quad & \pi^+ \approx 0 \\ \textit{primary constraint}: \quad & \pi^i - \partial_i A^+ + \partial_- A^i \approx 0 \\ \textit{secondary constraint}: \quad & \partial_- \pi^- + \partial_i \pi^i \approx 0 \\ \textit{gauge constraint}: \quad & A^+ \approx 0 \\ \textit{secondary gauge constraint}: \quad & \pi^- + \partial_- A^- \approx 0 . \end{aligned} \tag{A.26}$$

All of these constraints are now second class. Again, the matrix $C_{\alpha\beta}$ is formed and inverted to convert the five constraints into strong equations and give the total

Hamiltonian and equations of motion,

$$H = \int dx^- d^2\vec{x}_\perp \left[-\frac{1}{2}(\partial^+ A^-)^2 + \frac{1}{2}(F^{12})^2 - \partial^+ A^i \partial_i A^- \right] , \qquad \Box A^i = 0 . \tag{A.27}$$

The Dirac bracket conditions are given in Steinhardt (Ref. 24) on page 446. If one had included a current in this last example, the five constraints would have been

$$\begin{aligned} \textit{primary constraint}: &\quad \pi^+ \approx 0 \\ \textit{primary constraint}: &\quad \pi^i - \partial_i A^+ + \partial_- A^i \approx 0 \\ \textit{secondary constraint}: &\quad \partial_- \pi^- + \partial_i \pi^i + j^+ \approx 0 \\ \textit{gauge constraint}: &\quad A^+ \approx 0 \\ \textit{secondary gauge constraint}: &\quad \pi^- + \partial_- A^- \approx 0 . \end{aligned} \tag{A.28}$$

Solving for $C^{-1}_{\alpha\beta}$ gives

$$H = \int dx^- d^2\vec{x}_\perp \left[-\frac{1}{2}(\partial^+ A^-)^2 + \frac{1}{2}(F^{12})^2 - \partial^+ A^i \partial_i A^- - j^+ A^- - j^i A^i \right] . \tag{A.29}$$

This turns out to be just the Hamiltonian that is derived in Section 4 (if one includes the free fermion Hamiltonian). Three of the now strong constraint equations can be re-written

$$\pi^i = -\partial^+ A^i , \qquad \pi^- = -\partial^+ A^- , \qquad (\partial^+)^2 A^- + \partial^+ \partial_i A^i = j^+ . \tag{A.30}$$

One recognizes the first of these as just the definition of π^i used in Section 4 (Eq. (4.5)) and the third equation as the constraint equation used to solve for A^- (Eq. (4.6)).

One important point should be noted regarding light-cone gauge. One frequently finds mentioned in the literature that $A^+ = 0$ gauge still has residual

gauge degrees of freedom (i.e.: an x^- independent gauge condition can still be imposed). Dirac's method for constrained Hamiltonians shows that light-cone gauge is actually two conditions (corresponding to two first class constraints),

$$A^+ = 0 \,, \qquad \pi^- + \partial_- A^- = 0 \,. \tag{A.31}$$

This second condition arises from requiring that the first condition remain valid for all light-cone time x^+ and uses up any residual gauge freedom. One also finds a similar phenomenon in axial gauge ($A^3 = 0$): the gauge condition is really two conditions, which uses up any residual gauge freedom allowed in axial gauge. Details are in Hanson, Regge and Teitelboim (Ref. 20).

APPENDIX B

The derivation of the quantities

$$
\begin{aligned}
[n \mid m] &= \left(\frac{\pi}{L}\right)^2 \frac{1}{2L} \int_{-L}^{L} dx^- \left\{ e^{\frac{i}{2}k_n^+ x^-} \frac{1}{(i\partial^+)^2} e^{-\frac{i}{2}k_m^+ x^-} \right\}_{\text{sym}} , \\
\{n \mid m\} &= \left(\frac{\pi}{L}\right) \frac{1}{2L} \int_{-L}^{L} dx^- \left\{ e^{\frac{i}{2}k_n^+ x^-} \frac{1}{i\partial^+} e^{-\frac{i}{2}k_m^+ x^-} \right\}_{\text{sym}} ,
\end{aligned}
\tag{B.1}
$$

is given in this appendix following a method suggested by Hamer (Ref. 19). The definition of $\{\ldots\}_{\text{sym}}$ was given in Section 4. The following two definite integrals occur frequently and their values are given here:

$$
\begin{aligned}
&\int_{-L}^{L} dx^- \; x^- \; e^{-\frac{i}{2}k^+ x^-} \\
&= \begin{cases} 0 & k^+ = 0 \\ \frac{4i}{(k^+)^2}\left[k^+ L \cos\left(\frac{k^+ L}{2}\right) + 2\sin\left(\frac{k^+ L}{2}\right)\right] & k^+ \neq 0 \end{cases} , \\
&\int_{-L}^{L} dx^- \; (x^-)^2 \; e^{-\frac{i}{2}k^+ x^-} \\
&= \begin{cases} \frac{2}{3}L^3 & k^+ = 0 \\ \frac{4}{(k^+)^3}\left[(k^+ L)^2 \sin\left(\frac{k^+ L}{2}\right) + 4k^+ L \cos\left(\frac{k^+ L}{2}\right) - 8\sin\left(\frac{k^+ L}{2}\right)\right] & k^+ \neq 0 \end{cases} .
\end{aligned}
\tag{B.2}
$$

We start with the most general form of $\frac{1}{i\partial^+} e^{-\frac{i}{2}k_m^+ x^-}$ and $\frac{1}{(i\partial^+)^2} e^{-\frac{i}{2}k_m^+ x^-}$,

$$
\begin{aligned}
\frac{1}{i\partial^+} e^{-\frac{i}{2}k_m^+ x^-} &= \begin{cases} -\frac{i}{2}x^- + A_0 & m = 0 \\ \frac{1}{k_m^+} e^{-\frac{i}{2}k_m^+ x^-} + A_m & m \neq 0 \end{cases} , \\
\frac{1}{(i\partial^+)^2} e^{-\frac{i}{2}k_m^+ x^-} &= \begin{cases} -\frac{1}{8}(x^-)^2 + B_0 x^- + C_0 & m = 0 \\ \frac{1}{(k_m^+)^2} e^{-\frac{i}{2}k_m^+ x^-} + B_m x^- + C_m & m \neq 0 \end{cases} .
\end{aligned}
\tag{B.3}
$$

This form is substituted into the definition of $\{n|m\}$ to give

$$\{n \mid m\} \;=\; \begin{cases} 0 & n,m=0 \\ \frac{\pi}{2L}A_m - \frac{\pi}{2k_m^+ L}\cos\frac{k_m^+ L}{2} + \frac{\pi}{(k_m^+ L)^2}\sin\frac{k_m^+ L}{2} & n=0, m\neq 0 \\ -\frac{\pi}{2L}A_{-n} - \frac{\pi}{2k_n^+ L}\cos\frac{k_n^+ L}{2} + \frac{\pi}{(k_n^+ L)^2}\sin\frac{k_n^+ L}{2} & n\neq 0, m=0 \\ \frac{1}{m}\,\delta_{n,m} & n,m\neq 0\ . \end{cases} \tag{B.4}$$

$\{n|m\}$ appears in the instantaneous fermion interaction. For example, the interaction shown in Figure 35 is proportional to $\{p+m|q+n\}$ (see Fig. 7 in Section 4). Since we require conservation of light-cone momentum P^+, $\{n\neq 0|0\}$ and $\{0|m\neq 0\}$ must equal zero. Otherwise, this interaction would not conserve P^+. This requirement fixes A_m to be

$$A_m \;=\; 2L\left[\frac{1}{2k_m^+ L}\cos\frac{k_m^+ L}{2} - \frac{1}{(k_m^+ L)^2}\sin\frac{k_m^+ L}{2}\right]\ ,\quad m\neq 0\ . \tag{B.5}$$

Making use of this equality, we similarly evaluate $[n|m]$ by substituting the general form, Eq. (B.3), into the definition of $[n|m]$ to arrive at

$$[n \mid m] \;=$$

$$\begin{cases} \left(\frac{\pi}{L}\right)^2 \frac{1}{2L}\left[-\frac{1}{3}L^3 + 4LC_0 + 2LA_0^2\right] & n,m\neq 0 \\ \left(\frac{\pi}{L}\right)^2 \frac{1}{2L}\left[2LC_m + 2LA_0A_m - \frac{L^2}{2k_m^+}\sin\frac{k_m^+ L}{2} + 4iLB_0A_m\right] & n=0, m\neq 0 \\ \left(\frac{\pi}{L}\right)^2 \frac{1}{2L}\left[2LC_{-n} + 2LA_0A_{-n} - \frac{L^2}{2k_n^+}\sin\frac{k_n^+ L}{2} + 4iLB_0A_{-n}\right] & n\neq 0, m=0 \\ \frac{1}{m^2}\delta_{n,m} + \left(\frac{\pi}{L}\right)^2 \frac{1}{2L}\left[2LA_mA_{-n} + 4iLB_mA_{-n} + 4iLB_{-n}A_m\right] & n,m\neq 0\ . \end{cases} \tag{B.6}$$

As in the case for $\{n|m\}$, $[n|m]$ must be proportional to $\delta_{n,m}$. A representative diagram proportional to $[k-m|-l+n]$ is shown in Figure 36. This gives the following conditions on B_m and C_m for $m\neq 0$:

$$\begin{aligned} B_m \;&=\; \frac{i}{4}A_m\ , \\ C_m \;&=\; -A_0A_m + \frac{L}{4k_m^+}\sin\frac{k_m^+ L}{2} - 2iB_0A_m\ . \end{aligned} \tag{B.7}$$

Using the values for A_m, B_m, and C_m given above and defining

$$\kappa \;=\; \left(\frac{\pi}{L}\right)^2 \frac{1}{2L}\left[-\frac{1}{3}L^3 + 4LC_0 + 2LA_0^2\right] \tag{B.8}$$

gives us the final answer,

$$\begin{aligned}
\{n \mid m\} \;&=\; \begin{cases} 0 & n \text{ or } m = 0 \\ \frac{1}{n}\,\delta_{n,m} & n, m \neq 0 \end{cases} , \\
[n \mid m] \;&=\; \begin{cases} \kappa & n \text{ and } m = 0 \\ \frac{1}{n^2}\,\delta_{n,m} & n, m \neq 0 \\ 0 & \text{otherwise} . \end{cases}
\end{aligned} \tag{B.9}$$

Though κ is an undetermined quantity, its value turns out to be irrelevant in DLCQ as long as it is finite because the ultraviolet cut-off removes all occurrences of $[0|0]$ (see Section 5).

APPENDIX C

In this appendix the answer for tree-level Møller scattering ($e^-e^- \to e^-e^-$) derived using Feynman's S matrix approach is shown to be identical to that derived from light-cone perturbation theory (LCPTh). The rules for LCPTh are given in Appendix B in Ref. 15 and Appendix A in Ref. 16 and can be derived from the light-cone Hamiltonian H_{LC} given in Eq. (4.24).

The diagrams that must be considered in LCPTh are given in Figure 37 with light-cone time x^+ flowing from left to right and momenta assigned as shown. Using P^+ and $\vec{P}_\perp$ momentum conservation, q and q' are

$$\begin{aligned} q^+ &= l_i^+ - l_f^+ = k_f^+ - k_i^+ \ , \\ \vec{q}_\perp &= \vec{l}_{\perp i} - \vec{l}_{\perp f} = \vec{k}_{\perp f} - \vec{k}_{\perp i} \ , \\ q^- &= \frac{q_\perp^2 + \lambda^2}{q^+} \ , \\ q^{\mu'} &= -q^\mu \ . \end{aligned} \tag{C.1}$$

Note that the photon's 4-momentum, q, is on mass shell. Remember that P^- is not necessarily conserved, so

$$q^- \neq l_i^- - l_f^- \neq k_f^- - k_i^- \ . \tag{C.2}$$

Using the LCPTh rules found in Ref. 15 or 16 and performing the sum over photon polarizations gives the following for the three LCPTh graphs,

$$\begin{aligned} T_{fi}^{(1)} &= e^2 \ \bar{u}(l_f)\gamma_\mu u(l_i) \ \bar{u}(k_f)\gamma_\nu u(k_i) \ \frac{\eta^\mu \eta^\nu}{(q^+)^2} \ , \\ T_{fi}^{(2)} &= e^2 \ \theta(q^+) \ \bar{u}(l_f)\gamma_\mu u(l_i) \ \bar{u}(k_f)\gamma_\nu u(k_i) \\ &\quad \times \left[-g^{\mu\nu} + \frac{\eta^\mu q^\nu + \eta^\nu q^\mu}{q^+} \right] \frac{1}{q^+(l_i^- - l_f^-) - q^+ q^- + i\epsilon} \ , \\ T_{fi}^{(3)} &= e^2 \ \theta(-q^+) \ \bar{u}(l_f)\gamma_\mu u(l_i) \ \bar{u}(k_f)\gamma_\nu u(k_i) \\ &\quad \times \left[-g^{\mu\nu} + \frac{\eta^\mu q^\nu + \eta^\nu q^\mu}{q^+} \right] \frac{1}{-q^+(k_i^- - k_f^-) - q^+ q^- + i\epsilon} \ . \end{aligned} \tag{C.3}$$

where $\eta^\mu = (0, 2, \vec{0}_\perp)$. Note that $T_{fi}^{(1)}$ diverges like $1/(q^+)^2$ for small q^+. The sum

of these three amplitudes is

$$T_{fi} = e^2 A_\mu B_\nu \left\{ \frac{\eta^\mu \eta^\nu}{(q^+)^2} + \left[-g^{\mu\nu} + \frac{\eta^\mu q^\nu + \eta^\nu q^\mu}{q^+} \right] \right.$$
$$\left. \times \left[\frac{\theta(q^+)}{q^+(l_i^- - l_f^-) - q^+ q^- + i\epsilon} + \frac{\theta(-q^+)}{-q^+(k_i^- - k_f^-) - q^+ q^- + i\epsilon} \right] \right\} , \qquad \text{(C.4)}$$
$$A_\mu = \bar{u}(l_f)\gamma_\mu u(l_i) , \qquad B_\nu = \bar{u}(k_f)\gamma_\nu u(k_i) .$$

Writing out the components $\mu, \nu = +, -, 1, 2$ explicitly, one finds after some algebra,

$$A_\mu B_\nu \; \theta(q^+) \left[\frac{\eta^\mu \eta^\nu}{(q^+)^2} + \frac{\eta^\mu q^\nu + \eta^\nu q^\mu}{q^+} \frac{1}{q^+(l_i^- - l_f^-) - q^+ q^- + i\epsilon} \right]$$
$$= A_\mu B_\nu \frac{\theta(q^+)}{q^+} \frac{1}{q^+(l_i^- - l_f^-) - q^+ q^- + i\epsilon} \left[\eta^\mu (l_i - l_f)^\nu + \eta^\nu (l_i - l_f)^\mu \right] . \qquad \text{(C.5)}$$

This expression can be summed with a similar expression for the $\theta(-q^+)$ term to give

$$T_{fi} = e^2 \; \bar{u}(l_f)\gamma_\mu u(l_i) \; \bar{u}(k_f)\gamma_\nu u(k_i)$$
$$\times \left\{ -g^{\mu\nu} \left[\frac{\theta(q^+)}{q^+(l_i^- - l_f^-) - q_\perp^2 - \lambda^2 + i\epsilon} + \frac{\theta(-q^+)}{q^+(k_f^- - k_i^-) - q_\perp^2 - \lambda^2 + i\epsilon} \right] \right.$$
$$+ \theta(q^+) \frac{\eta^\mu (l_i - l_f)^\nu + \eta^\nu (l_i - l_f)^\mu}{q^+} \frac{1}{q^+(l_i^- - l_f^-) - q_\perp^2 - \lambda^2 + i\epsilon}$$
$$\left. + \theta(-q^+) \frac{\eta^\mu (k_f - k_i)^\nu + \eta^\nu (k_f - k_i)^\mu}{q^+} \frac{1}{q^+(k_f^- - k_i^-) - q_\perp^2 - \lambda^2 + i\epsilon} \right\} . \qquad \text{(C.6)}$$

This result is valid for on- or off-shell electrons and does not assume P^- momentum conservation. Now note that this final expression for T_{fi} diverges only like $1/q^+$ for small q^+. The leading $1/(q^+)^2$ behavior from $T_{fi}^{(1)}$ is exactly cancelled by a similar singularity from $T_{fi}^{(2)}$ and $T_{fi}^{(3)}$.

The Feynman rules answer can be obtained by first enforcing four-momentum conservation (i.e.: $k_i^- + l_i^- = k_f^- + l_f^-$),

$$T_{fi} = e^2\, \bar{u}(l_f)\gamma_\mu u(l_i)\, \bar{u}(k_f)\gamma_\nu u(k_i) \times \frac{1}{q^+(l_i^- - l_f^-) - q_\perp^2 - \lambda^2 + i\epsilon} \left[-g^{\mu\nu} + \frac{\eta^\mu (l_i - l_f)^\nu + \eta^\nu (l_i - l_f)^\mu}{q^+}\right], \quad \text{(C.7)}$$

and then requiring the electrons to be on-shell (i.e.: $\bar{u}(l_f)(\not{l}_i - \not{l}_f)u(l_i) = \bar{u}(k_f)(\not{l}_i - \not{l}_f)u(k_i) = \ldots = 0$),

$$T_{fi} = -e^2\, \bar{u}(l_f)\gamma_\mu u(l_i)\, \bar{u}(k_f)\gamma_\nu u(k_i)\, \frac{g^{\mu\nu}}{q_{FR}^2 - \lambda^2 + i\epsilon}\,. \quad \text{(C.8)}$$

q_{FR}^μ is defined to be $l_i^\mu - l_f^\mu = k_f^\mu - k_i^\mu$. This last answer is recognized as the familiar answer for Møller scattering using Feynman rules.

APPENDIX D

The calculation of various self-mass diagrams is given in this appendix. The first to be considered is the familiar one-loop fermion self-mass diagram shown in Figure 38. The various momenta are

$$
\begin{aligned}
p &= \left(xP, \frac{x^2 p_\perp^2 + m_e^2}{xP}, x\vec{p}_\perp \right) , \\
k_1 &= \left(yP, \frac{(\vec{k}_\perp + y\vec{p}_\perp)^2 + \lambda^2}{yP}, \vec{k}_\perp + y\vec{p}_\perp \right) , \\
k_2 &= \left((x-y)P, \frac{(-\vec{k}_\perp + (x-y)\vec{p}_\perp)^2 + m_e^2}{(x-y)P}, -\vec{k}_\perp + (x-y)\vec{p}_\perp \right) .
\end{aligned}
\tag{D.1}
$$

The light-cone perturbation theory (LCPTh) amplitude for this process is

$$
\begin{aligned}
T_{fi} &= \frac{g^2}{16\pi^3} \frac{1}{P} \int_0^x dy \int d^2\vec{k}_\perp \frac{1}{y(x-y)} \frac{N}{D + i\epsilon} , \\
N &= \bar{u}(p) \not{\epsilon} u(k_2)\ \bar{u}(k_2) \not{\epsilon}^* u(p) , \\
D &= \frac{x^2 p_\perp^2 + m_e^2}{xP} - \frac{(\vec{k}_\perp + y\vec{p}_\perp)^2 + \lambda^2}{yP} - \frac{(-\vec{k}_\perp + (x-y)\vec{p}_\perp)^2 + m_e^2}{(x-y)P} .
\end{aligned}
\tag{D.2}
$$

The rules for LCPTh QED are derived in Appendix B in Ref. 15 and Appendix A in Ref. 16. The photon spin sum can be done by making use of the relation

$$
\sum_\lambda \epsilon_\lambda^\mu \epsilon_\lambda^{\nu *} = -g^{\mu\nu} + \frac{\eta^\mu q^\nu + \eta^\nu q^\mu}{q^+} , \tag{D.3}
$$

which holds for the spinors given in Eq. (4.13) with $\eta^\mu = (0, 2, \vec{0}_\perp)$. Doing the numerator algebra and simplifying the denominator produces the desired answer

$$
T_{fi} = -\delta_{ss'} \frac{g^2}{8\pi^3} x \int_0^x dy \int d^2\vec{k}_\perp \frac{\frac{1}{x(x-y)}[x^2 k_\perp^2 + y^2 m_e^2] + \frac{2}{y^2}[x^2 k_\perp^2 + x(x-y)\lambda^2]}{x^2 k_\perp^2 + y^2 m_e^2 + \lambda^2 x(x-y) - i\epsilon} . \tag{D.4}
$$

$\delta_{ss'}$ is a delta function between the incoming and outgoing fermion spins. Note that as expected from Lorentz invariance, this answer is independent of $\vec{p}_\perp$. If one

changes variables to $z = y/x$, one also finds that the answer is independent of P and x. Since T_{fi} evidently does not depend on any of the quantum numbers of the incoming fermion, T_{fi} can be considered to be a pure mass renormalization.

The quantities actually discretized are x, y, $\vec{p}'_\perp = x\vec{p}_\perp$ and $\vec{k}'_\perp = \vec{k}_\perp + y\vec{p}_\perp$ or $-\vec{k}_\perp + (x-y)\vec{p}_\perp$. The choice between these last two is irrelevant. Rewriting T_{fi} in terms of these quantities gives

$$T_{fi} = -\delta_{ss'}\frac{g^2}{8\pi^3}\, x \int_0^x dy \int d^2\vec{k}_\perp \frac{\frac{1}{x(x-y)}\left[x^2(\vec{k}'_\perp - \frac{y}{x}\vec{p}'_\perp)^2 + y^2 m_e^2\right] + \frac{2}{y^2}\left[x^2(\vec{k}'_\perp - \frac{y}{x}\vec{p}'_\perp)^2 + x(x-y)\lambda^2\right]}{x^2(\vec{k}'_\perp - \frac{y}{x}\vec{p}'_\perp)^2 + y^2 m_e^2 + \lambda^2 x(x-y) - i\epsilon} \tag{D.5}$$

This answer is discretized by replacing

$$\begin{gathered} x = \frac{n}{K}\,, \qquad y = \frac{q}{K}\,, \qquad \vec{p}'_\perp = \frac{\pi\,\vec{n}_\perp}{L_\perp}\,, \qquad \vec{k}'_\perp = \frac{\pi\,\vec{q}_\perp}{L_\perp}\,, \\ \int dy = \frac{2}{K}\sum_q\,, \qquad \int d^2\vec{k}_\perp = \left(\frac{\pi}{L_\perp}\right)^2 \sum_{\vec{q}_\perp} \end{gathered} \tag{D.6}$$

where $\pi n/L$ and $\pi\vec{n}_\perp/L_\perp$ are the P^+ and $\vec{P}_\perp$ of the incoming fermion, respectively, and $q = 2, 4, 6, \ldots$. A factor of $1/x$ is also necessary because in the continuum, factors of $1/\sqrt{x}$ from external wavefunctions are conventionally associated with the wavefunctions themselves; whereas in the discretized case, the factors of $1/\sqrt{x}$ are absorbed into P^-. These steps give the result

$$T_{fi} = -\delta_{ss'}\; K\,\frac{2\alpha}{L_\perp^2}\sum_{q,\vec{q}_\perp} \frac{\frac{1}{2n(n-q)}\left[n^2\left(\vec{q}_\perp - \frac{q}{n}\vec{n}_\perp\right)^2 + q^2\beta_f\right] + \frac{n^2}{q^2}\left(\vec{q}_\perp - \frac{q}{n}\vec{n}_\perp\right)^2}{n^2\left(\vec{q}_\perp - \frac{q}{n}\vec{n}_\perp\right)^2 + q^2\beta_f + n(n-q)\;\beta_\gamma} \tag{D.7}$$

where $\beta_f = (mL_\perp/\pi)^2$ and $\beta_\gamma = (\lambda L_\perp/\pi)^2$. The photon mass, λ, has been set equal to zero in the numerator in this last expression.

Ultraviolet and infrared regulators are implemented by requiring that the in-

termediate state in Fig. 38 satisfy

$$\sum_i \frac{k_{\perp i}^2 + m_i^2}{x_i} \leq \Lambda^2 \,, \qquad \frac{(\vec{k}_\perp + y\vec{p}_\perp)^2 + \lambda^2}{y} \geq \epsilon \,, \tag{D.8}$$

which in terms of the discrete variables given above reads

$$\begin{aligned} \frac{q_\perp^2 + \beta_\gamma}{q} + \frac{(\vec{n}_\perp - \vec{q}_\perp)^2 + \beta_f}{n-q} &\leq \frac{1}{K}\left(\frac{\Lambda L_\perp}{\pi}\right)^2 - \sum_{\text{spec}} \frac{m_\perp^2 + \beta}{m} \quad , \\ \frac{q_\perp^2 + \beta_\gamma}{q} &\geq \frac{1}{K}\left(\frac{L_\perp}{\pi}\right)^2 \epsilon \quad . \end{aligned} \tag{D.9}$$

Here, β_i is equal to $(m_i L_\perp/\pi)^2$. The sum is over any spectator particles that might occur during the process. The correct mass counterterm that should be inserted in H_{LC} to ensure that the fermion's bare mass is equal to its physical mass is the negative of Eq. (D.7) where the sum is over $q^i = 0, \pm 1, \pm 2, \ldots$ and $q = 2, 4, 6, \ldots, n-1$ that satisfy Eq. (D.9).

The next self-mass diagram to consider is shown in Figure 39 . The momenta are assigned to be

$$\begin{aligned} p &= \left(xP, \frac{x^2 p_\perp^2 + m_e^2}{xP}, x\vec{p}_\perp\right) \,, \\ k_1 &= \left(yP, \frac{(\vec{k}_\perp + y\vec{p}_\perp)^2 + \lambda^2}{yP}, \vec{k}_\perp + y\vec{p}_\perp\right) \,, \\ k_2 &= \left((x-y)P, \frac{(-\vec{k}_\perp + (x-y)\vec{p}_\perp)^2 + m_e^2}{(x-y)P}, -\vec{k}_\perp + (x-y)\vec{p}_\perp\right) \,, \\ l_1 &= \left(zP, \frac{(\vec{l}_\perp + z\vec{p}_\perp)^2 + \lambda^2}{zP}, \vec{l}_\perp + z\vec{p}_\perp\right) \,, \\ l_2 &= \left((x-z)P, \frac{(-\vec{l}_\perp + (x-z)\vec{p}_\perp)^2 + m_e^2}{(x-z)P}, -\vec{l}_\perp + (x-z)\vec{p}_\perp\right) \end{aligned} \tag{D.10}$$

and the answer in LCPTh is

$$T_{fi} = \frac{1}{2xP}\left(\frac{g^2}{16\pi^3}\right)^2 \int_0^x dy\, dz \int d^2\vec{k}_\perp d^2 l_\perp \, \frac{1}{y(x-y)z(x-z)}\,\frac{N}{D}\,,$$

$$N = \bar{u}(p)\not{q}(l_1)u(l_2)\;\bar{u}(l_2)\not{q}(l_1)^*\gamma^+\not{q}(k_1)u(k_2)\;\bar{u}(k_2)\not{q}(k_1)^*u(p)\,,$$

$$D = \left[\frac{x^2p_\perp^2+m_e^2}{x} - \frac{(\vec{k}_\perp+y\vec{p}_\perp)^2+\lambda^2}{y} - \frac{(-\vec{k}_\perp+(x-y)\vec{p}_\perp)^2+m_e^2}{x-y} + i\epsilon\right]$$

$$\times\left[\frac{x^2p_\perp^2+m_e^2}{x} - \frac{(\vec{l}_\perp+z\vec{p}_\perp)^2+\lambda^2}{y} - \frac{(-\vec{l}_\perp+(x-z)\vec{p}_\perp)^2+m_e^2}{x-z} + i\epsilon\right]\,. \tag{D.11}$$

The numerator algebra is done by using the photon spin sum relation Eq. (D.3), applying symmetric integration to eliminate various terms proportional to k^i and l^i (upon simplification, the denominator turns out to only involve $k_\perp^2$ and $l_\perp^2$), and making use of the spinor properties shown in Appendix H. The answer for the numerator,

$$N = 8Pm_e^2\,\frac{zy}{x}\,\delta_{ss'}\,, \tag{D.12}$$

turns out to only have a contribution from the spin-flip interaction of H_{LC}. s and s' are the spins of the incoming and outgoing fermion, respectively. The complete answer is then

$$T_{fi} = \delta_{ss'}\left[\frac{m_e g^2}{8\pi^3}\int_0^x dy \int d^2\vec{k}_\perp \,\frac{y}{x^2k_\perp^2+y^2m_e^2+\lambda^2x(x-y)-i\epsilon}\right]^2\,. \tag{D.13}$$

Again, changing variables to $z = y/x$ demonstrates that this result is independent of x, P, and $\vec{p}_\perp$ and is therefore a pure mass renormalization.

Next, consider the case of N one-loop fermion self-mass pieces all connected by instantaneous fermions shown in Figure 40. As above, momenta are assigned and the LCPTh answer is written down for T_{fi}. The numerator and denominator

are both factorizable, giving an answer of

$$T_{fi}^{(N)} = \left[\frac{g^2}{8\pi^3 x} \int_0^x dy \int d^2\vec{k}_\perp \, \frac{1}{y} \, \frac{1}{\frac{m_e^2}{x} - \frac{k_\perp^2+\lambda^2}{y} - \frac{k_\perp^2+m_e^2}{x-y} + i\epsilon} \right]^{N-2} T_{fi}^{(2)} \qquad \text{(D.14)}$$

where $T_{fi}^{(2)}$ is the answer for the diagram in Fig. 39. Using

$$\sum_{N=2}^{\infty} = \frac{1}{1-x} \qquad \text{(D.15)}$$

and substituting in Eq. (D.13) for $T_{fi}^{(2)}$ yields

$$T_{fi} = \delta_{ss'} \, \frac{m_e^2 \left[\frac{g^2}{8\pi^3} \int_0^x dy \int d^2\vec{k}_\perp \, \frac{y}{x^2k_\perp^2+y^2m_e^2+\lambda^2x(x-y)-i\epsilon} \right]^2}{1 + \frac{g^2}{8\pi^3} \int_0^x dy \int d^2\vec{k}_\perp \, \frac{x-y}{x^2k_\perp^2+y^2m_e^2+\lambda^2x(x-y)-i\epsilon}} \qquad \text{(D.16)}$$

as the amplitude for the process shown in Figure 41. Similarly to above, this result is discretized by re-writing in terms of x, y, $\vec{p}'_\perp = x\vec{p}_\perp$ and $\vec{k}'_\perp = \vec{k}_\perp + y\vec{p}_\perp$ and making the substitutions in Eq. (D.6) to give

$$T_{fi} = \delta_{ss'} \, K \, \frac{\beta_f \pi^2}{n L_\perp^2} \, \frac{\left[\frac{\alpha}{\pi^2} \sum_{q,\vec{q}_\perp} \frac{q}{n^2\left(\vec{q}_\perp - \frac{q}{n}\vec{n}_\perp\right)^2 + q^2\beta_f + n(n-q)\beta_\gamma} \right]^2}{1 + \frac{\alpha}{\pi^2} \sum_{q,\vec{q}_\perp} \frac{n-q}{n^2\left(\vec{q}_\perp - \frac{q}{n}\vec{n}_\perp\right)^2 + q^2\beta_f + n(n-q)\beta_\gamma}} \, . \qquad \text{(D.17)}$$

This answer is subject to the same regulation conditions as above, Eq. (D.9). The mass counterterm necessary in H_{LC} is the negative of Eq. (D.17) subject to the conditions, Eq. (D.9). A combination of the mass counterterms, Eq. (D.7) and Eq. (D.17), provides the full mass renormalization needed in the truncated Fock space $(e^-, e^-\gamma)$ or $(e^+e^-, e^+e^-\gamma)$.

An added result that is not needed in this truncated Fock space but is included for information is the one-loop photon self-mass shown in Figure 42. The continuum answer for this graph is

$$T_{fi} = -\delta_{\lambda\lambda'}\frac{g^2}{8\pi^3}\, x \int_0^x dy \int d^2\vec{k}_\perp \frac{\left(\frac{y}{x-y}+\frac{x-y}{y}\right)(k_\perp^2+m_e^2)+2m_e^2}{x^2(k_\perp^2+m_e^2)-\lambda^2 y(x-y)-i\epsilon} \quad , \qquad \text{(D.18)}$$

which is a pure photon mass renormalization. The familiar answer derived from Feynman rules of zero is obtained because a term with massive, negative metric Pauli-Villars particles is subtracted that exactly cancels the original integral (more on this in Appendix E). Eq. (D.18) gives a non-zero photon mass renormalization since subtractive regulation has not been invoked.

APPENDIX E

The equivalence of answers derived using Feynman's S-matrix analysis and using infinite momentum frame time-ordered perturbation theory (TOPTh_∞) is demonstrated in this appendix for the one-loop fermion self-energy diagram in Feynman gauge and the one-loop vacuum polarization graph in light-cone gauge. Since it is believed that light-cone perturbation theory (LCPTh) and TOPTh_∞ are mathematically equivalent, this demonstration makes the equivalence of LCPTh and Feynman rules results for one-loop radiative corrections plausible. The analysis for the fermion self-energy is done in Feynman gauge for convenience, though the analysis should be similar for light-cone gauge.

First, the Feynman rules answer for the fermion self-energy graph shown in Figure 43 is described briefly. We start with the familiar result

$$T_{fi} = -\frac{ig^2}{(2\pi)^4}\int d^4k\,\frac{\bar{u}(p)\gamma^\mu(\not{p}-\not{k}+m_e)\gamma_\mu u(p)}{[(p-k)^2-m_e^2+i\epsilon]\;(k^2-\lambda^2+i\epsilon)}\quad . \tag{E.1}$$

A factor of $-i$ has been included to facilitate comparison with TOPTh_∞. Doing the numerator algebra, combining denominators, changing variables to $q^\mu = k^\mu - xp^\mu$, and eliminating terms proportional to q^μ by symmetric integration gives

$$T_{fi} = -\delta_{ss'}\,\frac{ig^2}{(2\pi)^4}\int d^4q\int_0^1 dx\,\frac{4m_e^2(1+x)}{\left[q^2-a^2+i\epsilon\right]^2}\quad , \tag{E.2}$$

$$a^2 = m_e^2x^2+\lambda^2(1-x)\;.$$

The delta function is between the spin of the incoming and outgoing fermion. Doing the q^0 integral by contour integration and then the q^3 integral by standard methods results in

$$T_{fi} = \delta_{ss'}\,\frac{g^2}{8\pi^3}\int_0^1 dx\int d^2\vec{q}_\perp\,\frac{2m_e^2\,(1+x)}{q_\perp^2+x^2m_e^2+\lambda^2(1-x)-i\epsilon}\quad . \tag{E.3}$$

This answer diverges like $\log q_\perp^2$ for large $\vec{q}_\perp$; it is therefore necessary to introduce a regulator such as subtracting a Pauli-Villars contribution.

Now consider the same process in TOPTh$_\infty$. The TOPTh rules for QED in Feynman gauge are given in Ref. 11. Two graphs need to be considered, the usual time-ordering and the Z-graph. These are pictured in Figure 44. Momenta are assigned to the various legs of the usual time-ordering contribution,

$$p = \left(E, \vec{0}_\perp, P\right) , \qquad k_1 = \left(E_1, \vec{k}_\perp, xP\right) , \qquad k_2 = \left(E_2, -\vec{k}_\perp, (1-x)P\right) ,$$

$$E = \sqrt{P^2 + m_e^2} , \qquad E_1 = \sqrt{x^2P^2 + \lambda_\perp^2} , \qquad E_2 = \sqrt{(1-x)^2P^2 + m_\perp^2} ,$$

$$\lambda_\perp^2 = k_\perp^2 + \lambda^2 , \qquad m_\perp^2 = k_\perp^2 + m_e^2 . \tag{E.4}$$

The TOPTh answer for this graph is

$$\begin{aligned} T_{fi} \;=\; & \frac{g^2}{4(2\pi)^3}\, P \int_{-\infty}^{\infty} dx \int d^2\vec{k}_\perp \, \frac{1}{E_1 E_2} \, \frac{N}{D + i\epsilon} \;-\; (\lambda \to \Lambda) , \\ & N = \bar{u}(p) \not{\epsilon} u(k_2)\; \bar{u}(k_2) \not{\epsilon}^* u(p) , \\ & D = E - E_1 - E_2 . \end{aligned} \tag{E.5}$$

A Pauli-Villars contribution has been subtracted for ultraviolet regulation. The TOPTh$_\infty$ answer is gotten by letting P approach infinity, and the numerator is evaluated with the help of the relation

$$\sum_\lambda \epsilon_\lambda^\mu \, \epsilon_\lambda^{\nu *} = -g^{\mu\nu} , \tag{E.6}$$

which is valid in Feynman gauge. This gives the result

$$T_{fi} = \lim_{P\to\infty} \delta_{ss'} \frac{g^2}{8\pi^3} \int_{-\infty}^{\infty} dx \int d^2\vec{k}_\perp \; [I(\lambda, P) - I(\Lambda, P)] ,$$

$$I(\lambda, P) = \frac{1}{\sqrt{x^2 + \left(\frac{\lambda_\perp}{P}\right)^2}\sqrt{(1-x)^2 + \left(\frac{m_\perp}{P}\right)^2}} \; \frac{\sqrt{1+\left(\frac{m_e}{P}\right)^2}\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2} - (1-x) - 2\left(\frac{m_e}{P}\right)^2}{\sqrt{1+\left(\frac{m_e}{P}\right)^2} - \sqrt{x^2+\left(\frac{\lambda_\perp}{P}\right)^2} - \sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2} + i\epsilon} \tag{E.7}$$

for the usual time-ordering in TOPTh$_\infty$. Note that all the square roots are assumed to be *positive.*

The usual procedure is then to take the limit $P \to \infty$ inside the integral to simplify $I(\lambda, P)$. This is valid as long as one is not near the points $x = 0, 1$, which are singular for $P = \infty$. It is necessary to do a more detailed analysis near these two points. The integral is split into three regions: $x < 0, 0 < x < 1, x > 1$.

1. In the first region, $E \to P\left[1 + \frac{1}{2}(\frac{m}{P})^2\right]$, $E_1 \to -xP\left[1 + \frac{1}{2}(\frac{\lambda_\perp}{xP})^2\right]$, and $E_2 \to (1-x)P\left[1 + \frac{1}{2}(\frac{m_\perp}{(1-x)P})^2\right]$ as $P \to \infty$. $I(\lambda, P)$ approaches

$$-\frac{1}{x(1-x)} \frac{\frac{1}{2}(1-x)\frac{m_e^2}{P^2} + \frac{1}{2}\frac{m_\perp^2}{(1-x)P^2} - 2\frac{m_e^2}{P^2}}{2x} \xrightarrow[P\to\infty]{} 0 \,, \tag{E.8}$$

which is non-singular. Therefore, taking the limit before doing the x integral is allowed, giving the result

$$T_{fi}^{(1)} = 0 \,. \tag{E.9}$$

2. In this region, $E \to P\left[1 + \frac{1}{2}(\frac{m}{P})^2\right]$, $E_1 \to xP\left[1 + \frac{1}{2}(\frac{\lambda_\perp}{xP})^2\right]$, $E_2 \to (1-x)P\left[1 + \frac{1}{2}(\frac{m_\perp}{(1-x)P})^2\right]$ and

$$\begin{aligned} I(\lambda, P) &\to \frac{1}{x(1-x)} \frac{(1-x)m_e^2 + \frac{k_\perp^2 + m_e^2}{1-x} - 4m_e^2}{m_e^2 - \frac{k_\perp^2+\lambda^2}{x} - \frac{k_\perp^2+m_e^2}{1-x} + i\epsilon} \\ &= \frac{1}{1-x} \frac{(1-x)^2 m_e^2 + k_\perp^2 + m_e^2 - 4m_e^2(1-x)}{x(1-x)m_e^2 - (1-x)(k_\perp^2+\lambda^2) - x(k_\perp^2+m_e^2) + i\epsilon} \end{aligned} \tag{E.10}$$

as $P \to \infty$. $I(\lambda, P)$ has a singularity near $x = 1$. The integral for region 2 is split again into two parts

$$T_{fi}^{(2)} = \lim_{\epsilon\to 0} \lim_{P\to\infty} \delta_{ss'} \frac{g^2}{8\pi^3} \left[\int_0^{1-\epsilon} dx + \int_{1-\epsilon}^{1} dx\right] \int d^2\vec{k}_\perp \, [I(\lambda, P) - I(\Lambda, P)] \,. \tag{E.11}$$

(a) In the region $0 < x < 1 - \epsilon$, we are away from the singularity so the

limit $P \to \infty$ can be taken inside the integral to produce the answer

$$T_{fi}^{(2a)} = \lim_{\epsilon\to 0} \delta_{ss'} \frac{g^2}{8\pi^3} \int_0^{1-\epsilon} dx \int d^2\vec{k}_\perp \frac{1}{x(1-x)} \frac{(1-x)m_e^2 + \frac{k_\perp^2+m_e^2}{1-x} - 4m_e^2}{m_e^2 - \frac{k_\perp^2+\lambda^2}{x} - \frac{k_\perp^2+m_e^2}{1-x} + i\epsilon} - (\lambda \to \Lambda) . \tag{E.12}$$

(b) The non-singular part of $I(\lambda, P)$ is expanded in powers of $(1-x)$ to give the form

$$I(\lambda, P) = \frac{1}{\sqrt{(1-x)^2 + \left(\frac{m_\perp}{P}\right)^2}} \sum_{n=0}^{\infty} A_n(\lambda, P)\,(1-x)^n \tag{E.13}$$

for $I(\lambda, P)$. The contribution to T_{fi} is then

$$T_{fi}^{(2b)} = \lim_{\epsilon\to 0} \lim_{P\to\infty} \delta_{ss'} \frac{g^2}{8\pi^3} \int_{1-\epsilon}^{1} dx \int d^2\vec{k}_\perp \frac{1}{\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2}} \sum_{n=0}^{\infty} [A_n(\lambda, P) - A_n(\Lambda, P)]\,(1-x)^n . \tag{E.14}$$

Since λ and Λ appear in I only as λ/P and Λ/P, it must be that $A_n(\lambda, P) - A_n(\Lambda, P)$ approaches zero at least like $1/P$ as $P \to \infty$. One can expand A_n in powers of $1/P$ to see this. As $P \to \infty$, the *most* divergent x integral is

$$\int_{1-\epsilon}^{\epsilon} dx \frac{(1-x)^0}{\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2}} = \log\left(\frac{\epsilon + \sqrt{\epsilon^2+\left(\frac{m_\perp}{P}\right)^2}}{\frac{|m_\perp|}{P}}\right) \xrightarrow[P\to\infty]{} \log\frac{2\epsilon P}{|m_\perp|} . \tag{E.15}$$

The final answer as $P \to \infty$ is then

$$T_{fi}^{(2b)} \to \frac{1}{P}\log P \to 0 . \tag{E.16}$$

3. Finally, in the third region, $x > 1$,

$$I(\lambda, P) \xrightarrow[P\to\infty]{} \frac{1}{x(1-x)} , \tag{E.17}$$

which is singular near $x = 1$. As above, the integral is split into two pieces,

one for $1 < x < 1+\epsilon$ and one for $1+\epsilon < x < \infty$. In the first region, the non-singular part of $I(\lambda, P)$ is expanded in powers of $(x-1)$, similarly to Eq. (E.13). Again, we find that $A_n(\lambda, P) - A_n(\Lambda, P) \to 1/P$ as $P \to \infty$ and that the x integrals diverge at most like $\log P$. Thus, this region gives a zero contribution to T_{fi}. The limit $P \to \infty$ can be taken inside the x integral for $1+\epsilon < x < \infty$ since we are away from the singularity to give

$$T_{fi}^{(3)} = \lim_{\epsilon \to 0} \delta_{ss'} \frac{g^2}{8\pi^3} \int_{1+\epsilon}^{\infty} dx \int d^2\vec{k}_\perp \left[\frac{1}{x(1-x)} - \frac{1}{x(1-x)} \right] = 0 \,. \quad \text{(E.18)}$$

The contributions from the three x regions are now summed to give the final answer for the usual time-ordering, one-loop fermion self-energy diagram,

$$\begin{aligned} T_{fi} &= \delta_{ss'} \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \, \frac{1}{x(1-x)} \, \frac{(1-x)m_e^2 + \frac{k_\perp^2 + m_e^2}{1-x} - 4m_e^2}{m_e^2 - \frac{k_\perp^2+\lambda^2}{x} - \frac{k_\perp^2+m_e^2}{1-x} + i\epsilon} - (\lambda \to \Lambda) \\ &= \delta_{ss'} \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \, \frac{1}{1-x} \, \frac{(2-2x-2x^2)m_e^2 - k_\perp^2}{k_\perp^2 + x^2 m_e^2 + (1-x)\lambda^2 - i\epsilon} - (\lambda \to \Lambda) \,. \end{aligned} \quad \text{(E.19)}$$

Note that this result diverges like Λ^2 for large Λ. A term

$$1 = \frac{k_\perp^2 + x^2 m_e^2 + (1-x)\lambda^2}{k_\perp^2 + x^2 m_e^2 + (1-x)\lambda^2} \quad \text{(E.20)}$$

can be added to the first term in the integrand and an analogous term with λ replaced by Λ subtracted from the second term to give

$$T_{fi} = \delta_{ss'} \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \, \frac{2m_e^2 + \lambda^2}{k_\perp^2 + x^2 m_e^2 + (1-x)\lambda^2 - i\epsilon} - (\lambda \to \Lambda) \,. \quad \text{(E.21)}$$

Now turn to the Z-graph contribution. A procedure similar to the above for the usual time-ordering is applied. The momenta are assigned to be

$$p = \left(E, \vec{0}_\perp, P\right) , \quad k_1 = \left(E_1, \vec{k}_\perp, -xP\right) , \quad k_2 = \left(E_2, -\vec{k}_\perp, -(1-x)P\right) ,$$

$$E = \sqrt{P^2 + m_e^2} , \quad E_1 = \sqrt{x^2P^2 + \lambda_\perp^2} , \quad E_2 = \sqrt{(1-x)^2P^2 + m_\perp^2} ,$$

$$\lambda_\perp^2 = k_\perp^2 + \lambda^2 , \quad m_\perp^2 = k_\perp^2 + m_e^2 . \tag{E.22}$$

The TOPTh$_\infty$ result for the Z-graph including Pauli-Villars regulation is

$$\begin{aligned} T_{fi} &= \lim_{P\to\infty} \frac{g^2}{4(2\pi)^3} P \int_{-\infty}^{\infty} dx \int d^2\vec{k}_\perp \frac{1}{E_1E_2} \frac{N}{D+i\epsilon} - (\lambda \to \Lambda) , \\ N &= -\bar{u}(p)\not{\epsilon}^* v(k_2)\ \bar{v}(k_2)\not{\epsilon} u(p) , \\ D &= -E - E_1 - E_2 . \end{aligned} \tag{E.23}$$

Doing the numerator algebra gives

$$T_{fi} = \lim_{P\to\infty} \delta_{ss'} \frac{g^2}{8\pi^3} \int_{-\infty}^{\infty} dx \int d^2\vec{k}_\perp \ [I(\lambda,P) - I(\Lambda,P)] ,$$

$$I(\lambda,P) = \frac{1}{\sqrt{x^2+\left(\frac{\lambda_\perp}{P}\right)^2}\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2}} \frac{\sqrt{1+\left(\frac{m_e}{P}\right)^2}\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2}+(1-x)+2\left(\frac{m_e}{P}\right)^2}{\sqrt{1+\left(\frac{m_e}{P}\right)^2}+\sqrt{x^2+\left(\frac{\lambda_\perp}{P}\right)^2}+\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2}-i\epsilon} . \tag{E.24}$$

Again, we find potential singularities in $I(\lambda, P)$ near $x = 0, 1$. The integral is again split into three regions: $x > 1, 0 < x < 1, x < 0$.

1. For $x > 1$, $E \to P\left[1 + \frac{1}{2}(\frac{m}{P})^2\right]$, $E_1 \to xP\left[1 + \frac{1}{2}(\frac{\lambda_\perp}{xP})^2\right]$, and $E_2 \to (x-1)P\left[1 + \frac{1}{2}(\frac{m_\perp}{(1-x)P})^2\right]$ as $P \to \infty$ and

$$I(\lambda,P) \xrightarrow[P\to\infty]{} \frac{1}{x(x-1)} \frac{\frac{1}{2}(x-1)\frac{m_e^2}{P^2} + \frac{1}{2}\frac{m_\perp^2}{(x-1)P^2} - 2\frac{m_e^2}{P^2}}{2x} \to 0 , \tag{E.25}$$

which is non-singular. The limit $P \to \infty$ can be taken inside to give

$$T_{fi}^{(1)} = 0 . \tag{E.26}$$

2. In the second region,

$$I(\lambda, P) \xrightarrow[P\to\infty]{} \frac{1}{x} , \tag{E.27}$$

which is singular near $x = 0$. The integral is split into two pieces,

$$T_{fi}^{(2)} = \lim_{\epsilon\to 0}\lim_{P\to\infty} \delta_{ss'}\frac{g^2}{8\pi^3}\left[\int_0^\epsilon dx + \int_\epsilon^1 dx\right]\int d^2\vec{k}_\perp \ [I(\lambda, P) - I(\Lambda, P)] \ . \tag{E.28}$$

(a) The non-singular part of $I(\lambda, P)$ is expanded in powers of x for the region $0 < x < \epsilon$ to give

$$I(\lambda, P) = \frac{1}{\sqrt{x^2 + \left(\frac{\lambda_\perp}{P}\right)^2}} \sum_{n=0}^{\infty} A_n(\lambda, P)\ x^n \ . \tag{E.29}$$

Focus specifically on the contribution of the term A_0 to T_{fi},

$$\begin{aligned} T_{fi}^{(2a_0)} &= \lim_{P\to\infty} \delta_{ss'}\frac{g^2}{8\pi^3}\int d^2\vec{k}_\perp \left[A_0(\lambda, P)\int_0^\epsilon \frac{dx}{\sqrt{x^2+\left(\frac{\lambda_\perp}{P}\right)^2}} - (\lambda \to \Lambda)\right] \\ &= \lim_{P\to\infty} \delta_{ss'}\frac{g^2}{8\pi^3}\int d^2\vec{k}_\perp \left[A_0(\lambda, P)\log\left(\frac{\epsilon+\sqrt{\epsilon^2+\left(\frac{\lambda_\perp}{P}\right)^2}}{\frac{|\lambda_\perp|}{P}}\right) - (\lambda \to \Lambda)\right] . \end{aligned} \tag{E.30}$$

As $P \to \infty$, $A_0(\lambda, P)$ and $A_0(\Lambda, P)$ both approach one and the log approaches $\log\frac{2\epsilon P}{|\lambda_\perp|}$. Using these relations, we find

$$T_{fi}^{(2a_0)} = \delta_{ss'}\frac{g^2}{8\pi^3}\int d^2\vec{k}_\perp \log\frac{|\Lambda_\perp|}{|\lambda_\perp|} = \delta_{ss'}\frac{g^2}{16\pi^3}\int d^2\vec{k}_\perp \log\frac{k_\perp^2+\Lambda^2}{k_\perp^2+\lambda^2} \ . \tag{E.31}$$

Analysis of the other terms $A_n, n = 1, 2, 3, \ldots$ reveals that their contribution to T_{fi} all approach zero as $P \to \infty$. So, the complete answer for

the region $0 < x < \epsilon$ is

$$T_{fi}^{(2a)} = \delta_{ss'} \frac{g^2}{16\pi^3} \int d^2\vec{k}_\perp \log \frac{k_\perp^2 + \Lambda^2}{k_\perp^2 + \lambda^2} \quad . \tag{E.32}$$

(b) For $\epsilon < x < 1$ the integrand is non-singular, so the limit can be taken inside the integral to give

$$T_{fi}^{(2b)} = \delta_{ss'} \frac{g^2}{8\pi^3} \int_\epsilon^1 dx \int d^2\vec{k}_\perp \left[\frac{1}{x} - \frac{1}{x}\right] = 0 \; . \tag{E.33}$$

3. For $x < 0$, the results are similar to $0 < x < 1$. There is a singularity in $I(\lambda, P)$ near $x = 0$. Expanding I in powers of $-x$ for $-\epsilon < x < 0$ reveals a contribution identical to Eq. (E.32) from the term A_0. All other contributions vanish as $P \to \infty$.

Summing contributions from $x > 1$, $0 < x < 1$ and $x < 0$ gives the total result

$$T_{fi} = \delta_{ss'} \frac{g^2}{8\pi^3} \int d^2\vec{k}_\perp \log \frac{k_\perp^2 + \Lambda^2}{k_\perp^2 + \lambda^2} \tag{E.34}$$

for the Z-graph contribution to the one-loop fermion self-energy diagram. This answer can be re-written as

$$T_{fi} = \delta_{ss'} \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \frac{-\lambda^2 + 2m_e^2 x}{k_\perp^2 + x^2 m_e^2 + (1-x)\lambda^2 - i\epsilon} - (\lambda \to \Lambda) \; . \tag{E.35}$$

Note that this answer disagrees with the Z-graph answer using a naive application of the tree graph rule for including backward moving particles given in Ref. 11 and Ref. 14. Of course, their rule continues to remain valid for tree graphs.

Summing this result with that for the usual time-ordering Eq. (E.21) yields an answer identical to the Feynman rules answer Eq. (E.3), demonstrating the equivalence of using TOPTh$_\infty$ and Feynman rules for the one-loop fermion self-energy. The final answer in TOPTh$_\infty$ is just the Feynman rules answer.

Summarizing, the usual time-ordering graph gives an answer in TOPTh_∞ that diverges like Λ^2 and is equal to the usual LCPTh answer for the fermion self-energy. There are no contributions to this graph from the regions near $x = 0$ or 1. The Z-graph contribution in TOPTh_∞ only has a contribution near $x = 0$ and sums with the usual time-ordering graph to give the familiar Feynman rules answer. This final answer diverges like $\ln \Lambda$ because the leading Λ^2 divergence cancels. In order to reconcile the LCPTh and Feynman rules answers for the one-loop fermion self-energy, an extra piece equal to the TOPTh_∞ Z-graph must be added to the light-cone Hamiltonian and the LCPTh rules.

Now consider the one-loop vacuum polarization graph. The answer in TOPTh_∞ is dealt with first. There are two contributions, which are shown in Figure 45. Momenta are assigned to the first graph,

$$p = \left(E, \vec{0}_\perp, P\right) , \qquad k_1 = \left(E_1, \vec{k}_\perp, xP\right) , \qquad k_2 = \left(E_2, -\vec{k}_\perp, (1-x)P\right) ,$$

$$E = P , \qquad E_1 = \sqrt{x^2P^2 + m_\perp^2} , \qquad E_2 = \sqrt{(1-x)^2P^2 + m_\perp^2} ,$$

$$m_\perp^2 = k_\perp^2 + m_e^2 , \tag{E.36}$$

giving a TOPTh_∞ answer for the usual time-ordering of

$$\begin{aligned} T_{fi} \;=\; \lim_{P\to\infty} \frac{-g^2}{4(2\pi)^3}\, P \int_{-\infty}^{\infty} dx \int d^2\vec{k}_\perp \, \frac{1}{E_1 E_2} \, \frac{N}{D + i\epsilon} \quad , \\ N = -\bar{u}(k_1) \not{\epsilon}^* v(k_2)\; \bar{v}(k_2) \not{\epsilon} u(k_1) \; , \\ D = E - E_1 - E_2 \; . \end{aligned} \tag{E.37}$$

The numerator algebra is shown in detail.

$$\begin{aligned} N &= -\,\text{Tr}\,\{(\not{k}_1 + m_e)\not{\epsilon}^*(\not{k}_2 - m_e)\not{\epsilon}\} \\ &= -4\{(k_1\cdot\epsilon)(k_2\cdot\epsilon)^* + (k_1\cdot\epsilon)^*(k_2\cdot\epsilon) - (k_1\cdot k_2)(\epsilon\cdot\epsilon)^* - m_e^2(\epsilon\cdot\epsilon^*)\} \; . \end{aligned} \tag{E.38}$$

The light-cone gauge photon spinors are

$$\epsilon^\mu = \left(\frac{\vec{\epsilon}_\perp \cdot \vec{p}_\perp}{E+p_z} \, , \vec{\epsilon}_\perp \, , \frac{\vec{\epsilon}_\perp \cdot \vec{p}_\perp}{E+p_z} \right) \underset{\vec{p}_\perp = 0}{=} (0 \, , \vec{\epsilon}_\perp , 0) \ ,$$
$$\vec{\epsilon}_\perp(\lambda) = -\frac{\lambda}{\sqrt{2}} (1 \, , i\lambda) \tag{E.39}$$

and satisfy

$$\epsilon_\lambda \cdot \epsilon^*_{\lambda'} = -\vec{\epsilon}_{\perp\lambda} \cdot \vec{\epsilon}^*_{\perp\lambda'} = -\delta_{\lambda\lambda'} \ . \tag{E.40}$$

Returning to the evaluation of the numerator after using Eq. (E.39),

$$N = -4\{-2(\vec{\epsilon}_{\perp\lambda} \cdot \vec{k}_\perp)(\vec{\epsilon}^*_{\perp\lambda'} \cdot \vec{k}_\perp) + \delta_{\lambda\lambda'} \left[E_1 E_2 + k_\perp^2 - x(1-x)p^2 + m_e^2 \right] \} \ . \tag{E.41}$$

The first term can be expanded by writing out the form of the spinor explicitly,

$$(\vec{\epsilon}_{\perp\lambda} \cdot \vec{k}_\perp)(\vec{\epsilon}^*_{\perp\lambda'} \cdot \vec{k}_\perp) = \frac{\lambda\lambda'}{2} \ \left[k_x^2 + k_x k_y (i\lambda - i\lambda') + \lambda\lambda' k_y^2 \right] \ . \tag{E.42}$$

The rest of the integrand only depends on $k_\perp^2$, so the symmetric integration relation

$$\int d^2\vec{k}_\perp \ k^i k^j F(k_\perp^2) = \frac{1}{2} \, \delta^{ij} \int d^2\vec{k}_\perp \ k_\perp^2 F(k_\perp^2) \tag{E.43}$$

may be used to give

$$(\vec{\epsilon}_{\perp\lambda} \cdot \vec{k}_\perp)(\vec{\epsilon}^*_{\perp\lambda'} \cdot \vec{k}_\perp) = \frac{1}{2} \, \delta_{\lambda\lambda'} \ k_\perp^2 \ . \tag{E.44}$$

Substituting this result into Eq. (E.41) produces

$$N = -4 \ \delta_{\lambda\lambda'} \ \left[E_1 E_2 - x(1-x)p^2 + m_e \right] \ . \tag{E.45}$$

Using this result for the numerator yields the following expression for the usual

time-ordering contribution

$$T_{fi} = \lim_{P\to\infty} \delta_{\lambda\lambda'} \frac{g^2}{8\pi^3} \int_{-\infty}^{\infty} dx \int d^2\vec{k}_\perp \int_0^\infty g(\lambda)\, d\lambda \;\left[I(m_e^2, P) - I(m_e^2+\lambda^2, P)\right] \;,$$

$$I(m_e^2, P) = \frac{1}{\sqrt{x^2+\left(\frac{m_\perp}{P}\right)^2}\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2}} \;\frac{\sqrt{x^2+\left(\frac{m_\perp}{P}\right)^2}\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2}-x(1-x)+\left(\frac{m_e}{P}\right)^2}{1-\sqrt{x^2+\left(\frac{m_\perp}{P}\right)^2}-\sqrt{(1-x)^2+\left(\frac{m_\perp}{P}\right)^2}+i\epsilon} \;. \tag{E.46}$$

The integral over $g(\lambda)d\lambda$ is included as an ultraviolet regulator.

The remainder of the analysis is similar to that for the fermion self-energy described above. One sees that $I(m_e^2, P)$ is potentially singular near $x = 0, 1$. In the region away from these singularities, the limit $P \to \infty$ can be taken inside the x integral. Near the singularity $x = 0$, the non-singular part of I is expanded in powers of x to find the contribution to T_{fi} from this region. A similar analysis is done for x near one. The answer for the usual time-ordering contribution to one-loop vacuum polarization has the following pieces:

1.

$$-\delta_{\lambda\lambda'}\frac{g^2}{8\pi^3}\int_0^1 dx \int d^2\vec{k}_\perp \int_0^\infty g(\lambda)d\lambda \;\left[\left(\tfrac{x}{1-x}+\tfrac{1-x}{x}\right)+\tfrac{2m_e^2}{k_\perp^2+m_e^2} - (m_e^2 \to m_e^2+\lambda^2)\right] \tag{E.47}$$

from $\epsilon < x < 1-\epsilon$,

2.

$$\delta_{\lambda\lambda'}\frac{g^2}{8\pi^3}\int_0^1 dx \int d^2\vec{k}_\perp \int_0^\infty g(\lambda)\, d\lambda \;\left[\log\left(k_\perp^2+m_e^2\right) \;-\; (m_e^2 \to m_e^2+\lambda^2)\right] \tag{E.48}$$

from $-\epsilon < x < \epsilon$,

3. and an identical contribution to Eq. (E.48) from $1-\epsilon < x < 1+\epsilon$, and zero contribution from other x regions.

Summing these pieces gives the final answer for the usual time-ordering contribution,

$$T_{fi} = \delta_{\lambda\lambda'} \frac{g^2}{4\pi^3} \int_0^1 dx \int d^2\vec{k}_\perp \int_0^\infty g(\lambda)\, d\lambda \; \left[I(m_e^2) - I(m_e^2 + \lambda^2) \right] \; ,$$
$$I(m_e^2) = \log\left(k_\perp^2 + m_e^2\right) - \frac{m_e^2}{k_\perp^2 + m_e^2} \quad . \tag{E.49}$$

This answer is in fact the complete answer for one-loop vacuum polarization in TOPTh_∞ because the Z-graph turns out to be zero. One can also show by doing the $\vec{k}_\perp$ integral that this answer is in fact zero, which is the expected answer for the vacuum polarization between on-shell photons. This zero result only occurs because a subtracted regulator term $I(m_e^2 + \lambda^2, P)$ has been included that exactly cancels the contribution from the original term $I(m_e^2, P)$.

The Feynman rules answer for the one-loop vacuum polarization graph shown in Figure 46 is

$$T_{fi} = -\frac{ig^2}{(2\pi)^4} \int d^4k \; \frac{\text{Tr}\left[\not{\epsilon}(\not{p} + \not{k} + m_e)\, \not{\epsilon}^*(\not{k} + m_e)\right]}{\left[(p+k)^2 - m_e^2 + i\epsilon\right]\left(k^2 - m_e^2 + i\epsilon\right)} \quad . \tag{E.50}$$

The numerator algebra is done in a fashion paralleling the steps Eq. (E.38) through Eq. (E.44) using the light-cone spinors given in Eq. (E.39). The result after changing variables to $q^\mu = k^\mu + xp^\mu$ and eliminating terms proportional to q^μ by symmetric integration is

$$N \; = \; \text{Tr}\left[\not{\epsilon}(\not{p} + \not{k} + m_e)\, \not{\epsilon}^*(\not{k} + m_e)\right] \; = \; 4\, \delta_{\lambda\lambda'} \left[q_\perp^2 + q^2 - m_e^2\right] \; ,$$
$$T_{fi} = -\delta_{\lambda\lambda'} \frac{4ig^2}{(2\pi)^4} \int_0^1 dx \int d^4q \; \frac{q_\perp^2 + q^2 - m_e^2}{\left[q^2 - m_e^2 + i\epsilon\right]^2} \quad . \tag{E.51}$$

The denominators have already been combined and p^2 set equal zero in this last

expression for T_{fi}. The q^0 integral is done by contour integration to give

$$T_{fi} = \delta_{\lambda\lambda'} \frac{g^2}{8\pi^3} \int_0^1 dx \int d^3\vec{q} \, \frac{-2q_3^2 - q_\perp^2 - 2m_e^2}{[\vec{q}^2 + m_e^2 - i\epsilon]^{\frac{3}{2}}} \quad . \tag{E.52}$$

Invoking an ultraviolet regulator and doing the q^3 integral yields the answer for one-loop vacuum polarization,

$$\begin{aligned} T_{fi} &= \delta_{\lambda\lambda'} \frac{g^2}{4\pi^3} \int_0^1 dx \int d^2\vec{q}_\perp \int_0^\infty g(\lambda)\, d\lambda \;\; [I(m_e^2) - I(m_e^2 + \lambda^2)] \;\; , \\ I(m_e^2) &= \log\left(q_\perp^2 + m_e^2\right) + \frac{q_\perp^2}{q_\perp^2 + m_e^2} \;\; , \end{aligned} \tag{E.53}$$

which is identical to the TOPTh$_\infty$ result Eq. (E.49) after adding and subtracting $\frac{k_\perp^2 + m_e^2}{k_\perp^2 + m_e^2}$.

The numerator algebra has been done in a "non-standard" way in this analysis. Let's see what happens if the numerator algebra is done the standard way,

$$\begin{aligned} N &= \epsilon_\mu \epsilon_\nu^* \;\; \mathrm{Tr}\left[\gamma^\mu (\not{p} + \not{k} + m_e)\gamma^\nu (\not{k} + m_e)\right] \\ &= \epsilon_\mu \epsilon_\nu^* \;\; \mathrm{Tr}\left[\gamma^\mu (\not{q} + (1-x)\not{p} + m_e)\gamma^\nu (\not{q} - x\not{p} + m_e)\right] \\ &= 4\;\epsilon_\mu \epsilon_\nu^* \;\left\{ \left[q^\mu + (1-x)\,p^\mu\right]\left[q^\nu - xp^\nu\right] - g^{\mu\nu}\left[q + (1-x)\,p\right]\cdot\left[q - xp\right] \right. \\ &\qquad \left. + \left[q^\nu + (1-x)\,p^\nu\right]\left[q^\mu - xp^\mu\right] + g^{\mu\nu} m_e^2 \right\} \\ &= 4\;\epsilon_\mu \epsilon_\nu^* \;\left[-\tfrac{1}{2} g^{\mu\nu} q^2 + g^{\mu\nu} m_e^2 + g^{\mu\nu} p^2 x(1-x) - 2p^\mu p^\nu x(1-x) \right] \\ &= (2q^2 - 4m_e^2)\;\delta_{\lambda\lambda'} \;. \end{aligned} \tag{E.54}$$

Variables were changed to $q^\mu = k^\mu + xp^\mu$ in the first step, the symmetric integration relations

$$\int d^4q \; q^\mu F(q^2) = 0 \;, \qquad \int d^4q \; q^\mu q^\nu F(q^2) = \frac{1}{4}\, g^{\mu\nu} \int d^4q \; q^2 F(q^2) \tag{E.55}$$

were used in the third step, and the spinor relations $\epsilon \cdot \epsilon' = -\delta_{\lambda\lambda'}$, $\epsilon \cdot p = 0$ were

used in the fourth step. Using this answer for the numerator gives

$$T_{fi} = \delta_{\lambda\lambda'} \frac{4ig^2}{(2\pi)^4} \int_0^1 dx \int d^4q \, \frac{-\frac{1}{2}q^2 + m_e^2}{[q^2 - m_e^2 + i\epsilon]^2} \quad , \tag{E.56}$$

which differs from the "non-standard" numerator algebra answer, Eq. (E.51). Doing the q^0 integral by contours, regulating, and doing the q^3 integral gives the result

$$\begin{aligned} T_{fi} &= \delta_{\lambda\lambda'} \frac{g^2}{8\pi^3} \int_0^1 dx \int d^2\vec{q}_\perp \int_0^\infty g(\lambda)\, d\lambda \, \left[I(m_e^2) - I(m_e^2 + \lambda^2) \right] \, , \\ I(m_e^2) &= \log\left(q_\perp^2 + m_e^2\right) + \frac{q_\perp^2}{q_\perp^2 + m_e^2} \, , \end{aligned} \tag{E.57}$$

which is half of the result Eq. (E.49) or Eq. (E.53). Since all of these results are zero after doing the $\vec{q}_\perp$ integral, the discrepancy of $\frac{1}{2}$ is a zero form, $0 = \frac{1}{2}0$. The formal $\frac{1}{2}$ difference comes from setting p^2 and $\epsilon \cdot p$ equal to zero at different points in the two methods of evaluating the numerator algebra. So, the answers for the one-loop vacuum polarization graph using Feynman rules and TOPTh$_\infty$ are identical if one is careful to do the numerator algebra the same way.

APPENDIX F

In this appendix, the approximate equivalence of the light-cone variational equation,

$$\langle\psi|\, H_{LC}\,|\psi\rangle \;=\; M^2\;, \tag{F.1}$$

in the Fock space $(e^+e^-, e^+e^-\gamma)$ and the momentum space Coulomb Schroedinger equation is shown for the choice

$$\begin{aligned}
M \;&=\; M_{\rm Bohr}\;,\\
|\psi\rangle \;&=\; \psi_{e^+e^-}\,|e^+e^-\rangle \;+\; \psi_{e^+e^-\gamma}\,|e^+e^-\gamma\rangle\;,\\
\psi_{e^+e^-} \;&=\; \frac{A}{\left[\vec{k}^2 - \frac{1}{4}m_e^2\alpha^2\right]^2} \;\approx\; \frac{A'}{\left[\frac{\vec{k}_\perp^2+m_e^2}{x(1-x)} - M_{\rm Bohr}^2\right]^2}\;,\\
\psi_{e^+e^-\gamma} \;&=\; \sum_{(e^+e^-)} \frac{1}{M_{e^+e^-}^2 - M_{e^+e^-\gamma}^2}\,\langle e^+e^-\gamma|\, H_{LC}\,|e^+e^-\rangle\;\psi_{e^+e^-}\\
&\approx\; \sum_{(e^+e^-)} \frac{1}{M_{\rm Bohr}^2 - M_{e^+e^-\gamma}^2}\,\langle e^+e^-\gamma|\, H_{LC}\,|e^+e^-\rangle\;\psi_{e^+e^-}\quad .
\end{aligned} \tag{F.2}$$

This demonstration establishes the correspondence principle. The definitions of A, $M_{\rm Bohr}$, $M_{e^+e^-}$, and $M_{e^+e^-\gamma}$ are just those given in Section 11,

$$\begin{aligned}
M_{\rm Bohr} \;&=\; 2m_e - \frac{1}{4}m_e\alpha^2\;,\\
M_{e^+e^-}^2 \;&=\; \sum_{i=e^+,e^-} \frac{k_{\perp i}^2 + m_i^2}{x_i}\;,\\
M_{e^+e^-\gamma}^2 \;&=\; \sum_{i=e^+,e^-,\gamma} \frac{k_{\perp i}^2 + m_i^2}{x_i}\;.
\end{aligned} \tag{F.3}$$

First, the momentum space Coulomb Schroedinger equation is derived. We start with the familiar expression

$$\int d^3\vec{r}\;\psi^*(\vec{r})\left[\frac{\vec{k}^2}{2m_{\rm red}} - \frac{\alpha}{r}\right]\psi(\vec{r}) \;=\; -\frac{1}{2}m_{\rm red}\;\alpha^2\;. \tag{F.4}$$

This equation is Fourier transformed according to

$$\psi(\vec{k}) \;=\; \frac{1}{(2\pi)^{3/2}} \int d^3\vec{r}\; \psi(\vec{r})\; e^{i\vec{k}\cdot\vec{r}} \tag{F.5}$$

to give

$$\begin{aligned} \int d^3\vec{k}\; |\psi(\vec{k})|^2\; \frac{\vec{k}^2}{m_e} \;-\; \frac{1}{(2\pi)^3} \int d^3\vec{r}\; d^3\vec{k}_i\; d^3\vec{k}_f\; \psi^*(\vec{k}_f)\psi(\vec{k}_i)\; e^{i(\vec{k}_i-\vec{k}_f)\cdot\vec{r}}\; \frac{\alpha}{r} \\ \;=\; -\frac{1}{4} m_e \alpha^2 \;. \end{aligned} \tag{F.6}$$

Note that this choice for the Fourier transform leaves the norm of the wavefunction unchanged,

$$\int d^3\vec{r}\; |\psi(\vec{r})|^2 \;=\; \int d^3\vec{k}\; |\psi(\vec{k})|^2 \;=\; 1 \;. \tag{F.7}$$

Making use of the equality,

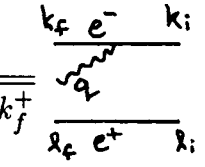
$$\int d^3\vec{r}\; \frac{\alpha}{r}\; e^{i\vec{q}\cdot\vec{r}} \;=\; \frac{e^2}{\vec{q}^2} \;, \tag{F.8}$$

results in the momentum space Coulomb Schroedinger equation we are after,

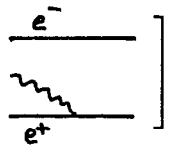
$$\int d^3\vec{k}\; |\psi(\vec{k})|^2\; \frac{\vec{k}^2}{m_e} - \frac{\alpha}{2\pi^2} \int d^3\vec{k}_i\; d^3\vec{k}_f\; \psi^*(\vec{k}_f)\psi(\vec{k}_i)\; \frac{1}{(\vec{k}_f - \vec{k}_i)^2} \;=\; -\frac{1}{4} m_e \alpha^2 \;. \tag{F.9}$$

Now turn to the light-cone variational equation. The contribution from the instantaneous fermion exchange, $V_{instferm}$, was shown in Section 13 to be small compared to the other interactions so it can be ignored. $\psi_{e^+e^-\gamma}$ may be symbolically written

$$\psi_{e^+e^-\gamma} = \sum_{k_i,l_i} \frac{1}{\sqrt{\Omega}} \frac{1}{D} \left[\frac{1}{\sqrt{q^+ k_i^+ k_f^+}} \right.$$



$$+ \frac{1}{\sqrt{q^+ l_i^+ l_f^+}}$$



$$\left. \right] \psi_{e^+e^-} \;. \tag{F.10}$$

D and Ω are defined to be

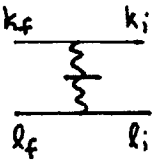
$$\begin{aligned} D \;&=\; P^-_{e^+e^-} - P^-_{e^+e^-\gamma} \;=\; \frac{1}{P^+} \left[M^2_{e^+e^-} - M^2_{e^+e^-\gamma} \right] \;, \\ \Omega \;&=\; 2L(2L_\perp)^2 \;. \end{aligned} \tag{F.11}$$

The diagrams represent the light-cone perturbation theory (LCPTh) answer for

these interactions, $g\bar{u}(k_f)\not{\epsilon}^* u(k_i)$ and $-g\bar{v}(l_f)\not{\epsilon}^* v(l_i)$. Note that whereas the incoming state is usually on the left and the outgoing on the right, in these diagrams, we have chosen to show the incoming state on the right and the outgoing on the left. The upper line in each diagram will always be the electron and the lower line the positron. The relationship $P^+ = \pi K/L$ and the fact that the DLCQ answer given in Eq. (4.24) for the three-point vertex shown in Figure 47 is equal the LCPTh answer, $g\bar{u}(k_f)\not{\epsilon}^* u(k_i)$, times a factor $\frac{\pi}{L\sqrt{\Omega}\sqrt{k_f^+ k_i^+ q^+}}$ are used. One also has that the DLCQ answer for the four-point instantaneous photon interaction shown in Figure 48 is $\frac{\pi}{L\Omega\sqrt{k_f^+ k_i^+ l_f^+ l_i^+}}$ times the LCPTh answer for this graph, $-g^2 \frac{\bar{u}(k_f)\gamma^+ u(k_i)\bar{v}(l_f)\gamma^+ v(l_i)}{(k_f^+ - k_i^+)^2}$.

The contribution to $\langle\psi|H_{LC}|\psi\rangle$ from H_2 is just

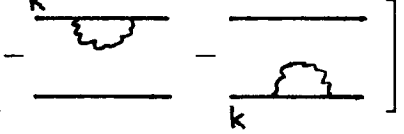
$$\langle\psi|\,H_2\,|\psi\rangle = \sum_{\substack{x_i,\vec{k}_{\perp i}\\ x_f,\vec{k}_{\perp f}}} \psi^{*f}_{e^+e^-}\psi^{i}_{e^+e^-}\frac{K\pi}{L\Omega}\frac{1}{\sqrt{k_i^+ k_f^+ l_i^+ l_f^+}}$$



. (F.12)

Similarly, the contribution from H_{self} is

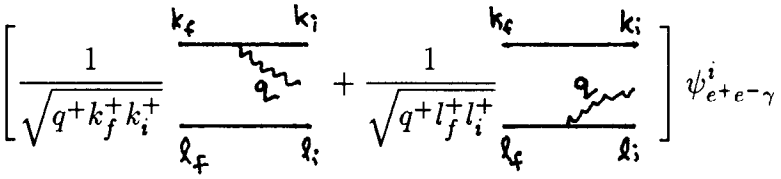
$$\langle\psi|\,H_{self}\,|\psi\rangle = \sum_{x,\vec{k}_\perp} |\,\psi_{e^+e^-}|^2 \frac{K\pi}{L\Omega k^+}$$



. (F.13)

Once again, the diagrams represent LCPTh expressions. Now consider the contribution to $\langle\psi|H_{LC}|\psi\rangle$ from H_1. It is

$$\langle\psi|\,H_1\,|\psi\rangle = \frac{K\pi}{L\sqrt{\Omega}}\psi^{f*}_{e^+e^-}$$

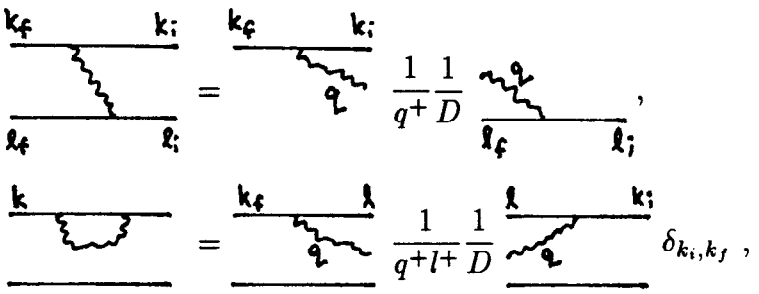


$$\psi^{i}_{e^+e^-\gamma} + \text{h.c.} \quad .$$

(F.14)

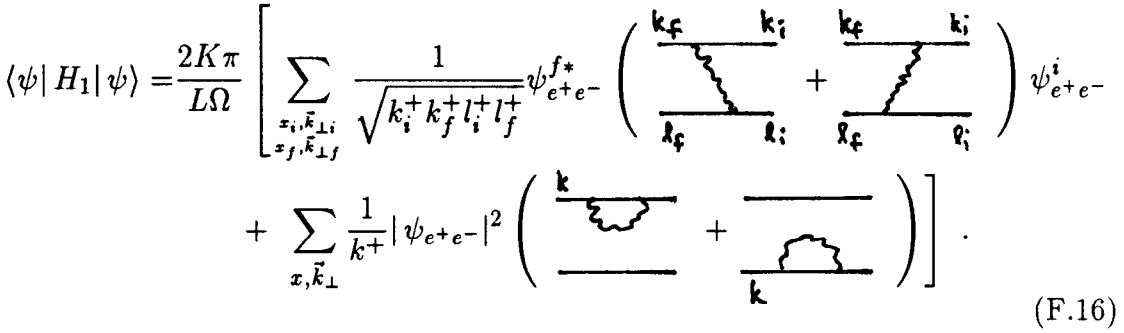
In this and upcoming expressions, a sum over $x_i, x_f, \vec{k}_{\perp i}, \vec{k}_{\perp f}$ is implied where appropriate. Using Eq. (F.10) for $\psi_{e^+e^-\gamma}$, collecting terms, and using the LCPTh

relationships



(F.15)

where D was defined in Eq. (F.11) produces



(F.16)

The factor of two comes from the hermitian conjugate term. Finally, turn to the contribution from H_0 to $\langle\psi|H_{LC}|\psi\rangle$,

$$\langle\psi|\,H_0|\,\psi\rangle \;=\; |\psi_{e^+e^-}|^2 M^2_{e^+e^-} \;+\; |\psi_{e^+e^-\gamma}|^2 M^2_{e^+e^-\gamma}\;. \tag{F.17}$$

$M_{e^+e^-}$ and $M_{e^+e^-\gamma}$ were defined previously in Eq. (F.3). Now add and subtract a term $|\psi_{e^+e^-\gamma}|^2 M^2_{e^+e^-}$ where $M^2_{e^+e^-}$ is the invariant mass squared of the (e^+e^-) states from which $\psi_{e^+e^-\gamma}$ is obtained (see definition of $\psi_{e^+e^-\gamma}$, Eq. (F.10)) to arrive at

$$\langle\psi|\,H_0|\,\psi\rangle \;=\; \left(|\psi_{e^+e^-}|^2 + |\psi_{e^+e^-\gamma}|^2\right) M^2_{e^+e^-} - |\psi_{e^+e^-\gamma}|^2 \left(M^2_{e^+e^-} - M^2_{e^+e^-\gamma}\right)\;. \tag{F.18}$$

Inserting Eq. (F.10) into the second term gives

$$-\frac{1}{\Omega}\frac{M^2_{e^+e^-}-M^2_{e^+e^-\gamma}}{D^2}\,\psi^{f*}_{e^+e^-}\left[\frac{1}{\sqrt{k_i^+k_f^+q^+}}\;+\;\frac{1}{\sqrt{l_i^+l_f^+q^+}}\right]\times\left[\frac{1}{\sqrt{k_i^+k_f^+q^+}}\;+\;\frac{1}{\sqrt{l_i^+l_f^+q^+}}\right]\psi^{i}_{e^+e^-}$$

$$=-\frac{K\pi}{L\Omega}\left[\sum_{\substack{x_i,\vec{k}_{\perp i}\\ x_f,\vec{k}_{\perp f}}}\frac{1}{\sqrt{k_i^+k_f^+l_i^+l_f^+}}\,\psi^{f*}_{e^+e^-}\left(\;+\;\right)\psi^{i}_{e^+e^-}\right.$$

$$\left.+\sum_{x,\vec{k}_\perp}\frac{1}{k^+}|\,\psi_{e^+e^-}|^2\left(\;+\;\right)\right]\quad .\tag{F.19}$$

The relation $M^2_{e^+e^-}-M^2_{e^+e^-\gamma}=P^+D$ has been used. If the norm of $\psi_{e^+e^-}$ is redefined to include the contribution to the norm from $\psi_{e^+e^-\gamma}$,

$$\sum_{x,\vec{k}_\perp}|\psi_{e^+e^-}|^2_{\rm new}\;=\;\sum_{x,\vec{k}_\perp}|\psi_{e^+e^-}|^2_{\rm old}\;+\;|\psi_{e^+e^-\gamma}|^2\;=\;1\;,\tag{F.20}$$

then first term can be approximated as

$$|\psi_{e^+e^-}|^2\;M^2_{e^+e^-}\;.\tag{F.21}$$

Collecting the contributions to $\langle\psi|H_{LC}|\psi\rangle$ from H_0, H_1, H_2, H_{self}, we find that the self-mass bubble completely cancels. What remains is

$$\langle\psi\,|\,H_{LC}\,|\psi\rangle=\sum_{x,\vec{k}_\perp}|\,\psi_{e^+e^-}|^2\,M^2_{e^+e^-}$$

$$+\sum_{\substack{x_i,\vec{k}_{\perp i}\\ x_f,\vec{k}_{\perp f}}}\frac{K\pi}{L\Omega}\frac{\psi^{f*}_{e^+e^-}\,\psi^{i}_{e^+e^-}}{\sqrt{k_i^+k_f^+l_i^+l_f^+}}\left[\;+\;+\;\right]\;.\tag{F.22}$$

Note that the real photon exchange diagrams (first two graphs in the [...]) come

from two places, H_1 and H_0. For convergence studies, it is convenient to define

$$
\begin{aligned}
PE_{flip} =& \sum_{\substack{x_i,\vec{k}_{\perp i}\\ x_f,\vec{k}_{\perp f}}} \frac{K\pi}{L\Omega}\frac{\psi^{f*}_{e^+e^-}\psi^{i}_{e^+e^-}}{\sqrt{k_i^+k_f^+l_i^+l_f^+}}\Bigg[\\
=& \frac{1}{2}\langle\psi\,|\,H_1\,|\psi\rangle_{flip} + \frac{1}{2}\langle\psi\,|\,H_{self}\,|\psi\rangle_{flip}\ , \\
PE_{noflip} =& \sum_{\substack{x_i,\vec{k}_{\perp i}\\ x_f,\vec{k}_{\perp f}}} \frac{K\pi}{L\Omega}\frac{\psi^{f*}_{e^+e^-}\psi^{i}_{e^+e^-}}{\sqrt{k_i^+k_f^+l_i^+l_f^+}}\Bigg[\\
=& \frac{1}{2}\langle\psi\,|\,H_1\,|\psi\rangle_{noflip} + \frac{1}{2}\langle\psi\,|\,H_{self}\,|\psi\rangle_{noflip}\ , \\
PE_{instphot} =& \sum_{\substack{x_i,\vec{k}_{\perp i}\\ x_f,\vec{k}_{\perp f}}} \frac{K\pi}{L\Omega}\frac{\psi^{f*}_{e^+e^-}\psi^{i}_{e^+e^-}}{\sqrt{k_i^+k_f^+l_i^+l_f^+}}\Bigg[\\
KE =& \sum_{x,\vec{k}_\perp} |\,\psi_{e^+e^-}\,|^2 M^2_{e^+e^-} - 4m_e^2 = \langle\psi\,|\,H_0\,|\psi\rangle + \frac{1}{2}\langle\psi\,|\,H_1\,|\psi\rangle - 4m_e^2\ , \\
PE =& PE_{flip} + PE_{noflip} + PE_{instphot}\ , \\
M^2 =& 4m_e^2 + PE + KE = \langle\psi\,|\,H_{LC}\,|\psi\rangle\ .
\end{aligned}
\tag{F.23}
$$

$[\ldots]_{flip}$ means that the spin-flip potential V_{flip} occurs at each of the two vertices. Note that the graph with V_{flip} at one vertex and V_{noflip} at the other is zero.

In Appendix C, we showed that the sum of the three LCPTh graphs in Eq. (F.22) is equal to

$$
e^2\ \bar{u}(k_f)\gamma_\mu u(k_i)\ \bar{v}(l_f)\gamma_\nu v(l_i)\ \frac{g^{\mu\nu}}{q^2_{FR}+i\epsilon}\ , \tag{F.24}
$$

where

$$
q^2_{FR} = \begin{cases} (k_f^+ - k_i^+)(k_f^- - k_i^-) - (\vec{k}_{\perp f} - \vec{k}_{\perp i})^2 & k_f^+ > k_i^+ \\ (l_f^+ - l_i^+)(l_f^- - l_i^-) - (\vec{l}_{\perp f} - \vec{l}_{\perp i})^2 & l_f^+ > l_i^+\ . \end{cases} \tag{F.25}
$$

In the non-relativistic regime, the largest contribution is from $\mu,\nu = 0$ since

$\bar{u}(k_f)\gamma^0 u(k_i) \gg \bar{u}(k_f)\gamma^i u(k_i)$ for $i = 1,2,3$. So, the numerator can be approximated as

$$\bar{u}(k_f)\gamma^+ u(k_i)\ \bar{v}(l_f)\gamma^+ v(k_i)\ . \tag{F.26}$$

Using this relation and $\bar{u}(p)\gamma^+ u(q) = 2\sqrt{p^+ q^+}\delta_{ss'}$ from Appendix A in Ref. 14 gives the result

$$\langle\psi\,|\,H_{LC}|\psi\rangle\ =\ \sum_{x,\vec{k}_\perp} |\psi_{e^+e^-}|^2 M^2_{e^+e^-} + \sum_{\substack{x_i,\vec{k}_{\perp i}\\ x_f,\vec{k}_{\perp f}}} \frac{4K\pi}{L\Omega}\psi^{f*}_{e^+e^-}\psi^{i}_{e^+e^-}\frac{e^2}{q^2_{FR}}\ =\ M^2_{\rm Bohr}\ . \tag{F.27}$$

This equation can be converted to the continuum by the replacement

$$\sum_x \to \frac{L}{2\pi}\int dk^+\ ,\quad \sum_{\vec{k}_\perp} \to \left(\frac{L_\perp}{\pi}\right)^2 \int d^2\vec{k}_\perp\ ,\quad \psi_{e^+e^-} \to \sqrt{\frac{2\pi}{L}}\frac{\pi}{L_\perp}\,\psi(k)\ . \tag{F.28}$$

Making these replacements produces the light-cone equation,

$$\begin{aligned}\int dk^+ d^2\vec{k}_\perp\ |\psi(k)|^2\ \frac{k^2_\perp + m^2_e}{x(1-x)}\\ +\ \frac{2\alpha m_e}{\pi^2}\int dk^+_i d^2\vec{k}_{\perp i}\ dk^+_f d^2\vec{k}_{\perp f}\ \psi^*(k_f)\psi(k_i)\ \frac{1}{q^2_{FR}}\ =\ M^2_{\rm Bohr}\ .\end{aligned} \tag{F.29}$$

The relations $\alpha = e^2/4\pi$ and $K\pi/L = P^+ \approx 2m_e$, which is approximately true for non-relativistic P^+, have been used to derive this answer. The kinetic energy can be re-written as

$$\begin{aligned}\frac{k^2_\perp + m^2_e}{x(1-x)} &= P^+\left[\frac{k^2_\perp + m^2_e}{k^+}\Big|_{e^-} + \frac{k^2_\perp + m^2_e}{k^+}\Big|_{e^+}\right]\\ &= \left(P^+_{e^-} + P^+_{e^+}\right)\left(P^-_{e^-} + P^-_{e^+}\right)\\ &= \left(P^0_{e^-} + P^3_{e^-} + P^0_{e^+} + P^3_{e^+}\right)\left(P^0_{e^-} - P^3_{e^-} + P^0_{e^+} - P^3_{e^+}\right)\\ &= \left(P^0_{e^-} + P^0_{e^+}\right)\left(P^0_{e^-} + P^0_{e^+}\right)\\ &= E^2_{\rm total} = 4\left(\vec{k}^2 + m^2_e\right)\ .\end{aligned} \tag{F.30}$$

In the fourth step, $P^3_{e^-} + P^3_{e^+}$ = total P^3 of the system equals zero was used. Substituting this result in Eq. (F.29) and approximating dk^+ as dk^3, q^2_{FR} as $-\vec{q}^2$

and $M^2_{\rm Bohr}$ as $4m_e^2 - m_e^2\alpha^2$, which are valid for nonrelativistic k^0, q^0 and small α^4 respectively, results in

$$4\int d^3\vec{k}\ |\psi(\vec{k})|^2\ \vec{k}^2\ -\ \frac{2\alpha m_e}{\pi^2}\int d^3\vec{k}_i\ d^3\vec{k}_f\ \psi^*(\vec{k}_f)\psi(\vec{k}_i)\ \frac{1}{\vec{q}^2}\ =\ -m_e^2\alpha^2\ , \quad \text{(F.31)}$$

which is identical to the momentum space Coulomb Schroedinger equation, Eq. (F.9).

APPENDIX G

In this appendix, it is shown that the light-cone variational equation,

$$\langle\psi\,|\,H_{LC}|\psi\rangle = M^2\ , \tag{G.1}$$

in the truncated Fock space $(e^-, e^-\gamma)$ is equivalent to

$$M = m_e\ . \tag{G.2}$$

The e^- and $e^-\gamma$ wavefunctions are chosen as in Section 11,

$$\begin{aligned}
|\psi\rangle &= \psi_{e^-}\,|e^-\rangle + \psi_{e^-\gamma}\,|e^-\gamma\rangle\ , \\
\psi_{e^-} &= A\ \delta(1-x)\ \delta^{(2)}(\vec{k}_\perp)\ , \\
\psi_{e^-\gamma} &= \frac{1}{m_e^2 - M_{e^-\gamma}^2}\ \langle e^-\gamma|\,H_{LC}|e^-\rangle\ \psi_{e^-}\ , \\
M_{e^-\gamma}^2 &= \sum_{i=e^-,\gamma} \frac{k_{\perp i}^2 + m_i^2}{x_i}\ .
\end{aligned} \tag{G.3}$$

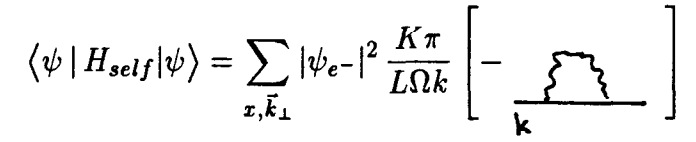
As in Appendix F, the instantaneous fermion interaction will be ignored because it is numerically small compared to the rest of the interactions. Also following Appendix F, $\psi_{e^-\gamma}$ can be symbolically written

$$\begin{aligned}
\psi_{e^-\gamma} &= \frac{1}{\sqrt{\Omega}}\frac{1}{D}\left[\frac{1}{q^+k_f^+k_i^+}\right.
\begin{array}{c}
k_f \qquad\quad k_i \\
\sim\!\sim\!\sim \\
\quad q
\end{array}
\left.\right]\ \psi_{e^-}\ , \\
D &= P_{e^-}^- - P_{e^-\gamma}^- = \frac{1}{P^+}\left[m_e^2 - M_{e^-\gamma}^2\right]\ .
\end{aligned} \tag{G.4}$$

The diagram represents the light-cone perturbation theory (LCPTh) answer for the three-point interaction, $g\bar{u}(k_f)\not{\epsilon}^*u(k_i)$.

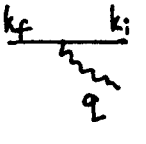
The contribution to $\langle\psi|H_{LC}|\psi\rangle$ from H_{self} is

$$\langle\psi\,|\,H_{self}|\psi\rangle = \sum_{x,\vec{k}_\perp} |\psi_{e^-}|^2 \frac{K\pi}{L\Omega k} \left[- \right] \quad . \tag{G.5}$$

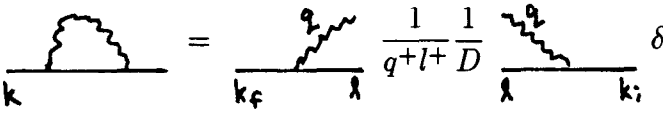


The contribution to $\langle\psi|H_{LC}|\psi\rangle$ from H_1 is

$$\langle\psi\,|\,H_1|\psi\rangle = \frac{K\pi}{L\sqrt{\Omega}} \psi^{f*}_{e^-} \left[\frac{1}{\sqrt{q^+ k_f^+ k_i^+}} \right] \psi^i_{e^-\gamma} + \text{h.c.} \quad . \tag{G.6}$$



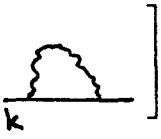
The relation that the DLCQ answer for the three-point vertex is just $\frac{\pi}{L\sqrt{\Omega}\sqrt{k_f^+ k_i^+ q^+}}$ times the LCPTh answer was used. As was the case in Appendix F, sums over x and $\vec{k}_\perp$ should be assumed wherever appropriate and outgoing states are shown to the left of a diagram and incoming states to the right. Substituting Eq. (G.4) for $\psi_{e^-\gamma}$ and using the LCPTh result,



$$\delta_{k_i,k_f} \ , \tag{G.7}$$

results in

$$\langle\psi\,|\,H_1|\psi\rangle = \sum_{x,\vec{k}_\perp} \frac{2K\pi}{L\Omega k^+} |\psi_{e^-}|^2 \left[\right] \quad . \tag{G.8}$$



The factor of two comes from the hermitian conjugate. Finally, the contribution to $\langle\psi|H_{LC}|\psi\rangle$ from H_0 is

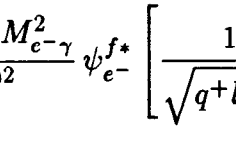
$$\langle\psi\,|\,H_0|\psi\rangle \;=\; |\psi_{e^-}|^2 m_e^2 \;+\; |\psi_{e^-\gamma}|^2 M^2_{e^-\gamma} \; . \tag{G.9}$$

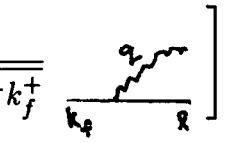
Again, as in Appendix F, a term $|\psi_{e^-\gamma}|^2 m_e^2$ is added and subtracted to give

$$\langle\psi\,|\,H_0|\psi\rangle \;=\; \left(|\psi_{e^-}|^2 + |\psi_{e^-\gamma}|^2\right) m_e^2 \;-\; |\psi_{e^-\gamma}|^2 \left(m_e^2 - M^2_{e^-\gamma}\right) \; . \tag{G.10}$$

The first term is just the norm, equal one, multiplied by m_e^2. The second term can

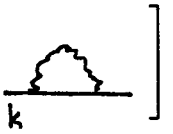
be written out using Eq. (G.4),

$$-\frac{1}{\Omega}\frac{m_e^2 - M_{e^-\gamma}^2}{D^2}\psi_{e^-}^{f*}\left[\frac{1}{\sqrt{q^+l^+k_f^+}}\right.$$


$$\left.\right]\left[\right.$$


$$\left.\frac{1}{\sqrt{q^+l^+k_i^+}}\right]\psi_{e^-}^{i} \quad . \tag{G.11}$$

The numerator $m_e^2 - M_{e^-\gamma}^2 = P^+D$ cancels one of the Ds in the denominator. Using the relation Eq. (G.7) gives

$$\langle\psi\,|\,H_0|\psi\rangle = m_e^2 - \sum_{x,\vec{k}_\perp}\frac{K\pi}{L\Omega k^+}|\psi_{e^-}|^2\left[\right.$$


$$\left.\right] \quad . \tag{G.12}$$

Summing the contributions from H_0, H_1 and H_2, we see that the one-loop self-mass bubble completely cancels, resulting in the final answer

$$\langle\psi\,|\,H_{LC}|\psi\rangle = M^2 = m_e^2 \; . \tag{G.13}$$

Note that whereas the equivalence of the light-cone variational equation and the momentum space Coulomb Schroedinger equation in the $(e^+e^-, e^+e^-\gamma)$ space shown in Appendix F was an approximate result, this result for the $(e^-, e^-\gamma)$ space is exact in the absence of the small instantaneous fermion interaction.

APPENDIX H

A set of useful spinor properties is given in this appendix.

$\bar{u}(k,s)\ldots u(k,s')$	$\uparrow\longrightarrow\uparrow$ $\downarrow\longrightarrow\downarrow$	$(s'\rightarrow s)$	$\uparrow\longrightarrow\downarrow$ $\downarrow\longrightarrow\uparrow$
$\bar{u}u$	$2m_e$		0
$\bar{u}\gamma^\mu u$	$2k^\mu$		0
$\bar{u}\gamma^+\gamma^- u$	$4m_e$		$4\left[\pm k^1 + ik^2\right]$
$\bar{u}\gamma^-\gamma^+ u$	$4m_e$		$4\left[\mp k^1 - ik^2\right]$
$\bar{u}\gamma^+\gamma^i u$	0		$2k^+\left[\pm\delta^{i1} + i\delta^{i2}\right]$
$\bar{u}\gamma^i\gamma^+ u$	0		$2k^+\left[\mp\delta^{i1} - i\delta^{i2}\right]$
$\bar{u}\gamma^-\gamma^+\gamma^- u$	$8\left[\frac{k_\perp^2+m_e^2}{k^+}\right]$		0
$\bar{u}\gamma^-\gamma^+\gamma^i u$	$4\left[k^i \mp i\epsilon^{ij}k^j\right]$		$4m_e\left[\pm\delta^{i1} + i\delta^{i2}\right]$
$\bar{u}\gamma^i\gamma^+\gamma^- u$	$4\left[k^i \pm i\epsilon^{ij}k^j\right]$		$4m_e\left[\mp\delta^{i1} - i\delta^{i2}\right]$
$\bar{u}\gamma^i\gamma^+\gamma^j u$	$2k^+\left[\delta^{ij} \pm i\epsilon^{ij}\right]$		0

$$\bar{v}(k,s)v(k,s') = -2m_e\delta_{ss'}$$

$$\bar{v}(k,s)\gamma^\mu v(k,s') = 2k^\mu\delta_{ss'}$$

$$\bar{v}(k,s)u(k,s') = \bar{u}(k,s)v(k,s') = 0$$

$$\bar{u}(k,s)\left[\gamma^\mu\gamma^\nu\gamma^\sigma + \gamma^\sigma\gamma^\nu\gamma^\mu\right]u(k,s') = \bar{v}(k,s)\left[\gamma^\mu\gamma^\nu\gamma^\sigma + \gamma^\sigma\gamma^\nu\gamma^\mu\right]v(k,s')$$
$$= \left[4g^{\mu\nu}k^\sigma - 4g^{\mu\sigma}k^\nu + 4g^{\nu\sigma}k^\mu\right]\delta_{ss'}$$

$$\bar{v}(k,s)\gamma^\mu\gamma^\nu\gamma^\sigma v(k',s') = \bar{u}(k',s')\gamma^\sigma\gamma^\nu\gamma^\mu u(k,s)$$

$$i = j = 1,2 \qquad \mu,\nu,\sigma = 0,1,2,3 \text{ or } +,-,1,2$$

APPENDIX I

The Coulomb data used to produce the fit, Eq. (13.13), is given below. All values are for $\alpha = .6$ and $\Lambda = 3.5m_e$. $L_\perp$ is in units of $\frac{1}{m_e}$.

K	$L_\perp$	$M^2 - 4m_e^2$
42	32	−0.03429031
42	36	−0.04021676
42	40	−0.04467159
42	44	−0.04791342
42	48	−0.05024107
42	52	−0.05215458
42	56	−0.05386172
42	60	−0.05514933
50	32	−0.05123963
58	32	−0.06340757
66	32	−0.07111165
74	32	−0.07938879
82	32	−0.08432295
90	32	−0.08854387
98	32	−0.09114766
106	32	−0.09459545
114	32	−0.09765245
122	32	−0.09978784

APPENDIX J

Two computer codes are presented in this appendix. The first is the program needed to implement the light-cone variational calculation described in Section 11. The results shown in Sections 12 and 13 for the charge -1 and zero sectors were produced by this code. The second program included runs what is described in Section 13 as "Coulomb data". That is, it numerically returns values for Eq. (13.10). All programs are written in Fortran.

The variational code is run by choosing values for the various parameters in the file QEDVAR.DATA. An example of this file is included. The Fortran code QEDVARIN is then run interactively. This program handles all inputs for the main code, QEDVAR. After successfully running QEDVARIN, a file named FILE20.FILE should be created. The main code QEDVAR is now run in interactive or batch mode. FILE20.FILE is the input file to QEDVAR, and QEDVAR.OUTPUT and STATES.OUTPUT are the output files. A sample of QEDVAR.OUTPUT is given.

The Coulomb code is run in a similar fashion: The initial parameters are chosen in COULOMB.DATA, COULOMBI is run first to set up inputs to the main code, the main code COULOMB is run in interactive or batch mode, the input file to COULOMB is FILE20.FILE and the output files are COULOMB.OUTPUT and STATESC.OUTPUT.

QEDVAR.DATA

```
            26    ! start value of kplustot
            26    ! end   value of kplustot
             0    ! kxtot     kxtot must be = 0
             0    ! kytot     kytot must be = 0
             0    ! icharge   ibc = 1 to keep only odd fermion kplus
             1    ! ibc       ibc = 2 to keep only even fermion kplus
            .6    ! alphag
         2.500    ! start value of alambda   in units of elecmass
         2.500    ! end   value of alambda
        20.000    ! start value of alperp    units of 1/elecmass
        20.000    ! end   value of alperp
         .0001    ! start value of epsilon   in units of elecmass**2
         .0001    ! end   value of epsilon
         1.00     ! variational parameter 1
         1.00     ! variational parameter 2
         1.00     ! variational parameter 3
         1.00     ! variational parameter 4 (not used)
         1.00     ! variational parameter 5 (not used)
          0.      ! photmass  in units of electron mass
          0.      ! rphomass      "       "       "
          1.      ! fermmass      "       "       "
          1.      ! rfermass      "       "       "
          1       ! fermions, afermions all have spin up.
          0       ! uv cut-off only applied to fermions, anti-fermions
          1       ! put in states with one photon
          0       ! remove ints. w/ afermion
          1       ! turn on h0
          1       ! turn on h1
          1       ! turn on instantaneous photon interaction
          0       ! turn on instantaneous fermion interaction
          1       ! turn on hself
          0       ! print variational fock states
          1       ! print output
!
! Notes: 1) all input in free format
!        2) photmass occurs in h0 and hself
!        3) rphomass occurs in the covariant regulator
!            (i.e.: in the generation of states
!                   and in self-induced inertias)
!        4) fermmass occurs in h0, vertex term and hself
!        5) rfermass occurs in the covariant regulator
!            (i.e.: in the generation of states
!                   and in self-induced inertias)
```

QEDVAR.OUTPUT

```
 INPUT (MASSES ARE IN UNITS OF ELECMASS):
 KPLUSTOT ICHARGE IBC =  26   0   1
 ALPHAG   =     0.6000
 ALAMBDA  =     2.5000
 ALPERP   =    20.0000*1/ELECMASS = 12.0000*BOHR  (RPERP = 15.9155)
 EPSILON  =     0.0001
 PHOTMASS =     0.0000
 RPHOMASS =     0.0000
 FERMMASS =     1.0000
 RFERMASS =     1.0000
 VAR PARAMETERS =  1.0000  1.0000  1.0000  1.0000  1.0000

 # OF FOCK STATES WITH NO PHOTONS =      363
 # OF FOCK STATES                =     7651

 CPU TIME TO FIND FOCK STATES     =    0.26 SEC
 CPU TIME TO WORK OUT VAR WF      =    4.05 SEC
 CPU TIME TO FIND HO    MATRIX EL =    0.05 SEC
 CPU TIME TO FIND H1    MATRIX EL =    4.75 SEC
 CPU TIME TO FIND H2    MATRIX EL =    0.42 SEC
 CPU TIME TO FIND HSELF MATRIX EL =    0.31 SEC
 TOTAL CPU TIME USED              =    9.84 SEC

 CONTRIBUTION TO M**2 FROM     HO =   4.3599108941
               LONG VERTEX = H1L =   0.0000000000
              TRANS VERTEX = H1T =  -0.2277568924
                          INSPHOT =  -0.3626713082
                          INSFERM =   0.0000000000
   LONG PART OF 1 LOOP SE = SEF1L =   0.0000000000
  TRANS PART OF 1 LOOP SE = SEF1T =   0.3518339387
        N CHAINED INST SE =  SEF2 =   0.0000000000
        2 CHAINED INST SE =  SEF3 =   0.0000000000

          LONG PHOT = .5H1L+SEF1L =   0.0000000000
         TRANS PHOT = .5H1T+SEF1T =   0.2379554926
          INST PHOT =     INSPHOT =  -0.3626713082
                 PE = L+T+I PHOT =  -0.1247158157
                 KE =  HO+.5H1-4 =   0.2460324479
      SUM+4 = HO+H1+INSPHOT+SEF1 =   4.1213166323

              SUM+4+INSFERM+SEF2 =   4.1213166323
              SUM+4+INSFERM+SEF3 =   4.1213166323


 FOCK STATE DECOMPOSITION: 94.19% 1 PAIR,0 PHOT  5.81% 1 PAIR,1 PHOT
```

STRUCTURE FUNCTION:

NOTES: 1) VALUES SHOULD BE MULTIPLIED BY 1/10000 (***=10000)

X=	2692	3462	4231	5000	5769	6538	7308
	51	463	2289	4654	2127	373	43

WAVE FUNCTION SQUARED AT KY=0:

NOTES: 1) VALUES SHOULD BE MULTIPLIED BY 1/10000 (***=10000)
2) KX IS IN UNITS OF ELECTRON MASS

6283	0	0	1	1	1	0	0
4712	0	3	8	9	5	2	0
3141	2	14	43	62	35	9	1
1570	4	34	210	461	203	31	3
0	5	53	475	1259	467	52	4
-1570	4	34	210	461	203	31	3
-3141	2	14	43	62	35	9	1
-4712	0	3	8	9	5	2	0
-6283	0	0	1	1	1	0	0
KX X	2692	3462	4231	5000	5769	6538	7308

COULOMB.DATA

```
            26    ! start value of kplustot
            26    ! end   value of kplustot
             0    ! kxtot     kxtot must be = 0
             0    ! kytot     kytot must be = 0
             0    ! icharge   ibc = 1 to keep only odd fermion kplus
             1    ! ibc       ibc = 2 to keep only even fermion kplus
            .6    ! alphag
         2.500    ! start value of alambda   in units of elecmass
         2.500    ! end   value of alambda
        20.000    ! start value of alperp    units of 1/elecmass
        20.000    ! end   value of alperp
         .0001    ! start value of epsilon   in units of elecmass**2
         .0001    ! end   value of epsilon
         1.00     ! variational parameter 1
         1.00     ! variational parameter 2
         1.00     ! variational parameter 3
         1.00     ! variational parameter 4 (not used)
         1.00     ! variational parameter 5 (not used)
          0.      ! photmass  in units of electron mass
          0.      ! rphomass      "       "       "
          1.      ! fermmass      "       "       "
          1.      ! rfermass      "       "       "
          0       ! uv cut-off only applied to fermions, anti-fermions
          0       ! print fock states
          1       ! print output
!
! Notes: 1) all input in free format
!        2) photmass occurs in h0 and hself
!        3) rphomass occurs in the covariant regulator
!            (i.e.: in the generation of states
!                   and in self-induced inertias)
!        4) fermmass occurs in h0, vertex term and hself
!        5) rfermass occurs in the covariant regulator
!            (i.e.: in the generation of states
!                   and in self-induced inertias)
```

```
COULOMB.OUTPUT
--------------

 INPUT (MASSES ARE IN UNITS OF ELECMASS):
 KPLUSTOT ICHARGE IBC =  26   0   1
 ALPHAG   =     0.6000
 ALAMBDA  =     2.5000
 ALPERP   =    20.0000*1/ELECMASS = 12.0000*BOHR  (RPERP = 15.9155)
 EPSILON  =     0.0001
 PHOTMASS =     0.0000
 RPHOMASS =     0.0000
 FERMMASS =     1.0000
 RFERMASS =     1.0000
 VAR PARAMETERS =  1.0000  1.0000  1.0000  1.0000  1.0000

 # OF FOCK STATES             =      363

 CPU TIME TO FIND FOCK STATES =     0.09 SEC
 CPU TIME TO WORK OUT WF      =     0.01 SEC
 CPU TIME TO FIND KE          =     0.00 SEC
 CPU TIME TO FIND PE          =     0.54 SEC
 TOTAL CPU TIME USED          =     0.64 SEC

                         KE =   0.2829467565
                         PE =  -0.1481503188
                     ENERGY =   0.1347964378


 WAVE FUNCTION SQUARED AT KY=0:

 NOTES: 1) VALUES SHOULD BE MULTIPLIED BY 1/10000 (***=10000)
        2) KX IS IN UNITS OF ELECTRON MASS

  6283 |    0    0    1    1    1    0    0
  4712 |    0    2    6    8    6    2    0
  3141 |    1    9   36   62   36    9    1
  1570 |    3   32  211  469  211   32    3
     0 |    5   55  476 1261  476   55    5
 -1570 |    3   32  211  469  211   32    3
 -3141 |    1    9   36   62   36    9    1
 -4712 |    0    2    6    8    6    2    0
 -6283 |    0    0    1    1    1    0    0
       |-------------------------------------
   KX  X 2692 3462 4231 5000 5769 6538 7308
```

QEDVARIN

```
C-------------------------------------------------------
C     THIS ROUTINE HANDLES INPUT, VARIABLE PARMETERS FOR QEDVAR.
C     IT GENERATES INPUT FILES FOR THE ROUTINE QEDVAR.
C     INPUT PARAMETERS ARE READ FROM THE FILE QEDVAR DATA.
C     MASSES ARE IN UNITS OF ELECMASS.
C     LENGTHS ARE IN UNITS OF 1/ELECMASS.
C
C     IBC = 1 MEANS KEEP ONLY ODD FERMION KPLUS
C         = 2 MEANS KEEP ONLY EVEN FERMION KPLUS
C
C     FERMIONS ASSUMED TO HAVE CHARGE -1.
C
C     CODES HAVE BEEN VECTORIZED ON AN IBM 3090 FORTRAN COMPILER.
C-------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

C-------------------------------------------------------
C     OPEN FILES.
C-------------------------------------------------------
      OPEN (UNIT=13,FILE='qed.data',STATUS='UNKNOWN')
      OPEN (UNIT=20,FILE='file20.file',STATUS='UNKNOWN')

      PI = 3.141592653589793D0

      WRITE (6,900)
 900  FORMAT (' WANT TO SEE INTRODUCTION?  TYPE 1 IF YES, 0 IF NO')
      READ (5,*) INTRO

      IF (INTRO .EQ. 1) THEN
        WRITE (6,901)
 901    FORMAT
     $(' /-------------------------------------------------------\'/
     $ ' |  WELCOME TO 3+1 VARIATIONAL QED IN DISCRETE LIGHT-CONE    |'/
     $ ' |  QUANTIZATION.  DO THE FOLLOWING TO RUN THIS PROGRAM.     |'/
     $ ' |  TO RUN PROGRAM "PROGRAM"                                 |'/
     $ ' |     A) ON IBM, JUST TYPE "PROGRAM",                       |'/
     $ ' |     B) ON DEC, TYPE "@PROGRAM".                           |'/
     $ ' |                                                           |'/
     $ ' |  1) RUN "QEDVARIN" (INTERACTIVE).                         |'/
     $ ' |     THIS ROUTINE HANDLES INPUT PARAMETERS SUCH AS         |'/
     $ ' |     PHOTMASS, FERMMASS, G.... AND SETS UP INPUT           |'/
     $ ' |     FILES FOR THE ROUTINE QEDVAR.  INPUT PARAMETERS       |'/
     $ ' |     ARE READ FROM THE FILE QEDVAR DATA.                   |')
        WRITE (6,902)
 902    FORMAT
     $(' |                                                           |'/
     $ ' |  2) RUN "QEDVAR".  THIS ROUTINES CALLS THE SUBROUTINES,   |'/
     $ ' |     IN ORDER, STATESNR, VARWF, HONR, H1NR, H2NR, HSELFNR, |'/
     $ ' |     PRINTOUT.                                             |'/
     $ ' |                                                           |'/
     $ ' |     STATESNR FINDS THE FOCK STATES CONSISTENT WITH THE    |'/
```

```
$ ' |     VALUES OF KPLUSTOT, ALAMBDA, ALPERP.                    |'/
$ ' |                                                             |'/
$ ' |     VARWF FINDS THE VARIATIONAL WAVE FUNCTION FOR THE       |'/
$ ' |     ABOVE FOCK STATES.                                      |'/
$ ' |                                                             |'/
$ ' |     HONR FINDS THE VALUE OF HO BETWEEN THESE VAR STATES.    |'/
$ ' |                                                             |'/
$ ' |     H1NR FINDS THE VALUE OF H1 BETWEEN THESE VAR STATES.    |'/
$ ' |                                                             |'/
$ ' |     H2NR FINDS THE VALUE OF H2 BETWEEN THESE VAR STATES.    |'/
$ ' |                                                             |'/
$ ' |     HSELFNR FINDS THE VALUE OF HSELF BETWEEN THESE          |'/
$ ' |     VAR STATES.                                             |'/
$ ' |                                                             |'/
$ ' |     PRINTOUT PRINTS OUT THE RESULTS.                        |'/
$ ' |                                                             |'/
$ ' |  NOTE 1: TO RUN ON DEC, UNCOMMENT SECTION FOLLOWING         |'/
$ ' |          "FOLLOWING NEEDED FOR DEC" IN QEDVAR, QEDVARIN.    |'/
$ ' |  NOTE 2: ABOVE CODES HAVE BEEN VECTORIZED                   |'/
$ ' |          ON AN IBM 3090 FORTRAN COMPILER.                   |'/
$ ' \-------------------------------------------------------------/')
 ENDIF

C-------------------------------------------------------
C     READ INPUT DATA.
C     NOTE: COVARIANT CUT-OFF SCHEME PRESENTLY ASSUMES KXTOT,KYTOT = 0.
C           CAN GET OTHER VALUES FOR KXTOT,KYTOT BY BOOSTING
C           (SEE NOTES ON BOUND STATES).
C     MASSES ARE IN UNITS OF ELECMASS.
C     LENGTHS ARE IN UNITS OF 1/ELECMASS.
C-------------------------------------------------------
      READ (13,*) KPLUSBEG
      READ (13,*) KPLUSEND
      READ (13,*) KXTOT
      READ (13,*) KYTOT
      READ (13,*) ICHARGE
      READ (13,*) IBC
      READ (13,*) ALPHAG
      READ (13,*) ALAMBBEG
      READ (13,*) ALAMBEND
      READ (13,*) ALPERBEG
      READ (13,*) ALPEREND
      READ (13,*) EPSILBEG
      READ (13,*) EPSILEND
      READ (13,*) PARA1
      READ (13,*) PARA2
      READ (13,*) PARA3
      READ (13,*) PARA4
      READ (13,*) PARA5
      READ (13,*) PHOTMASS
      READ (13,*) RPHOMASS
      READ (13,*) FERMMASS
      READ (13,*) RFERMASS
      READ (13,*) IFERMUP
      READ (13,*) IUVFERM
```

```
      READ (13,*) IONEPHOT
      READ (13,*) NOAFERI
      READ (13,*) IHO
      READ (13,*) IH1
      READ (13,*) IH2PHOT
      READ (13,*) IH2FERM
      READ (13,*) IHSELF
      READ (13,*) IPRIWF
      READ (13,*) IPRIOUT

C-----------------------------------------------------------
C     CHECK IF VALUES OF INPUT DATA ARE O.K.
C-----------------------------------------------------------
      IF (KPLUSBEG .LE. 0) THEN
        WRITE (6,920)
 920    FORMAT (' MSG FROM QEDVARIN: KPLUSBEG MUST BE .GT. 0')
        STOP
      ENDIF

      IF (KPLUSEND .LE. 0) THEN
        WRITE (6,921)
 921    FORMAT (' MSG FROM QEDVARIN: KPLUSEND MUST BE .GT. 0')
        STOP
      ENDIF

      IF ((IBC .NE. 1) .AND.
     $    (IBC .NE. 2)) THEN
        WRITE (6,922)
 922    FORMAT (' MSG FROM QEDVARIN: IBC MUST BE 1 OR 2')
        STOP
      ENDIF

      IF ((ICHARGE .NE.  0) .AND.
     $    (ICHARGE .NE. -1)) THEN
        WRITE (6,923)
 923    FORMAT (' MSG FROM QEDVARIN: ICHARGE MUST BE 0 OR -1')
        STOP
      ENDIF

      IF ((IBC .EQ. 2) .AND. (MOD(KPLUSBEG,2) .EQ. 1)) THEN
        WRITE (6,924)
 924    FORMAT(' MSG FROM QEDVARIN: KPLUSBEG MUST BE EVEN IF IBC = 2')
        STOP
      ENDIF

      IF ((IBC .EQ. 2) .AND. (MOD(KPLUSEND,2) .EQ. 1)) THEN
        WRITE (6,925)
 925    FORMAT(' MSG FROM QEDVARIN: KPLUSEND MUST BE EVEN IF IBC = 2')
        STOP
      ENDIF

      IF ((ICHARGE .EQ. 0) .AND. (MOD(KPLUSBEG,2) .EQ. 1)) THEN
        WRITE (6,926)
 926    FORMAT (' MSG FROM QEDVARIN:',
     $          ' KPLUSBEG MUST BE EVEN IF ICHARGE = 0')
```

```
        STOP
      ENDIF

      IF ((ICHARGE .EQ. 0) .AND. (MOD(KPLUSEND,2) .EQ. 1)) THEN
        WRITE (6,927)
  927   FORMAT (' MSG FROM QEDVARIN:',
     $          ' KPLUSEND MUST BE EVEN IF ICHARGE = 0')
        STOP
      ENDIF

      IF ((IBC .EQ. 1) .AND. (ICHARGE .EQ. -1) .AND.
     $    (MOD(KPLUSBEG,2) .EQ. 0)) THEN
        WRITE (6,928)
  928   FORMAT (' MSG FROM QEDVARIN:',
     $          ' KPLUSBEG MUST BE ODD IF IBC = 1 AND ICHARGE = -1')
        STOP
      ENDIF

      IF ((IBC .EQ. 1) .AND. (ICHARGE .EQ. -1) .AND.
     $    (MOD(KPLUSEND,2) .EQ. 0)) THEN
        WRITE (6,929)
  929   FORMAT (' MSG FROM QEDVARIN:',
     $          ' KPLUSEND MUST BE ODD IF IBC = 1 AND ICHARGE = -1')
        STOP
      ENDIF

      IF ((KXTOT .NE. 0) .OR.
     $    (KYTOT .NE. 0)) THEN
        WRITE (6,930)
  930   FORMAT (' MSG FROM QEDVARIN: KXTOT AND KYTOT MUST BE'/
     $          ' EQUAL TO ZERO.  OTHER VALUES CAN BE OBTAINED'/
     $          ' BY LORENTZ BOOSTING.')
        STOP
      ENDIF

      IF (EPSILON .LT. 0.0D0) THEN
        WRITE (6,931)
  931   FORMAT (' MSG FROM QEDVARIN: EPSIOON MUST BE .GE. 0.0D0')
        STOP
      ENDIF

      IF ((IFERMUP .NE. 0) .AND.
     $    (IFERMUP .NE. 1)) THEN
        WRITE (6,932)
  932   FORMAT (' MSG FROM QEDVARIN: IFERMUP MUST BE 0 OR 1')
        STOP
      ENDIF

      IF ((IUVFERM .NE. 0) .AND.
     $    (IUVFERM .NE. 1)) THEN
        WRITE (6,933)
  933   FORMAT (' MSG FROM QEDVARIN: IUVFERM MUST BE 0 OR 1')
        STOP
      ENDIF
```

```
      IF ((IONEPHOT .NE. 0) .AND.
     $    (IONEPHOT .NE. 1)) THEN
        WRITE (6,934)
  934   FORMAT (' MSG FROM QEDVARIN: IONEPHOT MUST BE 0 OR 1')
        STOP
      ENDIF

      IF ((NOAFERI .NE. 0) .AND.
     $    (NOAFERI .NE. 1)) THEN
        WRITE (6,935)
  935   FORMAT (' MSG FROM QEDVARIN: NOAFERI MUST BE 0 OR 1')
        STOP
      ENDIF

      IF ((IHO .NE. 0) .AND.
     $    (IHO .NE. 1)) THEN
        WRITE (6,937)
  937   FORMAT (' MSG FROM QEDVARIN: IHO MUST BE 0 OR 1')
        STOP
      ENDIF

      IF ((IH1 .NE. 0) .AND.
     $    (IH1 .NE. 1)) THEN
        WRITE (6,938)
  938   FORMAT (' MSG FROM QEDVARIN: IH1 MUST BE 0 OR 1')
        STOP
      ENDIF

      IF ((IH2PHOT .NE. 0) .AND.
     $    (IH2PHOT .NE. 1)) THEN
        WRITE (6,939)
  939   FORMAT (' MSG FROM QEDVARIN: IH2PHOT MUST BE 0 OR 1')
        STOP
      ENDIF

      IF ((IH2FERM .NE. 0) .AND.
     $    (IH2FERM .NE. 1)) THEN
        WRITE (6,940)
  940   FORMAT (' MSG FROM QEDVARIN: IH2FERM MUST BE 0 OR 1')
        STOP
      ENDIF

      IF ((IHSELF .NE. 0) .AND.
     $    (IHSELF .NE. 1)) THEN
        WRITE (6,941)
  941   FORMAT (' MSG FROM QEDVARIN: IHSELF MUST BE 0 OR 1')
        STOP
      ENDIF

      IF ((IH1 .EQ. 1) .AND.
     $    (IONEPHOT .NE. 1)) THEN
        WRITE (6,942)
  942   FORMAT (' MSG FROM QEDVARIN: IONEPHOT MUST = 1 IF IH1 = 1')
        STOP
      ENDIF
```

```
      IF ((IH2FERM .EQ. 1) .AND.
     $    (IONEPHOT .NE. 1)) THEN
        WRITE (6,943)
 943    FORMAT (' MSG FROM QEDVARIN: IONEPHOT MUST = 1 IF IH2FERM = 1')
        STOP
      ENDIF

C-------------------------------------------------------------
C     GENERATE PARAMETERS FILE.
C-------------------------------------------------------------
      WRITE (20,954) KPLUSBEG,KPLUSEND,KXTOT,KYTOT,ICHARGE,IBC
      WRITE (20,955) ALPHAG
      WRITE (20,955) ALAMBBEG,ALAMBEND
      WRITE (20,955) ALPERBEG,ALPEREND
      WRITE (20,955) EPSILBEG,EPSILEND
      WRITE (20,955) PARA1
      WRITE (20,955) PARA2
      WRITE (20,955) PARA3
      WRITE (20,955) PARA4
      WRITE (20,955) PARA5
      WRITE (20,955) PHOTMASS
      WRITE (20,955) RPHOMASS
      WRITE (20,955) FERMMASS
      WRITE (20,955) RFERMASS
      WRITE (20,954) IHO,IH1,IH2PHOT,IH2FERM,IHSELF
      WRITE (20,954) IFERMUP,IUVFERM,IONEPHOT,NOAFERI,IPRIWF,IPRIOUT
 954  FORMAT (6I8)
 955  FORMAT (2D30.22)

      STOP
      END
```

```
QEDVAR
------

C-------------------------------------------------------
C     THIS ROUTINE EVALUATES THE HAMILTONIAN MATRIX BETWEEN
C     VARIATIONAL STATES.
C
C     P_MINUS IS DEFINED TO BE L/PI*HAMILTONIAN.
C
C     MASSES ARE IN UNITS OF ELECMASS.
C     LENGTHS ARE IN UNITS OF 1/ELECMASS.
C
C     FERMIONS ASSUMED TO HAVE CHARGE -1.
C-------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(75555),AIMWF(75555),
     $  NPHOT(75555),NFERM(75555),NAFER(75555),
     $  KPHOT(75555),KXPHOT(75555),KYPHOT(75555),ISPHOT(75555),
     $  KFERM(75555),KXFERM(75555),KYFERM(75555),ISFERM(75555),
     $  KAFER(75555),KXAFER(75555),KYAFER(75555),ISAFER(75555)

C-------------------------------------------------------
C     FOLLOWING ARRAYS NEEDED IN VARWF, H1NR, H2NR FOR VECTORIZATION.
C-------------------------------------------------------
      DIMENSION WORK1(75555),WORK2(75555),WORK3(75555),
     $  TEMPWF(1222)

C-------------------------------------------------------
C     FOLLOWING ARRAY NEEDED IN SUBROUTINE PRINTOUT.
C-------------------------------------------------------
      DIMENSION WAVEFCN(222)

      LOGICAL TOOMANY

C-------------------------------------------------------
C     OPEN FILES.
C-------------------------------------------------------
      OPEN (UNIT=14,FILE='states.output',STATUS='UNKNOWN')
      OPEN (UNIT=15,FILE='qedvar.output',STATUS='UNKNOWN')
      OPEN (UNIT=16,FILE='qedvar.diagnose',STATUS='UNKNOWN')
      OPEN (UNIT=20,FILE='file20.file',STATUS='UNKNOWN')
      OPEN (UNIT=21,FILE='longphot.file',STATUS='UNKNOWN')
      OPEN (UNIT=22,FILE='tranphot.file',STATUS='UNKNOWN')
      OPEN (UNIT=23,FILE='instphot.file',STATUS='UNKNOWN')
      OPEN (UNIT=24,FILE='pe.file',STATUS='UNKNOWN')
      OPEN (UNIT=25,FILE='ke.file',STATUS='UNKNOWN')
      OPEN (UNIT=26,FILE='energy.file',STATUS='UNKNOWN')

C-------------------------------------------------------
C     DIM(REALWF,AIMWF,NPHOT,...) = NSIZE.  NSIZE SHOULD BE CHOSEN
C     .GE. THE NUMBER OF FOCK STATES (NSTATES).
C     DIM(TEMPWF) = NSIZEO.  NSIZEO SHOULD BE CHOSEN
C     .GE. THE NUMBER OF FOCK STATES WITH NO PHOTONS (NSTATEOP).
```

```
C     DIM(WAVEFCN) = KPLUSMAX.  KPLUSMAX SHOULD BE CHOSEN
C     .GE. KPLUSTOT.
C-------------------------------------------------------
      NSIZE = 75555
      NSIZEO = 1222
      KPLUSMAX = 222
      IF (KPLUSEND .GT. KPLUSMAX) THEN
        WRITE (15,900) KPLUSMAX
 900    FORMAT(' KPLUSTOT .GT. KPLUSMAX =',I5/
     $         ' RE-COMPILE QEDVAR WITH LARGER VALUE OF KPLUSMAX.')
        STOP
      ENDIF

      PI = 3.14159265358979300

C-------------------------------------------------------
C     READ PARAMETERS FILE (FILE20).
C-------------------------------------------------------
      READ (20,904) KPLUSBEG,KPLUSEND,KXTOT,KYTOT,ICHARGE,IBC
      READ (20,905) ALPHAG
      READ (20,905) ALAMBBEG,ALAMBEND
      READ (20,905) ALPERBEG,ALPEREND
      READ (20,905) EPSILBEG,EPSILEND
      READ (20,905) PARA1
      READ (20,905) PARA2
      READ (20,905) PARA3
      READ (20,905) PARA4
      READ (20,905) PARA5
      READ (20,905) PHOTMASS
      READ (20,905) RPHOMASS
      READ (20,905) FERMMASS
      READ (20,905) RFERMASS
      READ (20,904) IHO,IH1,IH2PHOT,IH2FERM,IHSELF
      READ (20,904) IFERMUP,IUVFERM,IONEPHOT,NOAFERI,IPRIWF,IPRIOUT
 904  FORMAT (6I8)
 905  FORMAT (2D30.22)

      WRITE (21,911) ALPHAG
 911  FORMAT (' ALPHAG =',F9.4/
     $   ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON     LONG PHOT'/
     $   ' ----------------------------------------------------')
      WRITE (22,912) ALPHAG
 912  FORMAT (' ALPHAG =',F9.4/
     $   ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON     TRAN PHOT'/
     $   ' ----------------------------------------------------')
      WRITE (23,913) ALPHAG
 913  FORMAT (' ALPHAG =',F9.4/
     $   ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON     INST PHOT'/
     $   ' ----------------------------------------------------')
      WRITE (24,914) ALPHAG
 914  FORMAT (' ALPHAG =',F9.4/
     $   ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON        PE    '/
     $   ' ----------------------------------------------------')
      WRITE (25,915) ALPHAG
 915  FORMAT (' ALPHAG =',F9.4/
```

```
     $   ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON        KE    '/
     $   ' ------------------------------------------------------')
      WRITE (26,916) ALPHAG
  916 FORMAT (' ALPHAG =',F9.4/
     $   ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON      ENERGY  '/
     $   ' ------------------------------------------------------')

      DO 100 KPLUSTOT = KPLUSBEG,KPLUSEND,4
      DO 100 ALAMBDA = ALAMBBEG,ALAMBEND,.05
      DO 100 ALPERP = ALPERBEG,ALPEREND,2.0
      DO 100 EPSILON = EPSILBEG,EPSILEND,.01

C-------------------------------------------------------
C     DETERMINE START CPU TIME.
C-------------------------------------------------------
        CALL VTTIME(IVIRTIME,ITOTTIME)
        START = DFLOAT(ITOTTIME)/100.0D0

C-------------------------------------------------------
C     CALL SUBROUTINE STATESNR TO GENERATE STATES CONSISTENT
C     WITH K, ALPERP, ALAMBDA.
C-------------------------------------------------------
        CALL STATESNR(NSIZE,KPLUSTOT,IBC,ICHARGE,ALAMBDA,ALPERP,EPSILON,
     $                RPHOMASS,RFERMASS,IFERMUP,IUVFERM,IONEPHOT,
     $                NSTATES,NSTATEOP,TOOMANY,
     $                NPHOT,NFERM,NAFER,
     $                KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $                KFERM,KXFERM,KYFERM,ISFERM,
     $                KAFER,KXAFER,KYAFER,ISAFER)
        IF (TOOMANY) THEN
          WRITE (15,918) NSIZE
  918     FORMAT (' NUMBER OF STATES GENERATED BY SUBROUTINE',
     $            ' STATESNR .GT. NSIZE =',I5/
     $            ' RE-COMPILE QEDVAR WITH LARGER VALUE OF NSIZE')
          STOP
        ENDIF
        IF (NSTATEOP .GT. NSIZEO) THEN
          WRITE (15,919) NSIZEO
  919     FORMAT (' NUMBER OF STATES WITH NO PHOTONS GENERATED BY'/
     $            ' SUBROUTINE STATESNR .GT. NSIZEO =',I5/
     $            ' RE-COMPILE QEDVAR WITH LARGER VALUE OF NSIZEO')
          STOP
        ENDIF

        CALL VTTIME(IVIRTIME,ITOTTIME)
        TIME1 = DFLOAT(ITOTTIME)/100.0D0-START

C-------------------------------------------------------
C     CALL SUBROUTINE VARWF TO WORK OUT VARIATIONAL WAVE-FUNCTION.
C-------------------------------------------------------
        CALL VARWF(REALWF,AIMWF,NSIZE,NSIZEO,NSTATES,NSTATEOP,
     $             ALPHAG,KPLUSTOT,ICHARGE,ALPERP,
     $             PHOTMASS,FERMMASS,IFERMUP,NOAFERI,
     $             PARA1,PARA2,PARA3,PARA4,PARA5,
     $             NPHOT,NFERM,NAFER,
```

```
     $            KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $            KFERM,KXFERM,KYFERM,ISFERM,
     $            KAFER,KXAFER,KYAFER,ISAFER,
     $            WORK1,WORK2,WORK3,TEMPWF)

C************DIAGNOSTICS: PRINT OUT RESULTS FROM VARWF
      RPERP = ALAMBDA*ALPERP/PI
      XALPERP = ALPERP*ALPHAG

      WRITE (14,920) KPLUSTOT,ICHARGE,IBC,
     $  ALPHAG,ALAMBDA,ALPERP,XALPERP,RPERP,EPSILON,PHOTMASS,RPHOMASS,
     $  FERMMASS,RFERMASS,NSTATEOP,NSTATES
  920 FORMAT (' INPUT (MASSES ARE IN UNITS OF ELECMASS):'/
     $        ' KPLUSTOT ICHARGE IBC =',3I4/
     $        ' ALPHAG   =',F11.4/
     $        ' ALAMBDA  =',F11.4/
     $        ' ALPERP   =',F11.4,'*1/ELECMASS =',F8.4,
     $                    '*BOHR  (RPERP =',F8.4,')'/
     $        ' EPSILON  =',F11.4/
     $        ' PHOTMASS =',F11.4/
     $        ' RPHOMASS =',F11.4/
     $        ' FERMMASS =',F11.4/
     $        ' RFERMASS =',F11.4/' '/
     $        ' # OF FOCK STATES WITH NO PHOTONS =',I9/
     $        ' # OF FOCK STATES                 =',I9)

      IF (IPRIWF .EQ. 1) THEN
        WRITE (14,921)
  921   FORMAT (' '/
     $  '       ',
     $  '    PHOTON    |    FERMION   |   AFERMION   |               '/
     $  ' STATE ',
     $  ' K+  KX  KY  S| K+  KX  KY  S| K+  KX  KY  S| REALWF  AIMWF '/
     $  ' ------',
     $  '--------------|--------------|--------------|--------------')
        DO 20 ISTATE=1,NSTATES
          IF ((NPHOT(ISTATE) .EQ. 1) .AND. (NAFER(ISTATE) .EQ. 1))
     $     WRITE (14,922) ISTATE,
     $     KPHOT(ISTATE),KXPHOT(ISTATE),KYPHOT(ISTATE),ISPHOT(ISTATE),
     $     KFERM(ISTATE),KXFERM(ISTATE),KYFERM(ISTATE),ISFERM(ISTATE),
     $     KAFER(ISTATE),KXAFER(ISTATE),KYAFER(ISTATE),ISAFER(ISTATE),
     $     REALWF(ISTATE),AIMWF(ISTATE)
  922      FORMAT (I6,3(1X,I3,1X,I3,1X,I3,1X,I2),2(1X,F7.4))
          IF ((NPHOT(ISTATE) .EQ. 1) .AND. (NAFER(ISTATE) .EQ. 0))
     $     WRITE (14,923) ISTATE,
     $     KPHOT(ISTATE),KXPHOT(ISTATE),KYPHOT(ISTATE),ISPHOT(ISTATE),
     $     KFERM(ISTATE),KXFERM(ISTATE),KYFERM(ISTATE),ISFERM(ISTATE),
     $     REALWF(ISTATE),AIMWF(ISTATE)
  923      FORMAT (I6,2(1X,I3,1X,I3,1X,I3,1X,I2),15X,2(1X,F7.4))
          IF ((NPHOT(ISTATE) .EQ. 0) .AND. (NAFER(ISTATE) .EQ. 1))
     $     WRITE (14,924) ISTATE,
     $     KFERM(ISTATE),KXFERM(ISTATE),KYFERM(ISTATE),ISFERM(ISTATE),
     $     KAFER(ISTATE),KXAFER(ISTATE),KYAFER(ISTATE),ISAFER(ISTATE),
     $     REALWF(ISTATE),AIMWF(ISTATE)
  924      FORMAT (I6,15X,2(1X,I3,1X,I3,1X,I3,1X,I2),2(1X,F7.4))
```

```
            IF ((NPHOT(ISTATE) .EQ. 0) .AND. (NAFER(ISTATE) .EQ. 0))
     $       WRITE (14,925) ISTATE,
     $       KFERM(ISTATE),KXFERM(ISTATE),KYFERM(ISTATE),ISFERM(ISTATE),
     $       REALWF(ISTATE),AIMWF(ISTATE)
 925         FORMAT (I6,15X,1X,I3,1X,I3,1X,I3,1X,I2,15X,2(1X,F7.4))
 20        CONTINUE
          ENDIF
C************

          CALL VTTIME(IVIRTIME,ITOTTIME)
          TIME2 = DFLOAT(ITOTTIME)/100.0D0-START-TIME1

C-------------------------------------------------------
C     CALL SUBROUTINES HONR, H1NR, H2NR, HSELFNR,
C     TO FIND HAMILTONIAN MATRIX ELEMENT.
C-------------------------------------------------------
          IF (IHO .EQ. 1)
     $     CALL HONR(REMSQHO,NSIZE,NSTATES,REALWF,AIMWF,
     $               ALPERP,PHOTMASS,FERMMASS,
     $               NPHOT,NFERM,NAFER,
     $               KPHOT,KXPHOT,KYPHOT,
     $               KFERM,KXFERM,KYFERM,
     $               KAFER,KXAFER,KYAFER)

          CALL VTTIME(IVIRTIME,ITOTTIME)
          TIME3 = DFLOAT(ITOTTIME)/100.0D0-START-TIME1-TIME2

          IF (IH1 .EQ. 1)
     $     CALL H1NR(REMSQH1L,REMSQH1T,
     $               NSIZE,NSTATES,NSTATEOP,REALWF,AIMWF,
     $               ALPHAG,ICHARGE,ALPERP,FERMMASS,NOAFERI,
     $               NPHOT,NFERM,NAFER,
     $               KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $               KFERM,KXFERM,KYFERM,ISFERM,
     $               KAFER,KXAFER,KYAFER,ISAFER,
     $               WORK1,WORK2,WORK3)

          CALL VTTIME(IVIRTIME,ITOTTIME)
          TIME4 = DFLOAT(ITOTTIME)/100.0D0-START-TIME1-TIME2-TIME3

          IF ((IH2PHOT .EQ. 1) .OR. (IH2FERM .EQ. 1))
     $     CALL H2NR(REMSQH2P,REMSQH2F,
     $               NSIZE,NSTATES,NSTATEOP,REALWF,AIMWF,
     $               ALPHAG,KPLUSTOT,ICHARGE,ALAMBDA,ALPERP,EPSILON,
     $               RPHOMASS,RFERMASS,IUVFERM,NOAFERI,IH2PHOT,IH2FERM,
     $               NPHOT,NFERM,NAFER,
     $               KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $               KFERM,KXFERM,KYFERM,ISFERM,
     $               KAFER,KXAFER,KYAFER,ISAFER,
     $               WORK1,WORK2)

          CALL VTTIME(IVIRTIME,ITOTTIME)
          TIME5 = DFLOAT(ITOTTIME)/100.0D0-START-TIME1-TIME2-TIME3-TIME4

          IF (IHSELF .EQ. 1)
```

```
     $     CALL HSELFNR(REMSQHSL,REMSQHST,REMSQHS2,REMSQHS3,
     $                  NSIZE,NSTATEOP,REALWF,AIMWF,
     $                  ALPHAG,KPLUSTOT,IBC,ICHARGE,ALAMBDA,ALPERP,EPSILON,
     $                  PHOTMASS,FERMMASS,RPHOMASS,RFERMASS,NOAFERI,
     $                  IFERMUP,IUVFERM,NPHOT,NFERM,NAFER,
     $                  KPHOT,KXPHOT,KYPHOT,
     $                  KFERM,KXFERM,KYFERM,
     $                  KAFER,KXAFER,KYAFER)

           CALL VTTIME(IVIRTIME,ITOTTIME)
           TIME6 = DFLOAT(ITOTTIME)/100.0D0-START-TIME1-TIME2-TIME3-TIME4
     $                                                          -TIME5

           REMSQHO = DFLOAT(KPLUSTOT)*REMSQHO
           REMSQH1L = DFLOAT(KPLUSTOT)*REMSQH1L
           REMSQH1T = DFLOAT(KPLUSTOT)*REMSQH1T
           REMSQH2P = DFLOAT(KPLUSTOT)*REMSQH2P
           REMSQH2F = DFLOAT(KPLUSTOT)*REMSQH2F
           REMSQHSL = DFLOAT(KPLUSTOT)*REMSQHSL
           REMSQHST = DFLOAT(KPLUSTOT)*REMSQHST
           REMSQHS2 = DFLOAT(KPLUSTOT)*REMSQHS2
           REMSQHS3 = DFLOAT(KPLUSTOT)*REMSQHS3

C-------------------------------------------------------
C     CALL SUBROUTINE PRINTOUT TO PRINT OUT RESULTS.
C-------------------------------------------------------
           IF (IPRIOUT .EQ. 1)
     $     CALL PRINTOUT(WAVEFCN,TIME1,TIME2,TIME3,TIME4,TIME5,TIME6,
     $                   KPLUSTOT,ICHARGE,IBC,NSTATES,NSTATEOP,NSIZE,
     $                   KPLUSMAX,ALPHAG,ALAMBDA,ALPERP,EPSILON,
     $                   PARA1,PARA2,PARA3,PARA4,PARA5,
     $                   PHOTMASS,RPHOMASS,FERMMASS,RFERMASS,
     $                   REMSQHO,REMSQH1L,REMSQH1T,REMSQH2P,REMSQH2F,
     $                   REMSQHSL,REMSQHST,REMSQHS2,REMSQHS3,
     $                   REALWF,AIMWF,NPHOT,NFERM,NAFER,
     $                   KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $                   KFERM,KXFERM,KYFERM,ISFERM,
     $                   KAFER,KXAFER,KYAFER,ISAFER)

           WRITE (21,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $     .5*REMSQH1L+REMSQHSL
           WRITE (22,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $     .5*REMSQH1T+REMSQHST
           WRITE (23,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $     REMSQH2P
           WRITE (24,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $     .5*REMSQH1L+.5*REMSQH1T+REMSQHSL+REMSQHST+REMSQH2P
           IF (ICHARGE .EQ. 0) THEN
             WRITE (25,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $       REMSQHO+.5*REMSQH1L+.5*REMSQH1T-4.0D0
             WRITE (26,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $       REMSQHO+REMSQH1L+REMSQH1T+REMSQH2P+REMSQHSL+REMSQHST-4.0D0
            ELSEIF (ICHARGE .EQ. -1) THEN
             WRITE (25,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $       REMSQHO+.5*REMSQH1L+.5*REMSQH1T-1.0D0
```

```
          WRITE (26,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $       REMSQHO+REMSQH1L+REMSQH1T+REMSQH2P+REMSQHSL+REMSQHST-1.ODO
        ENDIF
  930   FORMAT (3X,I4,2X,3F11.4,F15.10)

  100 CONTINUE

      STOP
      END


      SUBROUTINE STATESNR(NSIZE,KPLUSTOT,IBC,ICHARGE,
     $                    ALAMBDA,ALPERP,EPSILON,
     $                    RPHOMASS,RFERMASS,IFERMUP,IUVFERM,IONEPHOT,
     $                    NSTATES,NSTATEOP,TOOMANY,
     $                    NPHOT,NFERM,NAFER,
     $                    KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $                    KFERM,KXFERM,KYFERM,ISFERM,
     $                    KAFER,KXAFER,KYAFER,ISAFER)
C-------------------------------------------------------------
C     THIS SUBROUTINE GENERATES THE FOCK STATES CONSISTENT WITH
C     KPLUSTOT, ALAMBDA, ALPERP.
C     FOR Q=-1 ONLY KEEP STATES WITH 1 FERMION AND 0,1 PHOTONS.
C     FOR Q=0 ONLY KEEP STATES WITH 1 FERMION PAIR AND 0,1 PHOTONS.
C
C
C     OUTPUT VARIABLES:
C     NSTATES   NUMBER OF FOCK STATES.
C
C     NSTATEOP  NUMBER OF FOCK STATES WITH NO PHOTONS.
C
C     TOOMANY   LOGICAL VARIABLE.  TOOMANY=TRUE  IF NSTATES .GT. NSIZE.
C                                  TOOMANY=FALSE IF NSTATES .LE. NSIZE.
C
C     NPHOT,    # PHOTONS, FERMIONS, ANTI-FERMIONS IN FOCK STATES.
C     NFERM,    ARRAYS OF DIMENSION NSIZE.
C     NAFER
C
C     KPHOT,    KPLUS, KX, KY, SPIN OF PHOTON IN FOCK STATES.
C     KXPHOT,   ARRAYS OF DIMENSION NSIZE.
C     KYPHOT,
C     ISPHOT
C
C     KFERM,    KPLUS, KX, KY, SPIN OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM,
C     ISFERM
C
C     KAFER,    KPLUS, KX, KY, SPIN OF ANTI-FERMION IN FOCK STATES.
C     KXAFER,   ARRAYS OF DIMENSION NSIZE.
C     KYAFER,
C     ISAFER
C
C
C     INPUT VARIABLES:
```

```
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C               TO NSTATES.
C
C     KPLUSTOT  TOTAL KPLUS OF INCOMING, OUTGOING STATES.
C
C     ICHARGE   TOTAL CHARGE.
C
C     IBC       ONLY HAVE EVEN FERMION KPLUS IF IBC=2,
C               ODD FERMION KPLUS IF IBC=1.
C
C     ALAMBDA   VALUE OF CUT-OFF MASS.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     EPSILON   MINIMUM PHOTON INVMASS**2.
C
C     RPHOMASS  PHOTON MASS TO BE USED IN COVARIANT CUT-OFF.
C
C     RFERMASS  FERMION MASS TO BE USED IN COVARIANT CUT-OFF.
C
C     IFERMUP   FERMIONS, ANTI-FERMIONS ALL HAVE SPIN UP.
C
C     IUVFERM   UV CUT-OFF ONLY APPLIED TO FERMIONS, ANTI-FERMIONS.
C
C     IONEPHOT  PUT IN STATES WITH ONE PHOTON.
C
C
C     USAGE NOTES:
C     1) FOCK STATES 1 TO NSTATEOP HAVE NO PHOTONS,
C        STATES NSTATEOP+1 TO NSTATES HAVE ONE PHOTON.
C     2) THIS ROUTINE ASSUMES THAT THE TOTAL KPERP OF THE INCOMING
C        AND OUTGOING FOCK STATES IS ZERO.  THIS IS NECESSARY IN THE
C        CALCULATION OF INVARIANT MASSES.
C     3) FOCK STATES ARE GENERATED WITH KPERP=0.
C     4) FERMION CHARGE IS ASSUMED TO BE -1.
C     5) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     6) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)
      DIMENSION
     $  NPHOT(NSIZE),NFERM(NSIZE),NAFER(NSIZE),
     $  KPHOT(NSIZE),KFERM(NSIZE),KAFER(NSIZE),
     $  KXPHOT(NSIZE),KXFERM(NSIZE),KXAFER(NSIZE),
     $  KYPHOT(NSIZE),KYFERM(NSIZE),KYAFER(NSIZE),
     $  ISPHOT(NSIZE),ISFERM(NSIZE),ISAFER(NSIZE)

      LOGICAL TOOMANY

      PI = 3.1415926535897930
      SMALL = 1.0D-13

C-------------------------------------------------------
```

```
C     INITIALIZE VARIOUS ARRAYS TO ZERO.
C-------------------------------------------------------
      DO 10 I = 1,NSIZE
        NPHOT(I) = 0
        NFERM(I) = 0
        NAFER(I) = 0
        KPHOT(I) = 0
        KXPHOT(I) = 0
        KYPHOT(I) = 0
        ISPHOT(I) = 0
        KFERM(I) = 0
        KXFERM(I) = 0
        KYFERM(I) = 0
        ISFERM(I) = 0
        KAFER(I) = 0
        KXAFER(I) = 0
        KYAFER(I) = 0
        ISAFER(I) = 0
 10   CONTINUE

      TOOMANY = .FALSE.

C-------------------------------------------------------
C     GENERATE STATES WITH 1 FERMION PAIR, 0 PHOTONS (ICHARGE=0).
C     RECALL THAT KPLUSTOT MUST BE EVEN FOR ICHARGE = 0.
C-------------------------------------------------------
      IF (ICHARGE .EQ. 0) THEN
        ISTATE = 0

C************DIAGNOSTICS
C       WRITE (16,920) ALAMBDA**2
C920    FORMAT (' '/' 1 PAIR STATES  LAMBDA**2 =',F8.3/
C    $  '  KPLUSF KPLUSA KXF  KYF  KXA  KYA  INVMASS'/
C    $  '  ----------------------------------------')
C************

C       FIGURE OUT WHAT VALUES OF FERMION KPLUS TO RUN OVER.
        IF (IBC .EQ. 1) THEN
          IFIRSTKF = 1
          ILASTKF = KPLUSTOT - 1
         ELSEIF (IBC .EQ. 2) THEN
          IFIRSTKF = 2
          ILASTKF = KPLUSTOT - 2
        ENDIF

        DO 20 KPLUSF=IFIRSTKF,ILASTKF,2
          KPLUSA = KPLUSTOT-KPLUSF
          XF = DFLOAT(KPLUSF)/DFLOAT(KPLUSTOT)
          XA = DFLOAT(KPLUSA)/DFLOAT(KPLUSTOT)

C         ONLY CONTINUE IF SUM(MASS**2/X) .LE. LAMBDA**2.
          AINVMASS = RFERMASS**2/XF + RFERMASS**2/XA

C************DIAGNOSTICS
C         WRITE (16,921) KPLUSF,KPLUSA,AINVMASS
```

```
C921      FORMAT (3X,I4,3X,I4,21X,F8.3)
C*************

          IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 20

C         FIGURE OUT WHAT LARGEST ALLOWED FERMION KX, KY IS.
          ARGUMENT = ALAMBDA**2*XF*XA - RFERMASS**2
          IF (ARGUMENT .LT. SMALL) THEN
            KPFMAX = 0
           ELSE
            KPFMAX = INT(ALPERP/PI*DSQRT(ARGUMENT) + SMALL)
          ENDIF

          DO 22 KXF=-KPFMAX,KPFMAX
          DO 22 KYF=-KPFMAX,KPFMAX
            KXA = -KXF
            KYA = -KYF
            AKPFSQ = (PI/ALPERP)**2*(DFLOAT(KXF**2) + DFLOAT(KYF**2))

C           KEEP STATE IF INVMASS**2 .LE. LAMBDA**2.
            AINVMASS = (AKPFSQ + RFERMASS**2)/XF
     $               + (AKPFSQ + RFERMASS**2)/XA

C*************DIAGNOSTICS
C           WRITE (16,922) KXF,KYF,KXA,KYA,AINVMASS
C922        FORMAT (15X,I4,1X,I4,1X,I4,1X,I4,1X,F8.3)
C*************

            IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 22

            IF (IFERMUP .EQ. 0) IFIRSTS=-1
            IF (IFERMUP .EQ. 1) IFIRSTS=+1
            DO 24 ISPINF=IFIRSTS,1,2
            DO 24 ISPINA=IFIRSTS,1,2
              ISTATE = ISTATE + 1
              IF (ISTATE .GT. NSIZE) THEN
                TOOMANY = .TRUE.
                RETURN
              ENDIF
              NPHOT(ISTATE) = 0
              NFERM(ISTATE) = 1
              NAFER(ISTATE) = 1
              KFERM(ISTATE) = KPLUSF
              KXFERM(ISTATE) = KXF
              KYFERM(ISTATE) = KYF
              ISFERM(ISTATE) = ISPINF
              KAFER(ISTATE) = KPLUSA
              KXAFER(ISTATE) = KXA
              KYAFER(ISTATE) = KYA
              ISAFER(ISTATE) = ISPINA
24          CONTINUE
22        CONTINUE
20      CONTINUE

        NSTATEOP = ISTATE
```

```
C-------------------------------------------------------
C     GENERATE STATES WITH 1 FERMION PAIR, 1 PHOTON (ICHARGE=0).
C     RECALL THAT KPLUSTOT MUST BE EVEN FOR ICHARGE = 0.
C-------------------------------------------------------
        IF ((IONEPHOT .EQ. 1) .AND. (IUVFERM .EQ. 0)) THEN

C************DIAGNOSTICS
C         WRITE (16,930) ALAMBDA**2
C930      FORMAT (' '/' 1 PAIR,1 PHOTON STATES  LAMBDA**2 =',F8.3/
C     $    '  KPLUSP KPLUSF KPLUSA KXP  KYP  KXF  KYF  KXA  KYA  INVMASS'/
C     $    '  ------------------------------------------------------------')
C************

C         FIGURE OUT WHAT VALUES OF PHOTON KPLUS TO RUN OVER.
          IFIRSTKP = 2
          IF (IBC .EQ. 1) THEN
            ILASTKP = KPLUSTOT - 2
           ELSEIF (IBC .EQ. 2) THEN
            ILASTKP = KPLUSTOT - 4
          ENDIF

          DO 30 KPLUSP=IFIRSTKP,ILASTKP,2
            XP = DFLOAT(KPLUSP)/DFLOAT(KPLUSTOT)

C           FIGURE OUT WHAT VALUES OF FERMION KPLUS TO RUN OVER.
            IF (IBC .EQ. 1) THEN
              IFIRSTKF = 1
              ILASTKF = KPLUSTOT - KPLUSP - 1
             ELSEIF (IBC .EQ. 2) THEN
              IFIRSTKF = 2
              ILASTKF = KPLUSTOT - KPLUSP - 2
            ENDIF

            DO 31 KPLUSF=IFIRSTKF,ILASTKF,2
              KPLUSA = KPLUSTOT-KPLUSP-KPLUSF
              XF = DFLOAT(KPLUSF)/DFLOAT(KPLUSTOT)
              XA = DFLOAT(KPLUSA)/DFLOAT(KPLUSTOT)

C             ONLY CONTINUE IF SUM(MASS**2/X) .LE. LAMBDA**2.
              AINVMASS = RPHOMASS**2/XP + RFERMASS**2/XF + RFERMASS**2/XA

C************DIAGNOSTICS
C             WRITE (16,931) KPLUSP,KPLUSF,KPLUSA,AINVMASS
C931          FORMAT (3X,I4,3X,I4,3X,I4,31X,F8.3)
C************

              IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 31

C             FIGURE OUT WHAT LARGEST ALLOWED PHOTON KX, KY IS.
              ARGUMENT = XP*(ALAMBDA**2 - RFERMASS**2/XF
     $                                  - RFERMASS**2/XA)
     $                  - RPHOMASS**2
              IF (ARGUMENT .LT. SMALL) THEN
                KPPMAX = 0
```

```
                ELSE
                  KPPMAX = INT(ALPERP/PI*DSQRT(ARGUMENT) + SMALL)
                ENDIF

                DO 32 KXP=-KPPMAX,KPPMAX
                DO 32 KYP=-KPPMAX,KPPMAX
                  AKPPSQ = (PI/ALPERP)**2*(DFLOAT(KXP**2)
     $                                   + DFLOAT(KYP**2))

C               REMOVE PHOTONS WITH INVMASS**2 .LT. EPSILON.
                IF ((AKPPSQ + RPHOMASS**2)/XP .LT. EPSILON-SMALL) GOTO 32

C               FIGURE OUT WHAT LARGEST ALLOWED FERMION KX, KY IS.
                ARGUMENT = XF*(ALAMBDA**2 - (AKPPSQ+RPHOMASS**2)/XP
     $                                    - RFERMASS**2/XA)
     $                  - RFERMASS**2
                IF (ARGUMENT .LT. SMALL) THEN
                  KPFMAX = 0
                 ELSE
                  KPFMAX = INT(ALPERP/PI*DSQRT(ARGUMENT) + SMALL)
                ENDIF
                DO 33 KXF=-KPFMAX,KPFMAX
                DO 33 KYF=-KPFMAX,KPFMAX
                  KXA = -KXP-KXF
                  KYA = -KYP-KYF
                  AKPFSQ = (PI/ALPERP)**2*(DFLOAT(KXF**2)
     $                                   + DFLOAT(KYF**2))
                  AKPASQ = (PI/ALPERP)**2*(DFLOAT(KXA**2)
     $                                   + DFLOAT(KYA**2))

C               KEEP STATE IF INVMASS**2 .LE. LAMBDA**2.
                AINVMASS = (AKPPSQ + RPHOMASS**2)/XP
     $                   + (AKPFSQ + RFERMASS**2)/XF
     $                   + (AKPASQ + RFERMASS**2)/XA

C************DIAGNOSTICS
C               WRITE (16,932) KXP,KYP,KXF,KYF,KXA,KYA,AINVMASS
C932            FORMAT (22X,I4,1X,I4,1X,I4,1X,I4,1X,I4,1X,I4,1X,I4,1X,F8.3)
C************

                IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 33

                IF (IFERMUP .EQ. 0) IFIRSTS=-1
                IF (IFERMUP .EQ. 1) IFIRSTS=+1
                DO 34 ISPINP=-1,1,2
                DO 34 ISPINF=IFIRSTS,1,2
                DO 34 ISPINA=IFIRSTS,1,2
                  ISTATE = ISTATE + 1
                  IF (ISTATE .GT. NSIZE) THEN
                    TOOMANY = .TRUE.
                    RETURN
                  ENDIF
                  NPHOT(ISTATE) = 1
                  NFERM(ISTATE) = 1
                  NAFER(ISTATE) = 1
```

```
                KPHOT(ISTATE) = KPLUSP
                KXPHOT(ISTATE) = KXP
                KYPHOT(ISTATE) = KYP
                ISPHOT(ISTATE) = ISPINP
                KFERM(ISTATE) = KPLUSF
                KXFERM(ISTATE) = KXF
                KYFERM(ISTATE) = KYF
                ISFERM(ISTATE) = ISPINF
                KAFER(ISTATE) = KPLUSA
                KXAFER(ISTATE) = KXA
                KYAFER(ISTATE) = KYA
                ISAFER(ISTATE) = ISPINA
 34           CONTINUE
 33         CONTINUE
 32       CONTINUE
 31     CONTINUE
 30    CONTINUE
       ENDIF

       IF ((IONEPHOT .EQ. 1) .AND. (IUVFERM .EQ. 1)) THEN

C************DIAGNOSTICS
C        WRITE (16,940) ALAMBDA**2
C940     FORMAT (' '/' 1 PAIR,1 PHOTON STATES  LAMBDA**2 =',F8.3/
C    $    '  KPLUSF KPLUSA KXF  KYF  KXA  KYA  INVMASS'/
C    $    '  ----------------------------------------')
C************

C        FIGURE OUT WHAT VALUES OF FERMION KPLUS TO RUN OVER.
         IF (IBC .EQ. 1) THEN
           IFIRSTK = 1
           ILASTK = KPLUSTOT - 1
          ELSEIF (IBC .EQ. 2) THEN
           IFIRSTK = 2
           ILASTK = KPLUSTOT - 2
         ENDIF

         DO 40 KPLUSF=IFIRSTK,ILASTK,2
         DO 40 KPLUSA=IFIRSTK,KPLUSTOT-KPLUSF-2,2
           KPLUSP = KPLUSTOT-KPLUSF-KPLUSA
           XF = DFLOAT(KPLUSF)/DFLOAT(KPLUSTOT)
           XA = DFLOAT(KPLUSA)/DFLOAT(KPLUSTOT)
           XP = DFLOAT(KPLUSP)/DFLOAT(KPLUSTOT)

C          ONLY CONTINUE IF SUM(MASS**2/X) .LE. LAMBDA**2.
           AINVMASS = RFERMASS**2/XF + RFERMASS**2/XA

C************DIAGNOSTICS
C          WRITE (16,941) KPLUSF,KPLUSA,AINVMASS
C941       FORMAT (3X,I4,3X,I4,21X,F8.3)
C************

           IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 40

C          FIGURE OUT WHAT LARGEST ALLOWED FERMION KX, KY IS.
```

```
            ARGUMENT = XF*(ALAMBDA**2 - RFERMASS**2*(1.0/XF+1.0/XA))
            IF (ARGUMENT .LT. SMALL) THEN
              KPFMAX = 0
             ELSE
              KPFMAX = INT(ALPERP/PI*DSQRT(ARGUMENT) + SMALL)
            ENDIF

C           FIGURE OUT WHAT LARGEST ALLOWED AFERMION KX, KY IS.
            ARGUMENT = XA*(ALAMBDA**2 - RFERMASS**2*(1.0/XF+1.0/XA))
            IF (ARGUMENT .LT. SMALL) THEN
              KPAMAX = 0
             ELSE
              KPAMAX = INT(ALPERP/PI*DSQRT(ARGUMENT) + SMALL)
            ENDIF

            DO 42 KXF=-KPFMAX,KPFMAX
            DO 42 KYF=-KPFMAX,KPFMAX
            DO 42 KXA=-KPAMAX,KPAMAX
            DO 42 KYA=-KPAMAX,KPAMAX
              KXP = -KXF-KXA
              KYP = -KYF-KYA
              AKPFSQ = (PI/ALPERP)**2*(DFLOAT(KXF**2) + DFLOAT(KYF**2))
              AKPASQ = (PI/ALPERP)**2*(DFLOAT(KXA**2) + DFLOAT(KYA**2))
              AKPPSQ = (PI/ALPERP)**2*(DFLOAT(KXP**2) + DFLOAT(KYP**2))

C             REMOVE PHOTONS WITH INVMASS**2 .LT. EPSILON.
              IF ((AKPPSQ + RPHOMASS**2)/XP .LT. EPSILON-SMALL) GOTO 42

C             KEEP STATE IF INVMASS**2 .LE. LAMBDA**2.
              AINVMASS = (AKPFSQ + RFERMASS**2)/XF
     $                   + (AKPASQ + RFERMASS**2)/XA

C************DIAGNOSTICS
C             WRITE (16,942) KXF,KYF,KXA,KYA,AINVMASS
C942          FORMAT (15X,I4,1X,I4,1X,I4,1X,I4,1X,F8.3)
C*************

              IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 42

              IF (IFERMUP .EQ. 0) IFIRSTS=-1
              IF (IFERMUP .EQ. 1) IFIRSTS=+1
              DO 44 ISPINF=IFIRSTS,1,2
              DO 44 ISPINA=IFIRSTS,1,2
              DO 44 ISPINP=-1,1,2
                ISTATE = ISTATE + 1
                IF (ISTATE .GT. NSIZE) THEN
                  TOOMANY = .TRUE.
                  RETURN
                ENDIF
                NPHOT(ISTATE) = 1
                NFERM(ISTATE) = 1
                NAFER(ISTATE) = 1
                KFERM(ISTATE) = KPLUSF
                KXFERM(ISTATE) = KXF
                KYFERM(ISTATE) = KYF
```

```
                ISFERM(ISTATE) = ISPINF
                KAFER(ISTATE) = KPLUSA
                KXAFER(ISTATE) = KXA
                KYAFER(ISTATE) = KYA
                ISAFER(ISTATE) = ISPINA
                KPHOT(ISTATE) = KPLUSP
                KXPHOT(ISTATE) = KXP
                KYPHOT(ISTATE) = KYP
                ISPHOT(ISTATE) = ISPINP
 44           CONTINUE
 42         CONTINUE
 40       CONTINUE
          ENDIF

          NSTATES = ISTATE

C       END OF GENERATING STATES FOR ICHARGE = 0.
        ENDIF

C-------------------------------------------------------
C       GENERATE STATES WITH 1 FERMION, 0 PHOTONS (ICHARGE=-1).
C-------------------------------------------------------
        IF (ICHARGE .EQ. -1) THEN
          IF (IFERMUP .EQ. 0) THEN
            ISTATE = 2
            NPHOT(1) = 0
            NFERM(1) = 1
            NAFER(1) = 0
            KFERM(1) = KPLUSTOT
            KXFERM(1) = 0
            KYFERM(1) = 0
            ISFERM(1) = -1
            NPHOT(2) = 0
            NFERM(2) = 1
            NAFER(2) = 0
            KFERM(2) = KPLUSTOT
            KXFERM(2) = 0
            KYFERM(2) = 0
            ISFERM(2) = +1
          ENDIF
          IF (IFERMUP .EQ. 1) THEN
            ISTATE = 1
            NPHOT(1) = 0
            NFERM(1) = 1
            NAFER(1) = 0
            KFERM(1) = KPLUSTOT
            KXFERM(1) = 0
            KYFERM(1) = 0
            ISFERM(1) = 1
          ENDIF

          NSTATEOP = ISTATE

C-------------------------------------------------------
C       GENERATE STATES WITH 1 FERMION, 1 PHOTON (ICHARGE=-1).
```

```
C-------------------------------------------------------
        IF ((IONEPHOT .EQ. 1) .AND. (IUVFERM .EQ. 0)) THEN

C************DIAGNOSTICS
C         WRITE (16,950) ALAMBDA**2
C950      FORMAT (' '/' 1 FERMION,1 PHOTON STATES  LAMBDA**2 =',F8.3/
C     $    '  KPLUSP KPLUSF KXP  KYP  KXF  KYF  INVMASS'/
C     $    '  ----------------------------------------')
C************

C         FIGURE OUT WHAT VALUES OF PHOTON KPLUS TO RUN OVER.
          IFIRSTKP = 2
          IF (IBC .EQ. 1) THEN
            ILASTKP = KPLUSTOT - 1
           ELSEIF (IBC .EQ. 2) THEN
            ILASTKP = KPLUSTOT - 2
          ENDIF

          DO 50 KPLUSP=IFIRSTKP,ILASTKP,2
            KPLUSF = KPLUSTOT-KPLUSP
            XP = DFLOAT(KPLUSP)/DFLOAT(KPLUSTOT)
            XF = DFLOAT(KPLUSF)/DFLOAT(KPLUSTOT)

C           ONLY CONTINUE IF SUM(MASS**2/X) .LE. LAMBDA**2.
            AINVMASS = RPHOMASS**2/XP + RFERMASS**2/XF

C************DIAGNOSTICS
C           WRITE (16,951) KPLUSP,KPLUSF,AINVMASS
C951        FORMAT (3X,I4,3X,I4,21X,F8.3)
C************

            IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 50

C           FIGURE OUT WHAT LARGEST ALLOWED PHOTON KX, KY IS.
            ARGUMENT = XP*XF*(ALAMBDA**2 - RPHOMASS**2/XP
     $                                   - RFERMASS**2/XF)
            IF (ARGUMENT .LT. SMALL) THEN
              KPPMAX = 0
             ELSE
              KPPMAX = INT(ALPERP/PI*DSQRT(ARGUMENT) + SMALL)
            ENDIF

            DO 52 KXP=-KPPMAX,KPPMAX
            DO 52 KYP=-KPPMAX,KPPMAX
              KXF = -KXP
              KYF = -KYP
              AKPPSQ = (PI/ALPERP)**2*(DFLOAT(KXP**2) + DFLOAT(KYP**2))

C             REMOVE PHOTONS WITH INVMASS**2 .LT. EPSILON.
              IF ((AKPPSQ + RPHOMASS**2)/XP .LT. EPSILON-SMALL) GOTO 52

C             KEEP STATE IF INVMASS**2 .LE. LAMBDA**2.
              AINVMASS = (AKPPSQ + RPHOMASS**2)/XP
     $                 + (AKPPSQ + RFERMASS**2)/XF
```

```
C************DIAGNOSTICS
C             WRITE (16,952) KXP,KYP,KXF,KYF,AINVMASS
C952          FORMAT (15X,I4,1X,I4,1X,I4,1X,I4,1X,F8.3)
C*************

              IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 52

              IF (IFERMUP .EQ. 0) IFIRSTS=-1
              IF (IFERMUP .EQ. 1) IFIRSTS=+1
              DO 54 ISPINP=-1,1,2
              DO 54 ISPINF=IFIRSTS,1,2
                ISTATE = ISTATE + 1
                IF (ISTATE .GT. NSIZE) THEN
                  TOOMANY = .TRUE.
                  RETURN
                ENDIF
                NPHOT(ISTATE) = 1
                NFERM(ISTATE) = 1
                NAFER(ISTATE) = 0
                KPHOT(ISTATE) = KPLUSP
                KXPHOT(ISTATE) = KXP
                KYPHOT(ISTATE) = KYP
                ISPHOT(ISTATE) = ISPINP
                KFERM(ISTATE) = KPLUSF
                KXFERM(ISTATE) = KXF
                KYFERM(ISTATE) = KYF
                ISFERM(ISTATE) = ISPINF
 54           CONTINUE
 52         CONTINUE
 50       CONTINUE
         ENDIF

         IF ((IONEPHOT .EQ. 1) .AND. (IUVFERM .EQ. 1)) THEN

C************DIAGNOSTICS
C         WRITE (16,960) ALAMBDA**2
C960      FORMAT (' '/' 1 FERMION,1 PHOTON STATES  LAMBDA**2 =',F8.3/
C     $     '  KPLUSP KPLUSF KXP  KYP  KXF  KYF  INVMASS'/
C     $     '  ----------------------------------------')
C*************

C         FIGURE OUT WHAT VALUES OF PHOTON KPLUS TO RUN OVER.
          IFIRSTKP = 2
          IF (IBC .EQ. 1) THEN
            ILASTKP = KPLUSTOT - 1
          ELSEIF (IBC .EQ. 2) THEN
            ILASTKP = KPLUSTOT - 2
          ENDIF

          DO 60 KPLUSP=IFIRSTKP,ILASTKP,2
            KPLUSF = KPLUSTOT-KPLUSP
            XP = DFLOAT(KPLUSP)/DFLOAT(KPLUSTOT)
            XF = DFLOAT(KPLUSF)/DFLOAT(KPLUSTOT)

C           ONLY CONTINUE IF SUM(MASS**2/X) .LE. LAMBDA**2.
```

```
            AINVMASS = RFERMASS**2/XF

C************DIAGNOSTICS
C           WRITE (16,961) KPLUSP,KPLUSF,AINVMASS
C961        FORMAT (3X,I4,3X,I4,21X,F8.3)
C************

            IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 60

C           FIGURE OUT WHAT LARGEST ALLOWED FERMION KX, KY IS.
            ARGUMENT = XF*ALAMBDA**2 - RFERMASS**2
            IF (ARGUMENT .LT. SMALL) THEN
              KPFMAX = 0
             ELSE
              KPFMAX = INT(ALPERP/PI*DSQRT(ARGUMENT) + SMALL)
            ENDIF

            DO 62 KXF=-KPFMAX,KPFMAX
            DO 62 KYF=-KPFMAX,KPFMAX
              KXP = -KXF
              KYP = -KYF
              AKPPSQ = (PI/ALPERP)**2*(DFLOAT(KXP**2) + DFLOAT(KYP**2))

C             REMOVE PHOTONS WITH INVMASS**2 .LT. EPSILON.
              IF ((AKPPSQ + RPHOMASS**2)/XP .LT. EPSILON-SMALL) GOTO 62

C             KEEP STATE IF INVMASS**2 .LE. LAMBDA**2.
              AINVMASS = (AKPPSQ + RFERMASS**2)/XF

C************DIAGNOSTICS
C             WRITE (16,952) KXP,KYP,KXF,KYF,AINVMASS
C952          FORMAT (15X,I4,1X,I4,1X,I4,1X,I4,1X,F8.3)
C************

              IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 62

              IF (IFERMUP .EQ. 0) IFIRSTS=-1
              IF (IFERMUP .EQ. 1) IFIRSTS=+1
              DO 64 ISPINP=-1,1,2
              DO 64 ISPINF=IFIRSTS,1,2
                ISTATE = ISTATE + 1
                IF (ISTATE .GT. NSIZE) THEN
                  TOOMANY = .TRUE.
                  RETURN
                ENDIF
                NPHOT(ISTATE) = 1
                NFERM(ISTATE) = 1
                NAFER(ISTATE) = 0
                KPHOT(ISTATE) = KPLUSP
                KXPHOT(ISTATE) = KXP
                KYPHOT(ISTATE) = KYP
                ISPHOT(ISTATE) = ISPINP
                KFERM(ISTATE) = KPLUSF
                KXFERM(ISTATE) = KXF
                KYFERM(ISTATE) = KYF
```

```
                ISFERM(ISTATE) = ISPINF
 64           CONTINUE
 62         CONTINUE
 60       CONTINUE
         ENDIF

         NSTATES = ISTATE

C      END OF GENERATING STATES FOR ICHARGE = -1.
       ENDIF

       RETURN
       END


       SUBROUTINE VARWF(REALWF,AIMWF,NSIZE,NSIZEO,NSTATES,NSTATEOP,
      $                 ALPHAG,KPLUSTOT,ICHARGE,ALPERP,
      $                 PHOTMASS,FERMMASS,IFERMUP,NOAFERI,
      $                 PARA1,PARA2,PARA3,PARA4,PARA5,
      $                 NPHOT,NFERM,NAFER,
      $                 KPHOT,KXPHOT,KYPHOT,ISPHOT,
      $                 KFERM,KXFERM,KYFERM,ISFERM,
      $                 KAFER,KXAFER,KYAFER,ISAFER,
      $                 REALH1L,REALH1T,AIMH1T,TEMPWF)
C-------------------------------------------------------
C      THIS ROUTINE RETURNS THE VARIATIONAL WAVE FUNCTION
C      FOR THE INPUTTED FOCK STATES.
C
C
C      OUTPUT VARIABLES:
C      REALWF     REAL PART OF NORMALIZED FOCK STATE WAVE FUNCTIONS.
C
C      AIMWF      IMAGINARY PART OF NORMALIZED FOCK STATE WAVE FUNCTIONS.
C
C
C      INPUT VARIABLES:
C      NSIZE      ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C                 CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C                 TO NSTATES.
C
C      NSIZEO     ARRAY DIMENSION OF TEMPWF AS DEFINED IN CALLING ROUTINE
C                 IT SHOULD BE GREATER THAN OR EQUAL TO NSTATEOP.
C
C      NSTATES    NUMBER OF FOCK STATES.
C
C      NSTATEOP   NUMBER OF FOCK STATES WITH NO PHOTONS.
C
C      ALPHAG     VALUE OF COUPLING CONSTANT (=G**2/4PI).
C
C      KPLUSTOT   TOTAL KPLUS OF INCOMING, OUTGOING STATES.
C
C      ICHARGE    TOTAL CHARGE.
C
C      ALPERP     SIZE OF KPERP GRID.
C
```

```
C     PHOTMASS  PHOTON MASS IN LAGRANGIAN.
C
C     FERMMASS  FERMION MASS IN LAGRANGIAN.
C
C     IFERMUP   FERMIONS, ANTI-FERMIONS ALL HAVE SPIN UP.
C
C     NOAFERI   NO INTERACTIONS WITH ANTI-FERMION.
C
C     PARA1,
C     PARA2,    VARIATIONAL PARAMETERS (PARA4, PARA5 NOT USED).
C     PARA3,
C     PARA4,
C     PARA5
C
C     NPHOT,    # PHOTONS, FERMIONS, ANTI-FERMIONS IN FOCK STATES.
C     NFERM,    ARRAYS OF DIMENSION NSIZE.
C     NAFER
C
C     KPHOT,    KPLUS, KX, KY, SPIN OF PHOTON IN FOCK STATES.
C     KXPHOT,   ARRAYS OF DIMENSION NSIZE.
C     KYPHOT,
C     ISPHOT
C
C     KFERM,    KPLUS, KX, KY, SPIN OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM,
C     ISFERM
C
C     KAFER,    KPLUS, KX, KY, SPIN OF ANTI-FERMION IN FOCK STATES.
C     KXAFER,   ARRAYS OF DIMENSION NSIZE.
C     KYAFER,
C     ISAFER
C
C     TEMPWF    WORKING ARRAY OF DIMENSION NSIZEO.
C
C     REALH1L,
C     REALH1T,  WORKING ARRAYS OF DIMENSION NSIZE.
C     AIMH1T
C
C
C     USAGE NOTES:
C     1) FERMION CHARGE IS ASSUMED TO BE -1.
C     2) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     3) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(NSIZE),AIMWF(NSIZE),
     $  NPHOT(NSIZE),NFERM(NSIZE),NAFER(NSIZE),
     $  KPHOT(NSIZE),KXPHOT(NSIZE),KYPHOT(NSIZE),ISPHOT(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE),ISFERM(NSIZE),
     $  KAFER(NSIZE),KXAFER(NSIZE),KYAFER(NSIZE),ISAFER(NSIZE)
```

```
C-------------------------------------------------------
C     FOLLOWING NEEDED FOR VECTORIZATION.
C-------------------------------------------------------
      DIMENSION REALH1L(NSIZE),REALH1T(NSIZE),AIMH1T(NSIZE),
     $  TEMPWF(NSIZEO)

      PI = 3.141592653589793DO
      ALPERP2 = ALPERP**2
      PHOTMAS2 = PHOTMASS**2
      FERMMAS2 = FERMMASS**2
      PERPFACT = PI**2/ALPERP2
      AMSQARED = FERMMAS2*(2.0DO - .25*ALPHAG**2)**2

C-------------------------------------------------------
C     INITIALIZE TO ZERO.
C-------------------------------------------------------
      DO 50 ISTATE = 1,NSTATES
       REALWF(ISTATE) = 0.0DO
       AIMWF(ISTATE) = 0.0DO
 50   CONTINUE

C-------------------------------------------------------
C     PMINUS_IN = P_MINUS OF WHOLE SYSTEM.
C-------------------------------------------------------
      IF (ICHARGE .EQ. 0) THEN
       AKMINUSI = AMSQARED/DFLOAT(KPLUSTOT)
      ENDIF
      IF (ICHARGE .EQ. -1) THEN
       AKMINUSI = FERMMAS2/DFLOAT(KPLUSTOT)
      ENDIF

C-------------------------------------------------------
C     WORK OUT WAVE FUNC FOR STATES WITH 0 PHOTONS.
C-------------------------------------------------------
      IF (ICHARGE .EQ. 0) THEN
       DO 100 ISTATE = 1,NSTATEOP
        REALWF(ISTATE) = FFBARWF(KPLUSTOT,ALPHAG,
     $                    PERPFACT,FERMMAS2,AMSQARED,PARA1,IFERMUP,
     $                    KFERM(ISTATE),KXFERM(ISTATE),KYFERM(ISTATE),
     $                    ISFERM(ISTATE),ISAFER(ISTATE))
        AIMWF(ISTATE) = 0.0DO
        TEMPWF(ISTATE) = REALWF(ISTATE)
 100   CONTINUE
      ENDIF
      IF (ICHARGE .EQ. -1) THEN
       DO 200 ISTATE = 1,NSTATEOP
        REALWF(ISTATE) = FERMWF(KPLUSTOT,KFERM(ISTATE),
     $                    KXFERM(ISTATE),KYFERM(ISTATE),ISFERM(ISTATE))
        AIMWF(ISTATE) = 0.0DO
        TEMPWF(ISTATE) = REALWF(ISTATE)
200   CONTINUE
      ENDIF

C-------------------------------------------------------
C     WORK OUT WAVE FUNC FOR STATES WITH 1 PHOTON.
```

```
C     DO THIS BY LOOPING OVER STATES WITH 0 PHOTONS
C     AND FINDING NON-ZERO MATRIX ELEMENTS.
C-------------------------------------------------------------
C-------------------------------------------------------------
C     LOOP OVER 1 PAIR, 0 PHOTON STATES.
C-------------------------------------------------------------
      DO 110 INSTATE = 1,NSTATEOP
        IF (REALWF(INSTATE) .EQ. 0.0D0) GOTO 110

C-------------------------------------------------------------
C     LOOP OVER 1 PAIR, 1 PHOTON STATES.
C-------------------------------------------------------------
C       INITIALIZE TO ZERO.
        DO 120 IOUTSTAT = NSTATEOP+1,NSTATES
         REALH1L(IOUTSTAT) = 0.0D0
         REALH1T(IOUTSTAT) = 0.0D0
         AIMH1T(IOUTSTAT) = 0.0D0
 120    CONTINUE

C-------------------------------------------------------------
C     CALCULATE DIAGRAMS 1*,4*,5*.
C-------------------------------------------------------------
        DO 140 IOUTSTAT = NSTATEOP+1,NSTATES

C         CHECK IF SPECTATORS MATCH UP.
C         NOTE THAT KAFER, KXAFER, KYAFER, ISAFER ARE ALL 0
C         FOR ICHARGE = -1.
          IF (KAFER(INSTATE) .NE. KAFER(IOUTSTAT)) GOTO 140
          IF (KXAFER(INSTATE) .NE. KXAFER(IOUTSTAT)) GOTO 140
          IF (KYAFER(INSTATE) .NE. KYAFER(IOUTSTAT)) GOTO 140
          IF (ISAFER(INSTATE) .NE. ISAFER(IOUTSTAT)) GOTO 140

C         HAVE INTERACTION.  DETERMINE MATRIX ELEMENT.

C         DIAGRAM 1*
          IF ((ISPHOT(IOUTSTAT) .EQ. +1) .AND.
     $        (ISFERM(IOUTSTAT) .EQ. -1) .AND.
     $        (ISFERM(INSTATE) .EQ. +1)) THEN
            REALH1L(IOUTSTAT) = REALH1L(IOUTSTAT)
     $        + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $          *( 1.0D0/DFLOAT(KFERM(IOUTSTAT))
     $            -1.0D0/DFLOAT(KFERM(INSTATE)))
          ENDIF

          IF ((ISPHOT(IOUTSTAT) .EQ. -1) .AND.
     $        (ISFERM(IOUTSTAT) .EQ. +1) .AND.
     $        (ISFERM(INSTATE) .EQ. -1)) THEN
            REALH1L(IOUTSTAT) = REALH1L(IOUTSTAT)
     $        + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $          *( 1.0D0/DFLOAT(KFERM(IOUTSTAT))
     $            -1.0D0/DFLOAT(KFERM(INSTATE)))
          ENDIF

C         DIAGRAM 4*
          IF ((ISPHOT(IOUTSTAT) .EQ. +1) .AND.
```

```
     $       (ISFERM(IOUTSTAT) .EQ. +1) .AND.
     $       (ISFERM(INSTATE) .EQ. +1)) THEN
             REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)
     $         + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $           *(-DFLOAT(KXPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $             +DFLOAT(KXFERM(IOUTSTAT))/DFLOAT(KFERM(IOUTSTAT)))
             AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)
     $         + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $           *( DFLOAT(KYPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $             -DFLOAT(KYFERM(IOUTSTAT))/DFLOAT(KFERM(IOUTSTAT)))
           ENDIF

           IF ((ISPHOT(IOUTSTAT) .EQ. -1) .AND.
     $       (ISFERM(IOUTSTAT) .EQ. -1) .AND.
     $       (ISFERM(INSTATE) .EQ. -1)) THEN
             REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)
     $         + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $           *( DFLOAT(KXPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $             -DFLOAT(KXFERM(IOUTSTAT))/DFLOAT(KFERM(IOUTSTAT)))
             AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)
     $         + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $           *( DFLOAT(KYPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $             -DFLOAT(KYFERM(IOUTSTAT))/DFLOAT(KFERM(IOUTSTAT)))
           ENDIF

C          DIAGRAM 5*
           IF ((ISPHOT(IOUTSTAT) .EQ. -1) .AND.
     $       (ISFERM(IOUTSTAT) .EQ. +1) .AND.
     $       (ISFERM(INSTATE) .EQ. +1)) THEN
             REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)
     $         + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $           *( DFLOAT(KXPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $             -DFLOAT(KXFERM(INSTATE))/DFLOAT(KFERM(INSTATE)))
             AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)
     $         + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $           *( DFLOAT(KYPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $             -DFLOAT(KYFERM(INSTATE))/DFLOAT(KFERM(INSTATE)))
           ENDIF

           IF ((ISPHOT(IOUTSTAT) .EQ. +1) .AND.
     $       (ISFERM(IOUTSTAT) .EQ. -1) .AND.
     $       (ISFERM(INSTATE) .EQ. -1)) THEN
             REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)
     $         + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $           *(-DFLOAT(KXPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $             +DFLOAT(KXFERM(INSTATE))/DFLOAT(KFERM(INSTATE)))
             AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)
     $         + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $           *( DFLOAT(KYPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $             -DFLOAT(KYFERM(INSTATE))/DFLOAT(KFERM(INSTATE)))
           ENDIF

C        END OF DIAGRAMS 1*,4*,5*.
 140     CONTINUE
```

```
C-------------------------------------------------------
C     CALCULATE DIAGRAMS 2*,6*,7*.  ONLY HAVE THESE IF CHARGE = 0.
C-------------------------------------------------------
        IF (ICHARGE .EQ. -1) GOTO 165
        IF (NOAFERI .EQ. 1) GOTO 165
        DO 160 IOUTSTAT = NSTATEOP+1,NSTATES

C        CHECK IF SPECTATORS MATCH UP.
         IF (KFERM(INSTATE) .NE. KFERM(IOUTSTAT)) GOTO 160
         IF (KXFERM(INSTATE) .NE. KXFERM(IOUTSTAT)) GOTO 160
         IF (KYFERM(INSTATE) .NE. KYFERM(IOUTSTAT)) GOTO 160
         IF (ISFERM(INSTATE) .NE. ISFERM(IOUTSTAT)) GOTO 160

C        HAVE INTERACTION.  DETERMINE MATRIX ELEMENT.

C        DIAGRAM 2*
         IF ((ISPHOT(IOUTSTAT) .EQ. +1) .AND.
     $       (ISAFER(IOUTSTAT) .EQ. -1) .AND.
     $       (ISAFER(INSTATE) .EQ. +1)) THEN
           REALH1L(IOUTSTAT) = REALH1L(IOUTSTAT)
     $       + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $         *(-1.0D0/DFLOAT(KAFER(IOUTSTAT))
     $           +1.0D0/DFLOAT(KAFER(INSTATE)))
         ENDIF

         IF ((ISPHOT(IOUTSTAT) .EQ. -1) .AND.
     $       (ISAFER(IOUTSTAT) .EQ. +1) .AND.
     $       (ISAFER(INSTATE) .EQ. -1)) THEN
           REALH1L(IOUTSTAT) = REALH1L(IOUTSTAT)
     $       + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $         *(-1.0D0/DFLOAT(KAFER(IOUTSTAT))
     $           +1.0D0/DFLOAT(KAFER(INSTATE)))
         ENDIF

C        DIAGRAM 6*
         IF ((ISPHOT(IOUTSTAT) .EQ. +1) .AND.
     $       (ISAFER(IOUTSTAT) .EQ. -1) .AND.
     $       (ISAFER(INSTATE) .EQ. -1)) THEN
           REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)
     $       + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $         *( DFLOAT(KXPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $           -DFLOAT(KXAFER(INSTATE))/DFLOAT(KAFER(INSTATE)))
           AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)
     $       + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $         *(-DFLOAT(KYPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $           +DFLOAT(KYAFER(INSTATE))/DFLOAT(KAFER(INSTATE)))
         ENDIF

         IF ((ISPHOT(IOUTSTAT) .EQ. -1) .AND.
     $       (ISAFER(IOUTSTAT) .EQ. +1) .AND.
     $       (ISAFER(INSTATE) .EQ. +1)) THEN
           REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)
     $       + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $         *(-DFLOAT(KXPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $           +DFLOAT(KXAFER(INSTATE))/DFLOAT(KAFER(INSTATE)))
```

```
            AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)
     $        + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $          *(-DFLOAT(KYPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $            +DFLOAT(KYAFER(INSTATE))/DFLOAT(KAFER(INSTATE)))
          ENDIF

C         DIAGRAM 7*
          IF ((ISPHOT(IOUTSTAT) .EQ. -1) .AND.
     $        (ISAFER(IOUTSTAT) .EQ. -1) .AND.
     $        (ISAFER(INSTATE) .EQ. -1)) THEN
            REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)
     $        + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $          *(-DFLOAT(KXPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $            +DFLOAT(KXAFER(IOUTSTAT))/DFLOAT(KAFER(IOUTSTAT)))
            AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)
     $        + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $          *(-DFLOAT(KYPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $            +DFLOAT(KYAFER(IOUTSTAT))/DFLOAT(KAFER(IOUTSTAT)))
          ENDIF

          IF ((ISPHOT(IOUTSTAT) .EQ. +1) .AND.
     $        (ISAFER(IOUTSTAT) .EQ. +1) .AND.
     $        (ISAFER(INSTATE) .EQ. +1)) THEN
            REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)
     $        + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $          *( DFLOAT(KXPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $            -DFLOAT(KXAFER(IOUTSTAT))/DFLOAT(KAFER(IOUTSTAT)))
            AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)
     $        + 1.0D0/DSQRT(DFLOAT(KPHOT(IOUTSTAT)))
     $          *(-DFLOAT(KYPHOT(IOUTSTAT))/DFLOAT(KPHOT(IOUTSTAT))
     $            +DFLOAT(KYAFER(IOUTSTAT))/DFLOAT(KAFER(IOUTSTAT)))
          ENDIF

C       END OF DIAGRAMS 2*,6*,7*.
 160    CONTINUE
 165    CONTINUE

C-------------------------------------------------------
C     SUM CONTRIBUTIONS TO 1 PHOTON VARIATIONAL WAVE FUNC.
C-------------------------------------------------------
        IF (ICHARGE .EQ. 0) THEN
         DO 180 IOUTSTAT = NSTATEOP+1,NSTATES
          IF ((REALH1L(IOUTSTAT) .NE. 0.0D0) .OR.
     $        (REALH1T(IOUTSTAT) .NE. 0.0D0) .OR.
     $        (AIMH1T(IOUTSTAT)  .NE. 0.0D0)) THEN

            REALH1L(IOUTSTAT) = REALH1L(IOUTSTAT)*FERMMASS/ALPERP
            REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)*PI/ALPERP2
            AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)*PI/ALPERP2

            REALEL = DSQRT(ALPHAG)*(REALH1L(IOUTSTAT)+REALH1T(IOUTSTAT))
            AIMEL = DSQRT(ALPHAG)*AIMH1T(IOUTSTAT)

            AKMINUS0
     $       =(PERPFACT*DFLOAT(KXPHOT(IOUTSTAT)**2 + KYPHOT(IOUTSTAT)**2)
```

```
     $        +PHOTMAS2)/DFLOAT(KPHOT(IOUTSTAT))
     $       +(PERPFACT*DFLOAT(KXFERM(IOUTSTAT)**2 + KYFERM(IOUTSTAT)**2)
     $        +FERMMAS2)/DFLOAT(KFERM(IOUTSTAT))
     $       +(PERPFACT*DFLOAT(KXAFER(IOUTSTAT)**2 + KYAFER(IOUTSTAT)**2)
     $        +FERMMAS2)/DFLOAT(KAFER(IOUTSTAT))

           D = PARA3*AKMINUSI - AKMINUSO

C************DIAGNOSTICS
C          WRITE (16,9100) IOUTSTAT,INSTATE,REALEL,
C    $                     AIMEL,TEMPWF(INSTATE),AKMINUSI,AKMINUSO,D
C9100      FORMAT (' IOUTSTATE INSTATE  REALEL   AIMEL ',
C    $             ' FFBARWF AKMINUSI AKIMINUSO    D'/2I8,6F9.4)
C************

           REALWF(IOUTSTAT) = REALWF(IOUTSTAT)
     $                      + PARA2*REALEL*TEMPWF(INSTATE)/D
           AIMWF(IOUTSTAT)  = AIMWF(IOUTSTAT)
     $                      + PARA2*AIMEL*TEMPWF(INSTATE)/D
          ENDIF
 180     CONTINUE
        ENDIF

        IF (ICHARGE .EQ. -1) THEN
         DO 185 IOUTSTAT = NSTATEOP+1,NSTATES
          IF ((REALH1L(IOUTSTAT) .NE. 0.0D0) .OR.
     $        (REALH1T(IOUTSTAT) .NE. 0.0D0) .OR.
     $        (AIMH1T(IOUTSTAT)  .NE. 0.0D0)) THEN

           REALH1L(IOUTSTAT) = REALH1L(IOUTSTAT)*FERMMASS/ALPERP
           REALH1T(IOUTSTAT) = REALH1T(IOUTSTAT)*PI/ALPERP2
           AIMH1T(IOUTSTAT) = AIMH1T(IOUTSTAT)*PI/ALPERP2

           REALEL = DSQRT(ALPHAG)*(REALH1L(IOUTSTAT)+REALH1T(IOUTSTAT))
           AIMEL = DSQRT(ALPHAG)*AIMH1T(IOUTSTAT)

           AKMINUSO
     $      =(PERPFACT*DFLOAT(KXPHOT(IOUTSTAT)**2 + KYPHOT(IOUTSTAT)**2)
     $        +PHOTMAS2)/DFLOAT(KPHOT(IOUTSTAT))
     $       +(PERPFACT*DFLOAT(KXFERM(IOUTSTAT)**2 + KYFERM(IOUTSTAT)**2)
     $        +FERMMAS2)/DFLOAT(KFERM(IOUTSTAT))

           D = PARA3*AKMINUSI - AKMINUSO

C************DIAGNOSTICS
C          WRITE (16,9100) IOUTSTAT,INSTATE,REALEL,
C    $                     AIMEL,TEMPWF(INSTATE),AKMINUSI,AKMINUSO,D
C9100      FORMAT (' IOUTSTATE INSTATE  REALEL   AIMEL ',
C    $             ' FFBARWF AKMINUSI AKIMINUSO    D'/2I8,6F9.4)
C************

           REALWF(IOUTSTAT) = REALWF(IOUTSTAT)
     $                      + PARA2*REALEL*TEMPWF(INSTATE)/D
           AIMWF(IOUTSTAT)  = AIMWF(IOUTSTAT)
     $                      + PARA2*AIMEL*TEMPWF(INSTATE)/D
```

```
          ENDIF
 185     CONTINUE
        ENDIF

 110  CONTINUE

C-------------------------------------------------------
C     NORMALIZE VARIATIONAL WAVE FUNCTION.
C-------------------------------------------------------
      WFNORM = 0.0D0
      DO 300 ISTATE = 1,NSTATES
       WFNORM = WFNORM + REALWF(ISTATE)**2 + AIMWF(ISTATE)**2
 300  CONTINUE
      WFNORM = DSQRT(WFNORM)
      DO 301 ISTATE = 1,NSTATES
       REALWF(ISTATE) = REALWF(ISTATE)/WFNORM
       AIMWF(ISTATE) = AIMWF(ISTATE)/WFNORM
 301  CONTINUE

      RETURN
      END


      FUNCTION FFBARWF(KPLUSTOT,ALPHAG,PERPFACT,FERMMAS2,AMSQARED,
     $                 PARA1,IFERMUP,KFERM,KXFERM,KYFERM,ISFERM,ISAFER)
C-------------------------------------------------------
C     THIS ROUTINE RETURNS THE VARIATIONAL WAVE FUNCTION
C     FOR THE INPUTTED FFBAR STATE.
C
C     MASSES ARE IN UNITS OF ELECMASS.
C     LENGTHS ARE IN UNITS OF 1/ELECMASS.
C-------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      AKPERPSQ = PERPFACT*(DFLOAT(KXFERM)**2+DFLOAT(KYFERM)**2)
      X = DFLOAT(KFERM)/DFLOAT(KPLUSTOT)

      IF (IFERMUP .EQ. 0) THEN
        IF ((ISFERM .EQ. +1) .AND. (ISAFER .EQ. -1)) THEN
          FFBARWF = +1.0D0/
     $           (PARA1*AMSQARED - (AKPERPSQ+FERMMAS2)/(X*(1.0D0-X)))**2
         ELSEIF ((ISFERM .EQ. -1) .AND. (ISAFER .EQ. +1)) THEN
          FFBARWF = -1.0D0/
     $           (PARA1*AMSQARED - (AKPERPSQ+FERMMAS2)/(X*(1.0D0-X)))**2
         ELSE
          FFBARWF = 0.0D0
        ENDIF
      ENDIF
      IF (IFERMUP .EQ. 1) THEN
        FFBARWF = +1.0D0/
     $         (PARA1*AMSQARED - (AKPERPSQ+FERMMAS2)/(X*(1.0D0-X)))**2
      ENDIF

      RETURN
```

```
      END


      FUNCTION FERMWF(KPLUSTOT,KFERM,KXFERM,KYFERM,ISFERM)
C-----------------------------------------------------------
C     THIS ROUTINE RETURNS THE VARIATIONAL WAVE FUNCTION
C     FOR THE INPUTTED FERMION STATE.
C
C     MASSES ARE IN UNITS OF ELECMASS.
C     LENGTHS ARE IN UNITS OF 1/ELECMASS.
C-----------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      FERMWF = 0.0D0
      IF ((KFERM .EQ. KPLUSTOT) .AND.
     $    (KXFERM .EQ. 0) .AND.
     $    (KYFERM .EQ. 0) .AND.
     $    (ISFERM .EQ. 1)) THEN
        FERMWF = 1.0D0
      ENDIF

      RETURN
      END


      SUBROUTINE HONR(REMSQHO,NSIZE,NSTATES,REALWF,AIMWF,
     $                ALPERP,PHOTMASS,FERMMASS,
     $                NPHOT,NFERM,NAFER,
     $                KPHOT,KXPHOT,KYPHOT,
     $                KFERM,KXFERM,KYFERM,
     $                KAFER,KXAFER,KYAFER)
C-----------------------------------------------------------
C     THIS ROUTINE RETURNS THE VALUE OF THE HAMILTONIAN
C     BETWEEN THE INPUTTED VARIATIONAL STATES.
C
C
C     OUTPUT VARIABLES:
C     REMSQHO   CONTRIBUTION TO MATRIX ELEMENT FROM HO.
C
C
C     INPUT VARIABLES:
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C               TO NSTATES.
C
C     NSTATES   NUMBER OF FOCK STATES.
C
C     REALWF    REAL PART OF FOCK STATE WAVE FUNCTIONS.
C
C     AIMWF     IMAGINARY PART OF FOCK STATE WAVE FUNCTIONS.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     PHOTMASS  PHOTON MASS IN LAGRANGIAN.
```

```
C
C     FERMMASS  FERMION MASS IN LAGRANGIAN.
C
C     NPHOT,    # PHOTONS, FERMIONS, ANTI-FERMIONS IN FOCK STATES.
C     NFERM,    ARRAYS OF DIMENSION NSIZE.
C     NAFER
C
C     KPHOT,    KPLUS, KX, KY OF PHOTON IN FOCK STATES.
C     KXPHOT,   ARRAYS OF DIMENSION NSIZE.
C     KYPHOT
C
C     KFERM,    KPLUS, KX, KY OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM
C
C     KAFER,    KPLUS, KX, KY OF ANTI-FERMION IN FOCK STATES.
C     KXAFER,   ARRAYS OF DIMENSION NSIZE.
C     KYAFER
C
C
C     USAGE NOTES:
C     1) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     2) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-----------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(NSIZE),AIMWF(NSIZE),
     $  NPHOT(NSIZE),NFERM(NSIZE),NAFER(NSIZE),
     $  KPHOT(NSIZE),KXPHOT(NSIZE),KYPHOT(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE),
     $  KAFER(NSIZE),KXAFER(NSIZE),KYAFER(NSIZE)

      PI = 3.141592653589793DO
      PHOTMAS2 = PHOTMASS**2
      FERMMAS2 = FERMMASS**2
      PERPFACT = PI**2/ALPERP**2

C-----------------------------------------------------------
C     INITIALIZE TO ZERO.
C-----------------------------------------------------------
      REMSQHO = 0.0DO

C-----------------------------------------------------------
C     LOOP OVER DIAGONAL MATRIX ELEMENTS.
C-----------------------------------------------------------
      DO 10 ISTATE = 1,NSTATES

C       INITIALIZE TO ZERO.
        REALHO = 0.0DO

C-----------------------------------------------------------
C     CALCULATE HO.
C-----------------------------------------------------------
```

```
      IF (NPHOT(ISTATE) .EQ. 1) THEN
       AKPERPSQ = PERPFACT*DFLOAT(KXPHOT(ISTATE)**2+KYPHOT(ISTATE)**2)
       REALHO = REALHO
     $         + (PHOTMAS2 + AKPERPSQ)/DFLOAT(KPHOT(ISTATE))
      ENDIF

      AKPERPSQ = PERPFACT*DFLOAT(KXFERM(ISTATE)**2+KYFERM(ISTATE)**2)
      REALHO = REALHO
     $         + (FERMMAS2 + AKPERPSQ)/DFLOAT(KFERM(ISTATE))

      IF (NAFER(ISTATE) .EQ. 1) THEN
       AKPERPSQ = PERPFACT*DFLOAT(KXAFER(ISTATE)**2+KYAFER(ISTATE)**2)
       REALHO = REALHO
     $         + (FERMMAS2 + AKPERPSQ)/DFLOAT(KAFER(ISTATE))
      ENDIF

C************DIAGNOSTICS
C       WRITE (16,910) ISTATE,REALHO
C910    FORMAT (' ISTATE REALHO =',I7,F11.4)
C************

      REMSQHO  = REMSQHO
     $   +REALHO*(REALWF(ISTATE)**2 + AIMWF(ISTATE)**2)

C     FINISHED WITH THIS FOCK STATE.  GO TO NEXT FOCK STATE.
 10   CONTINUE

      RETURN
      END


      SUBROUTINE H1NR(REMSQH1L,REMSQH1T,
     $                NSIZE,NSTATES,NSTATEOP,REALWF,AIMWF,
     $                ALPHAG,ICHARGE,ALPERP,FERMMASS,NOAFERI,
     $                NPHOT,NFERM,NAFER,
     $                KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $                KFERM,KXFERM,KYFERM,ISFERM,
     $                KAFER,KXAFER,KYAFER,ISAFER,
     $                REALH1L,REALH1T,AIMH1T)
C-------------------------------------------------------
C     THIS ROUTINE RETURNS THE VALUE OF THE HAMILTONIAN
C     BETWEEN THE INPUTTED VARIATIONAL STATES.
C
C
C     OUTPUT VARIABLES:
C     REMSQH1L  CONTRIBUTION TO MATRIX ELEMENT FROM LONG. VERTEX.
C
C     REMSQH1T  CONTRIBUTION TO MATRIX ELEMENT FROM TRANS. VERTEX.
C
C
C     INPUT VARIABLES:
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C               TO NSTATES.
C
```

```
C     NSTATES   NUMBER OF FOCK STATES.
C
C     NSTATEOP  NUMBER OF FOCK STATES WITH NO PHOTONS.
C
C     REALWF    REAL PART OF FOCK STATE WAVE FUNCTIONS.
C
C     AIMWF     IMAGINARY PART OF FOCK STATE WAVE FUNCTIONS.
C
C     ALPHAG    VALUE OF COUPLING CONSTANT (=G**2/4PI).
C
C     ICHARGE   TOTAL CHARGE.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     FERMMASS  FERMION MASS IN LAGRANGIAN.
C
C     NOAFERI   NO INTERACTIONS WITH ANTI-FERMION.
C
C     NPHOT,    # PHOTONS, FERMIONS, ANTI-FERMIONS IN FOCK STATES.
C     NFERM,    ARRAYS OF DIMENSION NSIZE.
C     NAFER
C
C     KPHOT,    KPLUS, KX, KY, SPIN OF PHOTON IN FOCK STATES.
C     KXPHOT,   ARRAYS OF DIMENSION NSIZE.
C     KYPHOT,
C     ISPHOT
C
C     KFERM,    KPLUS, KX, KY, SPIN OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM,
C     ISFERM
C
C     KAFER,    KPLUS, KX, KY, SPIN OF ANTI-FERMION IN FOCK STATES.
C     KXAFER,   ARRAYS OF DIMENSION NSIZE.
C     KYAFER,
C     ISAFER
C
C     REALH1L,
C     REALH1T,  WORKING ARRAYS OF DIMENSION NSIZE.
C     AIMH1T
C
C
C     USAGE NOTES:
C     1) FERMION CHARGE IS ASSUMED TO BE -1.
C     2) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     3) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-----------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(NSIZE),AIMWF(NSIZE),
     $  NPHOT(NSIZE),NFERM(NSIZE),NAFER(NSIZE),
     $  KPHOT(NSIZE),KXPHOT(NSIZE),KYPHOT(NSIZE),ISPHOT(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE),ISFERM(NSIZE),
```

```
     $  KAFER(NSIZE),KXAFER(NSIZE),KYAFER(NSIZE),ISAFER(NSIZE)

C-----------------------------------------------------------
C     FOLLOWING NEEDED FOR VECTORIZATION.
C-----------------------------------------------------------
      DIMENSION
     $  REALH1L(NSIZE),REALH1T(NSIZE),AIMH1T(NSIZE)

      PI = 3.141592653589793D0
      ALPERP2 = ALPERP**2

C-----------------------------------------------------------
C     INITIALIZE TO ZERO.
C-----------------------------------------------------------
      REMSQH1L = 0.0D0
      REMSQH1T = 0.0D0

C-----------------------------------------------------------
C     LOOP OVER OUTGOING FOCK STATES.
C-----------------------------------------------------------
      DO 20 IOUTSTAT = 1,NSTATEOP
       IF ((REALWF(IOUTSTAT) .EQ. 0.0D0) .AND.
     $     ( AIMWF(IOUTSTAT) .EQ. 0.0D0)) GOTO 20

C-----------------------------------------------------------
C     LOOP OVER INCOMING FOCK STATES.
C     ASSUME HAMILTONIAN IS HERMITIAN, SO ONLY NEED TO CONSIDER
C     INSTATE .GT. IOUTSTAT.
C-----------------------------------------------------------
C      INITIALIZE TO ZERO.
       DO 30 INSTATE = NSTATEOP+1,NSTATES
        REALH1L(INSTATE) = 0.0D0
        REALH1T(INSTATE) = 0.0D0
        AIMH1T(INSTATE) = 0.0D0
 30    CONTINUE

C-----------------------------------------------------------
C     CALCULATE DIAGRAMS 1,4,5.
C-----------------------------------------------------------
       DO 100 INSTATE = NSTATEOP+1,NSTATES
        IF ((REALWF(INSTATE) .EQ. 0.0D0) .AND.
     $      ( AIMWF(INSTATE) .EQ. 0.0D0)) GOTO 100

C       CHECK IF SPECTATORS MATCH UP.
C       NOTE THAT KAFER, KXAFER, KYAFER, ISAFER ARE ALL 0
C       FOR ICHARGE = -1.
        IF (KAFER(INSTATE) .NE. KAFER(IOUTSTAT)) GOTO 100
        IF (KXAFER(INSTATE) .NE. KXAFER(IOUTSTAT)) GOTO 100
        IF (KYAFER(INSTATE) .NE. KYAFER(IOUTSTAT)) GOTO 100
        IF (ISAFER(INSTATE) .NE. ISAFER(IOUTSTAT)) GOTO 100

C       HAVE INTERACTION.  DETERMINE MATRIX ELEMENT.

C       DIAGRAM 1
        IF ((ISPHOT(INSTATE) .EQ. +1) .AND.
```

```
     $      (ISFERM(INSTATE) .EQ. -1) .AND.
     $      (ISFERM(IOUTSTAT) .EQ. +1)) THEN
            REALH1L(INSTATE) = REALH1L(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( 1.0D0/DFLOAT(KFERM(INSTATE))
     $          -1.0D0/DFLOAT(KFERM(IOUTSTAT)))
            ENDIF

            IF ((ISPHOT(INSTATE) .EQ. -1) .AND.
     $      (ISFERM(INSTATE) .EQ. +1) .AND.
     $      (ISFERM(IOUTSTAT) .EQ. -1)) THEN
            REALH1L(INSTATE) = REALH1L(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( 1.0D0/DFLOAT(KFERM(INSTATE))
     $          -1.0D0/DFLOAT(KFERM(IOUTSTAT)))
            ENDIF

C           DIAGRAM 4
            IF ((ISPHOT(INSTATE) .EQ. +1) .AND.
     $      (ISFERM(INSTATE) .EQ. +1) .AND.
     $      (ISFERM(IOUTSTAT) .EQ. +1)) THEN
            REALH1T(INSTATE) = REALH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *(-DFLOAT(KXPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          +DFLOAT(KXFERM(INSTATE))/DFLOAT(KFERM(INSTATE)))
            AIMH1T(INSTATE) = AIMH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *(-DFLOAT(KYPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          +DFLOAT(KYFERM(INSTATE))/DFLOAT(KFERM(INSTATE)))
            ENDIF

            IF ((ISPHOT(INSTATE) .EQ. -1) .AND.
     $      (ISFERM(INSTATE) .EQ. -1) .AND.
     $      (ISFERM(IOUTSTAT) .EQ. -1)) THEN
            REALH1T(INSTATE) = REALH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( DFLOAT(KXPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          -DFLOAT(KXFERM(INSTATE))/DFLOAT(KFERM(INSTATE)))
            AIMH1T(INSTATE) = AIMH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *(-DFLOAT(KYPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          +DFLOAT(KYFERM(INSTATE))/DFLOAT(KFERM(INSTATE)))
            ENDIF

C           DIAGRAM 5
            IF ((ISPHOT(INSTATE) .EQ. -1) .AND.
     $      (ISFERM(INSTATE) .EQ. +1) .AND.
     $      (ISFERM(IOUTSTAT) .EQ. +1)) THEN
            REALH1T(INSTATE) = REALH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( DFLOAT(KXPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          -DFLOAT(KXFERM(IOUTSTAT))/DFLOAT(KFERM(IOUTSTAT)))
            AIMH1T(INSTATE) = AIMH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *(-DFLOAT(KYPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
```

```
     $          +DFLOAT(KYFERM(IOUTSTAT))/DFLOAT(KFERM(IOUTSTAT)))
         ENDIF

         IF ((ISPHOT(INSTATE) .EQ. +1) .AND.
     $       (ISFERM(INSTATE) .EQ. -1) .AND.
     $       (ISFERM(IOUTSTAT) .EQ. -1)) THEN
           REALH1T(INSTATE) = REALH1T(INSTATE)
     $       + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $         *(-DFLOAT(KXPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $           +DFLOAT(KXFERM(IOUTSTAT))/DFLOAT(KFERM(IOUTSTAT)))
           AIMH1T(INSTATE) = AIMH1T(INSTATE)
     $       + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $         *(-DFLOAT(KYPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $           +DFLOAT(KYFERM(IOUTSTAT))/DFLOAT(KFERM(IOUTSTAT)))
         ENDIF

C      END OF DIAGRAMS 1,4,5.
 100   CONTINUE

C-------------------------------------------------------
C     CALCULATE DIAGRAMS 2,6,7.  ONLY HAVE THESE IF ICHARGE=0.
C-------------------------------------------------------
       IF (ICHARGE .EQ. -1) GOTO 205
       IF (NOAFERI .EQ. 1) GOTO 205
       DO 200 INSTATE = NSTATEOP+1,NSTATES
        IF ((REALWF(INSTATE) .EQ. 0.0D0) .AND.
     $      ( AIMWF(INSTATE) .EQ. 0.0D0)) GOTO 200

C       CHECK IF SPECTATORS MATCH UP.
        IF (KFERM(INSTATE) .NE. KFERM(IOUTSTAT)) GOTO 200
        IF (KXFERM(INSTATE) .NE. KXFERM(IOUTSTAT)) GOTO 200
        IF (KYFERM(INSTATE) .NE. KYFERM(IOUTSTAT)) GOTO 200
        IF (ISFERM(INSTATE) .NE. ISFERM(IOUTSTAT)) GOTO 200

C       HAVE INTERACTION.  DETERMINE MATRIX ELEMENT.

C       DIAGRAM 2
        IF ((ISPHOT(INSTATE) .EQ. +1) .AND.
     $      (ISAFER(INSTATE) .EQ. -1) .AND.
     $      (ISAFER(IOUTSTAT) .EQ. +1)) THEN
          REALH1L(INSTATE) = REALH1L(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *(-1.0D0/DFLOAT(KAFER(INSTATE))
     $          +1.0D0/DFLOAT(KAFER(IOUTSTAT)))
        ENDIF

        IF ((ISPHOT(INSTATE) .EQ. -1) .AND.
     $      (ISAFER(INSTATE) .EQ. +1) .AND.
     $      (ISAFER(IOUTSTAT) .EQ. -1)) THEN
          REALH1L(INSTATE) = REALH1L(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *(-1.0D0/DFLOAT(KAFER(INSTATE))
     $          +1.0D0/DFLOAT(KAFER(IOUTSTAT)))
        ENDIF
```

```
C       DIAGRAM 6
        IF ((ISPHOT(INSTATE) .EQ. +1) .AND.
     $      (ISAFER(INSTATE) .EQ. -1) .AND.
     $      (ISAFER(IOUTSTAT) .EQ. -1)) THEN
          REALH1T(INSTATE) = REALH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( DFLOAT(KXPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          -DFLOAT(KXAFER(IOUTSTAT))/DFLOAT(KAFER(IOUTSTAT)))
          AIMH1T(INSTATE) = AIMH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( DFLOAT(KYPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          -DFLOAT(KYAFER(IOUTSTAT))/DFLOAT(KAFER(IOUTSTAT)))
        ENDIF

        IF ((ISPHOT(INSTATE) .EQ. -1) .AND.
     $      (ISAFER(INSTATE) .EQ. +1) .AND.
     $      (ISAFER(IOUTSTAT) .EQ. +1)) THEN
          REALH1T(INSTATE) = REALH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *(-DFLOAT(KXPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          +DFLOAT(KXAFER(IOUTSTAT))/DFLOAT(KAFER(IOUTSTAT)))
          AIMH1T(INSTATE) = AIMH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( DFLOAT(KYPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          -DFLOAT(KYAFER(IOUTSTAT))/DFLOAT(KAFER(IOUTSTAT)))
        ENDIF

C       DIAGRAM 7
        IF ((ISPHOT(INSTATE) .EQ. -1) .AND.
     $      (ISAFER(INSTATE) .EQ. -1) .AND.
     $      (ISAFER(IOUTSTAT) .EQ. -1)) THEN
          REALH1T(INSTATE) = REALH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *(-DFLOAT(KXPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          +DFLOAT(KXAFER(INSTATE))/DFLOAT(KAFER(INSTATE)))
          AIMH1T(INSTATE) = AIMH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( DFLOAT(KYPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          -DFLOAT(KYAFER(INSTATE))/DFLOAT(KAFER(INSTATE)))
        ENDIF

        IF ((ISPHOT(INSTATE) .EQ. +1) .AND.
     $      (ISAFER(INSTATE) .EQ. +1) .AND.
     $      (ISAFER(IOUTSTAT) .EQ. +1)) THEN
          REALH1T(INSTATE) = REALH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( DFLOAT(KXPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          -DFLOAT(KXAFER(INSTATE))/DFLOAT(KAFER(INSTATE)))
          AIMH1T(INSTATE) = AIMH1T(INSTATE)
     $      + 1.0D0/DSQRT(DFLOAT(KPHOT(INSTATE)))
     $        *( DFLOAT(KYPHOT(INSTATE))/DFLOAT(KPHOT(INSTATE))
     $          -DFLOAT(KYAFER(INSTATE))/DFLOAT(KAFER(INSTATE)))
        ENDIF

C     END OF DIAGRAMS 2,6,7.
```

```
 200   CONTINUE
 205   CONTINUE

C-------------------------------------------------------
C     SUM CONTRIBUTIONS TO H1.
C-------------------------------------------------------
       DO 300 INSTATE = NSTATEOP+1,NSTATES

C        COMMENT OUT THIS LINE FOR CRAY.
         IF ((REALH1L(INSTATE) .EQ. 0.0D0) .AND.
     $       (REALH1T(INSTATE) .EQ. 0.0D0) .AND.
     $       (AIMH1T(INSTATE)  .EQ. 0.0D0)) GOTO 300

         REALH1L(INSTATE) = REALH1L(INSTATE)*FERMMASS/ALPERP
         REALH1T(INSTATE) = REALH1T(INSTATE)*PI/ALPERP2
         AIMH1T(INSTATE) = AIMH1T(INSTATE)*PI/ALPERP2

C************DIAGNOSTICS
C        WRITE (16,920) INSTATE,IOUTSTAT,
C    $     REALH1L(INSTATE),REALH1T(INSTATE),AIMH1T(INSTATE)
C920     FORMAT (' INSTATE IOUTSTAT REALH1L REALH1T AIMH1T ='/
C    $          2I7,3F11.4)
C************

         REMSQH1L = REMSQH1L
     $     + 2.0*REALH1L(INSTATE)*DSQRT(ALPHAG)
     $             *( REALWF(IOUTSTAT)*REALWF(INSTATE)
     $               +AIMWF(IOUTSTAT)*AIMWF(INSTATE))
         REMSQH1T = REMSQH1T
     $     + 2.0*REALH1T(INSTATE)*DSQRT(ALPHAG)
     $             *( REALWF(IOUTSTAT)*REALWF(INSTATE)
     $               +AIMWF(IOUTSTAT)*AIMWF(INSTATE))
     $     + 2.0*AIMH1T(INSTATE)*DSQRT(ALPHAG)
     $             *(-REALWF(IOUTSTAT)*AIMWF(INSTATE)
     $               +AIMWF(IOUTSTAT)*REALWF(INSTATE))
 300   CONTINUE
 20    CONTINUE

       RETURN
       END


       SUBROUTINE H2NR(REMSQH2P,REMSQH2F,
     $                 NSIZE,NSTATES,NSTATEOP,REALWF,AIMWF,
     $                 ALPHAG,KPLUSTOT,ICHARGE,ALAMBDA,ALPERP,EPSILON,
     $                 RPHOMASS,RFERMASS,IUVFERM,NOAFERI,IH2PHOT,IH2FERM,
     $                 NPHOT,NFERM,NAFER,
     $                 KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $                 KFERM,KXFERM,KYFERM,ISFERM,
     $                 KAFER,KXAFER,KYAFER,ISAFER,
     $                 REALH2P,REALH2F)
C-------------------------------------------------------
C     THIS ROUTINE RETURNS THE VALUE OF THE HAMILTONIAN
C     BETWEEN THE INPUTTED VARIATIONAL STATES.
C
```

```
C
C     OUTPUT VARIABLES:
C     REMSQH2P  CONTRIBUTION TO MATRIX ELEMENT FROM INST. PHOTON.
C
C     REMSQH2F  CONTRIBUTION TO MATRIX ELEMENT FROM INST. FERMION.
C
C
C
C     INPUT VARIABLES:
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C               TO NSTATES.
C
C     NSTATES   NUMBER OF FOCK STATES.
C
C     NSTATEOP  NUMBER OF FOCK STATES WITH NO PHOTONS.
C
C     REALWF    REAL PART OF FOCK STATE WAVE FUNCTIONS.
C
C     AIMWF     IMAGINARY PART OF FOCK STATE WAVE FUNCTIONS.
C
C     ALPHAG    VALUE OF COUPLING CONSTANT (=G**2/4PI).
C
C     KPLUSTOT  TOTAL KPLUS OF INCOMING, OUTGOING STATES.
C
C     ICHARGE   TOTAL CHARGE.
C
C     ALAMBDA   VALUE OF CUT-OFF MASS.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     EPSILON   MINIMUM PHOTON INVMASS**2.
C
C     RPHOMASS  PHOTON MASS TO BE USED IN COVARIANT CUT-OFF.
C
C     RFERMASS  FERMION MASS TO BE USED IN COVARIANT CUT-OFF.
C
C     NOAFERI   NO INTERACTIONS WITH ANTI-FERMION.
C
C     IUVFERM   UV CUT-OFF ONLY APPLIED TO FERMIONS, ANTI-FERMIONS.
C
C     IH2PHOT   TURN ON INSTANTANEOUS PHOTON INTERACTION.
C
C     IH2FERM   TURN ON INSTANTANEOUS FERMION INTERACTION.
C
C     NPHOT,    # PHOTONS, FERMIONS, ANTI-FERMIONS IN FOCK STATES.
C     NFERM,    ARRAYS OF DIMENSION NSIZE.
C     NAFER
C
C     KPHOT,    KPLUS, KX, KY, SPIN OF PHOTON IN FOCK STATES.
C     KXPHOT,   ARRAYS OF DIMENSION NSIZE.
C     KYPHOT,
C     ISPHOT
C
C     KFERM,    KPLUS, KX, KY, SPIN OF FERMION IN THE FOCK STATES.
```

```
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM,
C     ISFERM
C
C     KAFER,    KPLUS, KX, KY, SPIN OF ANTI-FERMION IN FOCK STATES.
C     KXAFER,   ARRAYS OF DIMENSION NSIZE.
C     KYAFER,
C     ISAFER
C
C     REALH2P,  WORKING ARRAYS OF DIMENSION NSIZE.
C     REALH2F
C
C
C     USAGE NOTES:
C     1) THIS ROUTINE ASSUMES THAT THE TOTAL KPERP OF THE INCOMING
C        AND OUTGOING FOCK STATES IS ZERO.  THIS IS NECESSARY IN THE
C        CALCULATION OF INVARIANT MASSES.
C     2) FERMION CHARGE IS ASSUMED TO BE -1.
C     3) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     4) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-----------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(NSIZE),AIMWF(NSIZE),
     $  NPHOT(NSIZE),NFERM(NSIZE),NAFER(NSIZE),
     $  KPHOT(NSIZE),KXPHOT(NSIZE),KYPHOT(NSIZE),ISPHOT(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE),ISFERM(NSIZE),
     $  KAFER(NSIZE),KXAFER(NSIZE),KYAFER(NSIZE),ISAFER(NSIZE)

C-----------------------------------------------------------
C     FOLLOWING NEEDED FOR VECTORIZATION.
C-----------------------------------------------------------
      DIMENSION REALH2P(NSIZE),REALH2F(NSIZE)

      SMALL = 1.0D-13
      PI = 3.141592653589793D0
      ALPERP2 = ALPERP**2

      BETARPHO = (RPHOMASS*ALPERP/PI)**2
      BETARFER = (RFERMASS*ALPERP/PI)**2
      ALPHA1 = (ALAMBDA*ALPERP/PI)**2/DFLOAT(KPLUSTOT) + SMALL
      ALPHA2 = (ALPERP/PI)**2*(EPSILON/DFLOAT(KPLUSTOT)) - SMALL

C-----------------------------------------------------------
C     INITIALIZE TO ZERO.
C-----------------------------------------------------------
      REMSQH2P = 0.0D0
      REMSQH2F = 0.0D0

C-----------------------------------------------------------
C     CALCULATE DIAGRAM 13.  ONLY HAVE DIAGRAM 13 IF ICHARGE=0.
C     LOOP OVER OUTGOING FOCK STATES WITH 0 PHOTONS.
C-----------------------------------------------------------
```

```
      IF ((ICHARGE .EQ. 0) .AND. (IH2PHOT .EQ. 1)
     $    .AND. (NOAFERI .EQ. 0)) THEN
        DO 20 IOUTSTAT = 1,NSTATEOP
         IF ((REALWF(IOUTSTAT) .EQ. 0.0D0) .AND.
     $       ( AIMWF(IOUTSTAT) .EQ. 0.0D0)) GOTO 20

C-------------------------------------------------------
C     LOOP OVER INCOMING FOCK STATES WITH 0 PHOTONS.
C     ASSUME HAMILTONIAN IS HERMITIAN, SO ONLY NEED TO CONSIDER
C     IOUTSTAT .GE. INSTATE.
C-------------------------------------------------------
        DO 30 INSTATE = 1,IOUTSTAT
         IF ((REALWF(INSTATE) .EQ. 0.0D0) .AND.
     $       ( AIMWF(INSTATE) .EQ. 0.0D0)) GOTO 30

C        INITIALIZE TO ZERO.
         REALH2P(INSTATE) = 0.0D0

C        CHECK IF HAVE CORRECT SPINS.
         IF (ISFERM(INSTATE) .NE. ISFERM(IOUTSTAT)) GOTO 300
         IF (ISAFER(INSTATE) .NE. ISAFER(IOUTSTAT)) GOTO 300

C        NO MATRIX ELEMENT IF KFERM(INSTATE) = KFERM(IOUTSTAT).
         IF (KFERM(INSTATE) .EQ. KFERM(IOUTSTAT)) GOTO 300

C        NO MATRIX ELEMENT IF INVMASS**2 OF INSTANTANEOUS PHOTON
C        .LT. EPSILON.
         IF (
     $    ( (DFLOAT( (KXFERM(INSTATE) - KXFERM(IOUTSTAT))**2
     $             +(KYFERM(INSTATE) - KYFERM(IOUTSTAT))**2)
     $       +BETARPHO)/DFLOAT(IABS(KFERM(INSTATE) - KFERM(IOUTSTAT)))
     $    ) .LT. ALPHA2)
     $   GOTO 300

         IF (KFERM(INSTATE) .GT. KFERM(IOUTSTAT)) THEN

C         CHECK IF INVMASS**2 OF INTERMEDIATE STATE IS
C         .LE. LAMBDA**2.
          IF (IUVFERM .EQ. 0) THEN
           IF (
     $      ( (DFLOAT( (KXFERM(INSTATE) - KXFERM(IOUTSTAT))**2
     $               +(KYFERM(INSTATE) - KYFERM(IOUTSTAT))**2)
     $          +BETARPHO)/DFLOAT(KFERM(INSTATE) - KFERM(IOUTSTAT))
     $        +(DFLOAT(KXFERM(IOUTSTAT)**2 + KYFERM(IOUTSTAT)**2)
     $          +BETARFER)/DFLOAT(KFERM(IOUTSTAT))
     $        +(DFLOAT(KXAFER(INSTATE)**2 + KYAFER(INSTATE)**2)
     $          +BETARFER)/DFLOAT(KAFER(INSTATE))
     $      ) .GT. ALPHA1) GOTO 300
          ENDIF
          IF (IUVFERM .EQ. 1) THEN
           IF (
     $      ( (DFLOAT(KXFERM(IOUTSTAT)**2 + KYFERM(IOUTSTAT)**2)
     $          +BETARFER)/DFLOAT(KFERM(IOUTSTAT))
     $        +(DFLOAT(KXAFER(INSTATE)**2 + KYAFER(INSTATE)**2)
     $          +BETARFER)/DFLOAT(KAFER(INSTATE))
```

```
     $      ) .GT. ALPHA1) GOTO 300
            ENDIF

C           HAVE INTERACTION.  DETERMINE MATRIX ELEMENT.
            REALH2P(INSTATE) = REALH2P(INSTATE)
     $           - 1.0/DFLOAT((KFERM(INSTATE) - KFERM(IOUTSTAT))**2)
           ENDIF

           IF (KFERM(INSTATE) .LT. KFERM(IOUTSTAT)) THEN

C           CHECK IF INVMASS**2 OF INTERMEDIATE STATE IS
C           .LE. LAMBDA**2.
            IF (IUVFERM .EQ. 0) THEN
             IF (
     $        ( (DFLOAT( (KXFERM(IOUTSTAT) - KXFERM(INSTATE))**2
     $                 +(KYFERM(IOUTSTAT) - KYFERM(INSTATE))**2)
     $           +BETARPHO)/DFLOAT(KFERM(IOUTSTAT) - KFERM(INSTATE))
     $         +(DFLOAT(KXFERM(INSTATE)**2 + KYFERM(INSTATE)**2)
     $           +BETARFER)/DFLOAT(KFERM(INSTATE))
     $         +(DFLOAT(KXAFER(IOUTSTAT)**2 + KYAFER(IOUTSTAT)**2)
     $           +BETARFER)/DFLOAT(KAFER(IOUTSTAT))
     $        ) .GT. ALPHA1) GOTO 300
            ENDIF
            IF (IUVFERM .EQ. 1) THEN
             IF (
     $        ( (DFLOAT(KXFERM(INSTATE)**2 + KYFERM(INSTATE)**2)
     $           +BETARFER)/DFLOAT(KFERM(INSTATE))
     $         +(DFLOAT(KXAFER(IOUTSTAT)**2 + KYAFER(IOUTSTAT)**2)
     $           +BETARFER)/DFLOAT(KAFER(IOUTSTAT))
     $        ) .GT. ALPHA1) GOTO 300
            ENDIF

C           HAVE INTERACTION.  DETERMINE MATRIX ELEMENT.
            REALH2P(INSTATE) = REALH2P(INSTATE)
     $           - 1.0/DFLOAT((KFERM(IOUTSTAT) - KFERM(INSTATE))**2)
           ENDIF

C          END OF DIAGRAM 13.
 300       CONTINUE

           IF (REALH2P(INSTATE) .NE. 0.0D0) THEN
            REALH2P(INSTATE) = 2.0*REALH2P(INSTATE)/ALPERP2

C************DIAGNOSTICS
C           WRITE (16,920) INSTATE,IOUTSTAT,REALH2P(INSTATE)
C920        FORMAT (' INSTATE IOUTSTAT REALH2P =',2I7,2F11.4)
C************

            FACT = 2.0D0
            IF (INSTATE .EQ. IOUTSTAT) FACT = 1.0D0
            REMSQH2P = REMSQH2P + FACT*REALH2P(INSTATE)
     $              *ALPHAG*( REALWF(IOUTSTAT)*REALWF(INSTATE)
     $                       +AIMWF(IOUTSTAT)*AIMWF(INSTATE))
           ENDIF
```

```
C       FINISHED WITH THIS FOCK STATE.  GO TO NEXT FOCK STATE.
 30     CONTINUE
 20    CONTINUE
      ENDIF

C-----------------------------------------------------------
C     CALCULATE DIAGRAMS 16, 18.
C     LOOP OVER OUTGOING FOCK STATES WITH 1 PHOTON.
C-----------------------------------------------------------
      IF (IH2FERM .EQ. 1) THEN
       DO 40 IOUTSTAT = NSTATEOP+1,NSTATES
        IF ((REALWF(IOUTSTAT) .EQ. 0.0D0) .AND.
     $      ( AIMWF(IOUTSTAT) .EQ. 0.0D0)) GOTO 40

C-----------------------------------------------------------
C     LOOP OVER INCOMING FOCK STATES WITH 1 PHOTON.
C     ASSUME HAMILTONIAN IS HERMITIAN, SO ONLY NEED TO CONSIDER
C     IOUTSTAT .GE. INSTATE.
C-----------------------------------------------------------
        DO 50 INSTATE = NSTATEOP+1,IOUTSTAT
         IF ((REALWF(INSTATE) .EQ. 0.0D0) .AND.
     $       ( AIMWF(INSTATE) .EQ. 0.0D0)) GOTO 50

C        INITIALIZE TO ZERO.
         REALH2F(INSTATE) = 0.0D0

C-----------------------------------------------------------
C     GET RID OF UNNECESSARY CALLS TO H2NR.
C-----------------------------------------------------------
C        MAKE SURE THAT SPECTATOR MATCHES UP.
         IF (ICHARGE .EQ. 0) THEN
          IF ((KAFER(IOUTSTAT) .NE. KAFER(INSTATE)) .AND.
     $        (KFERM(IOUTSTAT) .NE. KFERM(INSTATE))) GOTO 50

C         FERMION IS THE SPECTATOR.
          IF ((KFERM(IOUTSTAT) .EQ. KFERM(INSTATE)) .AND.
     $        (KAFER(IOUTSTAT) .NE. KAFER(INSTATE))) THEN
           IF (KXFERM(IOUTSTAT) .NE. KXFERM(INSTATE)) GOTO 50
           IF (KYFERM(IOUTSTAT) .NE. KYFERM(INSTATE)) GOTO 50
           IF (ISFERM(IOUTSTAT) .NE. ISFERM(INSTATE)) GOTO 50
          ENDIF

C         AFERMION IS THE SPECTATOR.
          IF ((KAFER(IOUTSTAT) .EQ. KAFER(INSTATE)) .AND.
     $        (KFERM(IOUTSTAT) .NE. KFERM(INSTATE))) THEN
           IF (KXAFER(IOUTSTAT) .NE. KXAFER(INSTATE)) GOTO 50
           IF (KYAFER(IOUTSTAT) .NE. KYAFER(INSTATE)) GOTO 50
           IF (ISAFER(IOUTSTAT) .NE. ISAFER(INSTATE)) GOTO 50
          ENDIF

C         DON'T KNOW WHICH IS SPECTATOR.
          IF ((KAFER(IOUTSTAT) .EQ. KAFER(INSTATE)) .AND.
     $        (KFERM(IOUTSTAT) .EQ. KFERM(INSTATE))) THEN
           IF ((KXAFER(IOUTSTAT) .NE. KXAFER(INSTATE)) .AND.
     $         (KXFERM(IOUTSTAT) .NE. KXFERM(INSTATE))) GOTO 50
```

```
            IF ((KYAFER(IOUTSTAT) .NE. KYAFER(INSTATE)) .AND.
     $          (KYFERM(IOUTSTAT) .NE. KYFERM(INSTATE))) GOTO 50
            IF ((ISAFER(IOUTSTAT) .NE. ISAFER(INSTATE)) .AND.
     $          (ISFERM(IOUTSTAT) .NE. ISFERM(INSTATE))) GOTO 50
           ENDIF
          ENDIF

C-------------------------------------------------------
C     CALCULATE DIAGRAM 16.
C-------------------------------------------------------
C         CHECK IF HAVE CORRECT SPINS.
          IF (ISPHOT(INSTATE) .NE. ISPHOT(IOUTSTAT)) GOTO 400
          IF (ISFERM(INSTATE) .NE. ISFERM(IOUTSTAT)) GOTO 400
          IF (ISPHOT(INSTATE) .EQ. ISFERM(IOUTSTAT)) GOTO 400

C         CHECK IF SPECTATORS MATCH UP.
          IF (ICHARGE .EQ. 0) THEN
           IF (KAFER(INSTATE) .NE. KAFER(IOUTSTAT)) GOTO 400
           IF (KXAFER(INSTATE) .NE. KXAFER(IOUTSTAT)) GOTO 400
           IF (KYAFER(INSTATE) .NE. KYAFER(IOUTSTAT)) GOTO 400
           IF (ISAFER(INSTATE) .NE. ISAFER(IOUTSTAT)) GOTO 400
          ENDIF

C         HAVE INTERACTION.  DETERMINE MATRIX ELEMENT.
          REALH2F(INSTATE) = REALH2F(INSTATE)
     $     + 1.0/(DSQRT(DFLOAT(KPHOT(INSTATE)*KPHOT(IOUTSTAT)))
     $           *DFLOAT(KPHOT(INSTATE) + KFERM(INSTATE)))

C         END OF DIAGRAM 16.
 400      CONTINUE

C-------------------------------------------------------
C     CALCULATE DIAGRAM 18.  ONLY HAVE DIAGRAM 18 IF ICHARGE=0.
C-------------------------------------------------------
          IF (ICHARGE .EQ. -1) GOTO 500
          IF (NOAFERI .EQ. 1) GOTO 500

C         CHECK IF HAVE CORRECT SPINS.
          IF (ISPHOT(INSTATE) .NE. ISPHOT(IOUTSTAT)) GOTO 500
          IF (ISAFER(INSTATE) .NE. ISAFER(IOUTSTAT)) GOTO 500
          IF (ISPHOT(INSTATE) .EQ. ISAFER(IOUTSTAT)) GOTO 500

C         CHECK IF SPECTATORS MATCH UP.
          IF (KFERM(INSTATE) .NE. KFERM(IOUTSTAT)) GOTO 500
          IF (KXFERM(INSTATE) .NE. KXFERM(IOUTSTAT)) GOTO 500
          IF (KYFERM(INSTATE) .NE. KYFERM(IOUTSTAT)) GOTO 500
          IF (ISFERM(INSTATE) .NE. ISFERM(IOUTSTAT)) GOTO 500

C         HAVE INTERACTION.  DETERMINE MATRIX ELEMENT.
          REALH2F(INSTATE) = REALH2F(INSTATE)
     $     + 1.0/(DSQRT(DFLOAT(KPHOT(INSTATE)*KPHOT(IOUTSTAT)))
     $           *DFLOAT(KPHOT(INSTATE) + KAFER(INSTATE)))

C         END OF DIAGRAM 18.
 500      CONTINUE
```

```
          IF (REALH2F(INSTATE) .NE. 0.0D0) THEN
           REALH2F(INSTATE) = REALH2F(INSTATE)/ALPERP2

C************DIAGNOSTICS
C         WRITE (16,940) INSTATE,IOUTSTAT,REALH2F(INSTATE)
C940      FORMAT (' INSTATE IOUTSTAT REALH2F =',2I7,2F11.4)
C************

          FACT = 2.0D0
          IF (INSTATE .EQ. IOUTSTAT) FACT = 1.0D0
          REMSQH2F = REMSQH2F + FACT*REALH2F(INSTATE)
     $               *ALPHAG*( REALWF(IOUTSTAT)*REALWF(INSTATE)
     $                        +AIMWF(IOUTSTAT)*AIMWF(INSTATE))
         ENDIF

C       FINISHED WITH THIS FOCK STATE.  GO TO NEXT FOCK STATE.
 50     CONTINUE
 40    CONTINUE
      ENDIF

      RETURN
      END


      SUBROUTINE HSELFNR(REMSQHSL,REMSQHST,REMSQHS2,REMSQHS3,
     $                   NSIZE,NSTATEOP,REALWF,AIMWF,
     $                   ALPHAG,KPLUSTOT,IBC,ICHARGE,
     $                   ALAMBDA,ALPERP,EPSILON,
     $                   PHOTMASS,FERMMASS,RPHOMASS,RFERMASS,NOAFERI,
     $                   IFERMUP,IUVFERM,NPHOT,NFERM,NAFER,
     $                   KPHOT,KXPHOT,KYPHOT,
     $                   KFERM,KXFERM,KYFERM,
     $                   KAFER,KXAFER,KYAFER)
C-------------------------------------------------------
C     THIS ROUTINE RETURNS THE VALUE OF THE HAMILTONIAN
C     BETWEEN THE INPUTTED VARIATIONAL STATES.
C
C
C     OUTPUT VARIABLES:
C     REMSQHSL  CONTRIB TO MATRIX ELEMENT FROM LONG. PART OF SEFERM1.
C
C     REMSQHST  CONTRIB TO MATRIX ELEMENT FROM TRANS. PART OF SEFERM1.
C
C     REMSQHS2  CONTRIBUTION TO MATRIX ELEMENT FROM SEFERM2.
C
C     REMSQHS3  CONTRIBUTION TO MATRIX ELEMENT FROM SEFERM3.
C
C
C     INPUT VARIABLES:
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.
C
C     NSTATEOP  NUMBER OF FOCK STATES WITH NO PHOTONS.
C
```

```
C     REALWF    REAL PART OF FOCK STATE WAVE FUNCTIONS.
C
C     AIMWF     IMAGINARY PART OF FOCK STATE WAVE FUNCTIONS.
C
C     ALPHAG    VALUE OF COUPLING CONSTANT (=G**2/4PI).
C
C     KPLUSTOT  TOTAL KPLUS OF INCOMING, OUTGOING STATES.
C
C     IBC       ONLY HAVE EVEN FERMION KPLUS IF IBC=2,
C               ODD FERMION KPLUS IF IBC=1.
C
C     ICHARGE   TOTAL CHARGE.
C
C     ALAMBDA   VALUE OF CUT-OFF MASS.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     EPSILON   MINIMUM PHOTON INVMASS**2.
C
C     PHOTMASS  PHOTON MASS IN LAGRANGIAN.
C
C     FERMMASS  FERMION MASS IN LAGRANGIAN.
C
C     RPHOMASS  PHOTON MASS TO BE USED IN COVARIANT CUT-OFF.
C
C     RFERMASS  FERMION MASS TO BE USED IN COVARIANT CUT-OFF.
C
C     NOAFERI   NO INTERACTIONS WITH ANTI-FERMION.
C
C     IFERMUP   FERMIONS, ANTI-FERMIONS ALL HAVE SPIN UP.
C
C     IUVFERM   UV CUT-OFF ONLY APPLIED TO FERMIONS, ANTI-FERMIONS.
C
C     NPHOT,    # PHOTONS, FERMIONS, ANTI-FERMIONS IN FOCK STATES.
C     NFERM,    ARRAYS OF DIMENSION NSIZE.
C     NAFER
C
C     KPHOT,    KPLUS, KX, KY OF PHOTON IN FOCK STATES.
C     KXPHOT,   ARRAYS OF DIMENSION NSIZE.
C     KYPHOT
C
C     KFERM,    KPLUS, KX, KY OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM
C
C     KAFER,    KPLUS, KX, KY OF ANTI-FERMION IN FOCK STATES.
C     KXAFER,   ARRAYS OF DIMENSION NSIZE.
C     KYAFER
C
C
C     USAGE NOTES:
C     1) THIS ROUTINE ASSUMES THAT THE TOTAL KPERP OF THE INCOMING
C        AND OUTGOING FOCK STATES IS ZERO.  THIS IS NECESSARY IN THE
C        CALCULATION OF INVARIANT MASSES.
C     2) FERMION CHARGE IS ASSUMED TO BE -1.
```

```
C     3) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     4) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-----------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(NSIZE),AIMWF(NSIZE),
     $  NPHOT(NSIZE),NFERM(NSIZE),NAFER(NSIZE),
     $  KPHOT(NSIZE),KXPHOT(NSIZE),KYPHOT(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE),
     $  KAFER(NSIZE),KXAFER(NSIZE),KYAFER(NSIZE)

      SMALL = 1.0D-13
      PI = 3.141592653589793D0

      ALPHA = (ALAMBDA*ALPERP/PI)**2/DFLOAT(KPLUSTOT)
      ALPHA2 = (ALPERP/PI)**2*(EPSILON/DFLOAT(KPLUSTOT))
      BETAPHOT = (PHOTMASS*ALPERP/PI)**2
      BETAFERM = (FERMMASS*ALPERP/PI)**2
      BETARPHO = (RPHOMASS*ALPERP/PI)**2
      BETARFER = (RFERMASS*ALPERP/PI)**2

C-----------------------------------------------------------
C     INITIALIZE TO ZERO.
C-----------------------------------------------------------
      REMSQHSL = 0.0D0
      REMSQHST = 0.0D0
      REMSQHS2 = 0.0D0
      REMSQHS3 = 0.0D0

C-----------------------------------------------------------
C     ONLY HAVE SELF-ENERGY CONTRIBUTION IF NO PHOTONS IN FOCK STATE.
C     LOOP OVER DIAGONAL MATRIX ELEMENTS WITH NO PHOTONS.
C-----------------------------------------------------------
      DO 10 ISTATE = 1,NSTATEOP
       IF ((REALWF(ISTATE) .NE. 0.0D0) .OR.
     $     ( AIMWF(ISTATE) .NE. 0.0D0)) THEN

C       INITIALIZE TO ZERO.
        REHSF1L = 0.0D0
        REHSF1T = 0.0D0
        REHSF2 = 0.0D0
        REHSF3 = 0.0D0

C-----------------------------------------------------------
C     CALCULATE HSELFNR.
C-----------------------------------------------------------
C-----------------------------------------------------------
C     FIND CONTRIBUTION TO HSELF FROM FERMION
C-----------------------------------------------------------
        INPLUS = KFERM(ISTATE)
        INX = KXFERM(ISTATE)
        INY = KYFERM(ISTATE)
```

```
C       CALCULATE CUT-OFF.
        ALPHA1 = ALPHA
        IF (ICHARGE .EQ. 0) THEN
         AKPERPSQ = DFLOAT(KXAFER(ISTATE)**2 + KYAFER(ISTATE)**2)
         SPECINVM = (AKPERPSQ + BETARFER)/DFLOAT(KAFER(ISTATE))
         ALPHA1 = ALPHA1 - SPECINVM
        ENDIF
        IF (ABS(ALPHA1) .LE. SMALL) ALPHA1 = 0.0D0

C       SUBROUTINE SEFERM RETURNS VALUES OF SELF-ENERGIES.
        CALL SEFERM(SEFERM1L,SEFERM1T,SEFERM2,SEFERM3,IFERMUP,
     $              ALPHAG,ALPERP,IBC,INPLUS,INX,INY,IUVFERM,
     $              ALPHA1,ALPHA2,BETAPHOT,BETAFERM,BETARPHO,BETARFER)
        REHSF1L = REHSF1L + SEFERM1L
        REHSF1T = REHSF1T + SEFERM1T
        REHSF2 = REHSF2 + SEFERM2
        REHSF3 = REHSF3 + SEFERM3

C-------------------------------------------------------
C     FIND CONTRIBUTION TO HSELF FROM ANTI-FERMION.
C     ONLY HAVE THIS CONTRIBUTION IF ICHARGE=0.
C-------------------------------------------------------
        IF ((ICHARGE .EQ. 0) .AND. (NOAFERI .EQ. 0)) THEN
         INPLUS = KAFER(ISTATE)
         INX = KXAFER(ISTATE)
         INY = KYAFER(ISTATE)

C        CALCULATE CUT-OFF.
         AKPERPSQ = DFLOAT(KXFERM(ISTATE)**2 + KYFERM(ISTATE)**2)
         SPECINVM = (AKPERPSQ + BETARFER)/DFLOAT(KFERM(ISTATE))
         ALPHA1 = ALPHA - SPECINVM
         IF (ABS(ALPHA1) .LE. SMALL) ALPHA1 = 0.0D0

C        SUBROUTINE SEFERM RETURNS VALUES OF SELF-ENERGIES.
         CALL SEFERM(SEFERM1L,SEFERM1T,SEFERM2,SEFERM3,IFERMUP,
     $               ALPHAG,ALPERP,IBC,INPLUS,INX,INY,IUVFERM,
     $               ALPHA1,ALPHA2,BETAPHOT,BETAFERM,BETARPHO,BETARFER)
         REHSF1L = REHSF1L + SEFERM1L
         REHSF1T = REHSF1T + SEFERM1T
         REHSF2 = REHSF2 + SEFERM2
         REHSF3 = REHSF3 + SEFERM3
        ENDIF

C************DIAGNOSTICS
C       WRITE (16,910) ISTATE,REHSF1L,REHSF1T,REHSF2,REHSF3
C910    FORMAT (' ISTATE REHSF1L REHSF1T REHSF2 REHSF3 ='/
C    $          I7,4F11.4)
C************

        REMSQHSL = REMSQHSL
     $   +ALPHAG*REHSF1L*(REALWF(ISTATE)**2 + AIMWF(ISTATE)**2)
        REMSQHST = REMSQHST
     $   +ALPHAG*REHSF1T*(REALWF(ISTATE)**2 + AIMWF(ISTATE)**2)
        REMSQHS2 = REMSQHS2
     $   +ALPHAG*REHSF2*(REALWF(ISTATE)**2 + AIMWF(ISTATE)**2)
```

```
        REMSQHS3 = REMSQHS3
     $   +ALPHAG*REHSF3*(REALWF(ISTATE)**2 + AIMWF(ISTATE)**2)

C       FINISHED WITH THIS FOCK STATE.  GO TO NEXT FOCK STATE.
        ENDIF
 10   CONTINUE

      RETURN
      END


      SUBROUTINE SEFERM(SEFERM1L,SEFERM1T,SEFERM2,SEFERM3,IFERMUP,
     $                  ALPHAG,ALPERP,IBC,INPLUS,INX,INY,IUVFERM,
     $                  ALPHA1,ALPHA2,BETAPHOT,BETAFERM,BETARPHO,BETARFER)
C-------------------------------------------------------------
C     THIS SUBROUTINE RETURNS THE VALUE FOR THE FERMION SELF-ENERGY
C     FOR INCOMING (KPLUS,KX,KY) = (INPLUS,INX,INY).
C-------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      PI = 3.14159265358979300
      SMALL = 1.0D-13

C-------------------------------------------------------------
C     INITIALIZE TO ZERO.
C-------------------------------------------------------------
      SEFERM1L = 0.0D0
      SEFERM1T = 0.0D0
      SEFERM2 = 0.0D0
      SEFERM3 = 0.0D0
      ANUM2 = 0.0D0
      DENOM2 = 0.0D0

C-------------------------------------------------------------
C     WORK OUT SELF-ENERGIES.
C-------------------------------------------------------------
      ANPLUS = DFLOAT(INPLUS)

C************DIAGNOSTICS
C       WRITE (16,920) INPLUS,INX,INY,ALPHA1
C920    FORMAT (' '/' NPLUS NX NY ALPHA1 =',3I4,F8.3/
C    $  '  QPLUS  QX  QY  INVMASS'/
C    $  '  ----------------------')
C************

      IF (IBC .EQ. 1) IQPLUSMX = INPLUS - 1
      IF (IBC .EQ. 2) IQPLUSMX = INPLUS - 2
      DO 20 IQPLUS = 2,IQPLUSMX,2
        AQPLUS = DFLOAT(IQPLUS)

C       CONTINUE ONLY IF AINVMASS .LE. ALPHA1.
        AINVMASS = BETARFER/(ANPLUS-AQPLUS)

C************DIAGNOSTICS
```

```
C         WRITE (16,921) IQPLUS,AINVMASS
C921      FORMAT (3X,I3,10X,F8.3)
C************

        IF (AINVMASS .GT. (ALPHA1+SMALL)) GOTO 20

C       FIGURE OUT WHAT LARGEST ALLOWED PHOTON KX, KY IS.
        ARGUMENT = ALPHA1*(ANPLUS-AQPLUS) - BETARFER
        IF (ARGUMENT .LT. SMALL) THEN
          IQPERPMX = 0
         ELSE
          IQPERPMX = INT(DSQRT(ARGUMENT) + SMALL)
        ENDIF
        IQXMIN = INX - IQPERPMX
        IQYMIN = INY - IQPERPMX
        IQXMAX = INX + IQPERPMX
        IQYMAX = INY + IQPERPMX

        DO 30 IQX = IQXMIN,IQXMAX
        DO 30 IQY = IQYMIN,IQYMAX

C         REMOVE THIS POINT IF PHOTON INVMASS**2 .LT. EPSILON.
          IF ((DFLOAT(IQX**2+IQY**2) + BETARPHO)/AQPLUS .LT.
     $        ALPHA2-SMALL) GOTO 30

C         KEEP THIS TERM IN SUM FOR SE IF INVMASS**2 .LE. LAMBDA**.
          IF (IUVFERM .EQ. 0)
     $      AINVMASS = (DFLOAT(IQX**2 + IQY**2) + BETARPHO)/AQPLUS
     $               + (DFLOAT((INX-IQX)**2 + (INY-IQY)**2) + BETARFER)
     $                 /(ANPLUS-AQPLUS)
          IF (IUVFERM .EQ. 1)
     $      AINVMASS = (DFLOAT((INX-IQX)**2 + (INY-IQY)**2) + BETARFER)
     $                 /(ANPLUS-AQPLUS)

C************DIAGNOSTICS
C           WRITE (16,922) IQX,IQY,AINVMASS
C922      FORMAT (8X,I3,1X,I3,1X,F8.3)
C************

          IF (AINVMASS .GT. (ALPHA1+SMALL)) GOTO 30

C         ADD CONTRIBUTIONS TO SEFERM1L,SEFERM1T,SEFERM2,SEFERM3.
          PERPSQAR = (DFLOAT(IQX) - (AQPLUS/ANPLUS)*DFLOAT(INX))**2
     $             + (DFLOAT(IQY) - (AQPLUS/ANPLUS)*DFLOAT(INY))**2
          DENOM = ANPLUS**2*PERPSQAR + AQPLUS**2*BETAFERM
     $          + ANPLUS*(ANPLUS-AQPLUS)*BETAPHOT
          ANUMT = (ANPLUS**2*PERPSQAR)/
     $            (2.0D0*ANPLUS*(ANPLUS-AQPLUS))
     $          + (ANPLUS**2*PERPSQAR)/(AQPLUS**2)
          SEFERM1T = SEFERM1T + ANUMT/DENOM
          IF (IFERMUP .EQ. 0) THEN
            ANUML = (AQPLUS**2*BETAFERM)/
     $              (2.0D0*ANPLUS*(ANPLUS-AQPLUS))
            SEFERM1L = SEFERM1L + ANUML/DENOM
            ANUM2 = ANUM2 + AQPLUS/DENOM
```

```
            DENOM2 = DENOM2 + (ANPLUS - AQPLUS)/DENOM
            SEFERM3 = SEFERM3 + AQPLUS/DENOM
          ENDIF

 30     CONTINUE
 20   CONTINUE

      SEFERM1T = (2.0/ALPERP**2)*SEFERM1T
      IF (IFERMUP .EQ. 0) THEN
       SEFERM1L = (2.0/ALPERP**2)*SEFERM1L
       ANUM2 = (ALPHAG/PI**4)*ANUM2**2
       DENOM2 = 1 + (ALPHAG/PI**2)*DENOM2
       SEFERM2 = -(BETAFERM*PI**2)/(ANPLUS*ALPERP**2)*(ANUM2/DENOM2)
       SEFERM3 = -(ALPHAG*BETAFERM*SEFERM3**2)/
     $            (ANPLUS*ALPERP**2*PI**2)
      ENDIF

C************DIAGNOSTICS
C     WRITE (16,923) SEFERM1L,SEFERM1T,SEFERM2,SEFERM3
C923  FORMAT
C    $  (' SEFERM1L SEFERM1T SEFERM2 SEFERM3 =',4F8.4)
C************

      RETURN
      END


      SUBROUTINE PRINTOUT(WAVEFCN,TIME1,TIME2,TIME3,TIME4,TIME5,TIME6,
     $                    KPLUSTOT,ICHARGE,IBC,NSTATES,NSTATEOP,NSIZE,
     $                    KPLUSMAX,ALPHAG,ALAMBDA,ALPERP,EPSILON,
     $                    PARA1,PARA2,PARA3,PARA4,PARA5,
     $                    PHOTMASS,RPHOMASS,FERMMASS,RFERMASS,
     $                    REMSQHO,REMSQH1L,REMSQH1T,REMSQH2P,REMSQH2F,
     $                    REMSQHSL,REMSQHST,REMSQHS2,REMSQHS3,
     $                    REALWF,AIMWF,NPHOT,NFERM,NAFER,
     $                    KPHOT,KXPHOT,KYPHOT,ISPHOT,
     $                    KFERM,KXFERM,KYFERM,ISFERM,
     $                    KAFER,KXAFER,KYAFER,ISAFER)
C-------------------------------------------------------------
C     THIS SUBROUTINE PRINTS OUT RESULTS OF PROGRAM QEDVAR.
C-------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION WAVEFCN(KPLUSMAX)

      DIMENSION REALWF(NSIZE),AIMWF(NSIZE),
     $  NPHOT(NSIZE),NFERM(NSIZE),NAFER(NSIZE),
     $  KPHOT(NSIZE),KXPHOT(NSIZE),KYPHOT(NSIZE),ISPHOT(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE),ISFERM(NSIZE),
     $  KAFER(NSIZE),KXAFER(NSIZE),KYAFER(NSIZE),ISAFER(NSIZE)

      LOGICAL ODDKPLUS

      PI = 3.141592653589793D0
```

```
C************DIAGNOSTICS: RESULTS FROM QEDVAR
      RPERP = ALAMBDA*ALPERP/PI
      XALPERP = ALPERP*ALPHAG

      WRITE (15,910)
     $  KPLUSTOT,ICHARGE,IBC,
     $  ALPHAG,ALAMBDA,ALPERP,XALPERP,RPERP,EPSILON,
     $  PHOTMASS,RPHOMASS,FERMMASS,RFERMASS,
     $  PARA1,PARA2,PARA3,PARA4,PARA5,NSTATEOP,NSTATES
      WRITE (15,911)
     $  TIME1,TIME2,TIME3,TIME4,TIME5,TIME6,
     $  TIME1+TIME2+TIME3+TIME4+TIME5+TIME6
      WRITE (15,912)
     $  REMSQHO,REMSQH1L,REMSQH1T,REMSQH2P,REMSQH2F,
     $  REMSQHSL,REMSQHST,REMSQHS2,REMSQHS3,
     $  .5*REMSQH1L+REMSQHSL,.5*REMSQH1T+REMSQHST,REMSQH2P,
     $  .5*REMSQH1L+.5*REMSQH1T+REMSQHSL+REMSQHST+REMSQH2P
      IF (ICHARGE .EQ. 0) THEN
        WRITE (15,913) REMSQHO+.5*REMSQH1L+.5*REMSQH1T-4.0D0
       ELSEIF (ICHARGE .EQ. -1) THEN
        WRITE (15,914) REMSQHO+.5*REMSQH1L+.5*REMSQH1T-1.0D0
      ENDIF
      WRITE (15,915)
     $  REMSQHO+REMSQH1L+REMSQH1T+REMSQH2P+REMSQHSL+REMSQHST,
     $  REMSQHO+REMSQH1L+REMSQH1T+REMSQH2P+REMSQH2F
     $                          +REMSQHSL+REMSQHST+REMSQHS2,
     $  REMSQHO+REMSQH1L+REMSQH1T+REMSQH2P+REMSQH2F
     $                          +REMSQHSL+REMSQHST+REMSQHS3
 910  FORMAT (' INPUT (MASSES ARE IN UNITS OF ELECMASS):'/
     $        ' KPLUSTOT ICHARGE IBC =',3I4/
     $        ' ALPHAG   =',F11.4/
     $        ' ALAMBDA  =',F11.4/
     $        ' ALPERP   =',F11.4,'*1/ELECMASS =',F8.4,
     $                          '*BOHR  (RPERP =',F8.4,')'/
     $        ' EPSILON  =',F11.4/
     $        ' PHOTMASS =',F11.4/
     $        ' RPHOMASS =',F11.4/
     $        ' FERMMASS =',F11.4/
     $        ' RFERMASS =',F11.4/
     $        ' VAR PARAMETERS =',5F8.4/' '/
     $        ' # OF FOCK STATES WITH NO PHOTONS =',I9/
     $        ' # OF FOCK STATES                 =',I9/' ')
 911  FORMAT (' CPU TIME TO FIND FOCK STATES     =',F8.2,' SEC'/
     $        ' CPU TIME TO WORK OUT VAR WF      =',F8.2,' SEC'/
     $        ' CPU TIME TO FIND HO    MATRIX EL =',F8.2,' SEC'/
     $        ' CPU TIME TO FIND H1    MATRIX EL =',F8.2,' SEC'/
     $        ' CPU TIME TO FIND H2    MATRIX EL =',F8.2,' SEC'/
     $        ' CPU TIME TO FIND HSELF MATRIX EL =',F8.2,' SEC'/
     $        ' TOTAL CPU TIME USED              =',F8.2,' SEC'/' ')
 912  FORMAT (' CONTRIBUTION TO M**2 FROM     HO =',F15.10/
     $        '                  LONG VERTEX = H1L =',F15.10/
     $        '                 TRANS VERTEX = H1T =',F15.10/
     $        '                            INSPHOT =',F15.10/
     $        '                            INSFERM =',F15.10/
```

```
     $        '   LONG PART OF 1 LOOP SE = SEF1L =',F15.10/
     $        '  TRANS PART OF 1 LOOP SE = SEF1T =',F15.10/
     $        '        N CHAINED INST SE =  SEF2 =',F15.10/
     $        '        2 CHAINED INST SE =  SEF3 =',F15.10/' '/
     $        '          LONG PHOT = .5H1L+SEF1L =',F15.10/
     $        '         TRANS PHOT = .5H1T+SEF1T =',F15.10/
     $        '          INST PHOT =     INSPHOT =',F15.10/
     $        '                  PE = L+T+I PHOT =',F15.10)
 913  FORMAT ('                  KE =  HO+.5H1-4 =',F15.10)
 914  FORMAT ('                  KE =  HO+.5H1-1 =',F15.10)
 915  FORMAT ('       SUM+4 = HO+H1+INSPHOT+SEF1 =',F15.10/' '/
     $        '               SUM+4+INSFERM+SEF2 =',F15.10/
     $        '               SUM+4+INSFERM+SEF3 =',F15.10)
C************

C-------------------------------------------------------
C     DETERMINE FRACTION OF EIGENSTATE THAT CONSISTS OF
C     VARIOUS DIFFERENT FOCK STATES.
C-------------------------------------------------------
C-------------------------------------------------------
C     PRINTOUT FOR ICHARGE = 0.  THE 2 FOCK STATES ANALYZED
C     ARE 1 PAIR AND 0,1 PHOTONS.
C-------------------------------------------------------
      IF (ICHARGE .EQ. 0) THEN

C       INTIALIZE AFOCK1,AFOCK2
        AFOCK1 = 0.0D0
        AFOCK2 = 0.0D0

        DO 20 ISTATE = 1,NSTATES
          IF ((NPHOT(ISTATE) .EQ. 0) .AND.
     $        (NFERM(ISTATE) .EQ. 1)) THEN
            AFOCK1 = AFOCK1 + REALWF(ISTATE)**2 + AIMWF(ISTATE)**2
          ENDIF

          IF ((NPHOT(ISTATE) .EQ. 1) .AND.
     $        (NFERM(ISTATE) .EQ. 1)) THEN
            AFOCK2 = AFOCK2 + REALWF(ISTATE)**2 + AIMWF(ISTATE)**2
          ENDIF
 20     CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
        WRITE (15,920) 100.*AFOCK1,100.*AFOCK2
 920    FORMAT (' '/' '/' '/' FOCK STATE DECOMPOSITION:',
     $          F6.2,'% 1 PAIR,0 PHOT',F6.2,'% 1 PAIR,1 PHOT')
C************

      ENDIF

C-------------------------------------------------------
C     PRINTOUT FOR ICHARGE = -1.  THE 2 FOCK STATES ANALYZED
C     ARE 1 FERMION AND 0,1 PHOTONS.
C-------------------------------------------------------
      IF (ICHARGE .EQ. -1) THEN
```

```
C       INTIALIZE AFOCK1,AFOCK2
        AFOCK1 = 0.0D0
        AFOCK2 = 0.0D0

        DO 30 ISTATE = 1,NSTATES
          IF ((NPHOT(ISTATE) .EQ. 0) .AND.
     $        (NFERM(ISTATE) .EQ. 1)) THEN
            AFOCK1 = AFOCK1 + REALWF(ISTATE)**2 + AIMWF(ISTATE)**2
          ENDIF

          IF ((NPHOT(ISTATE) .EQ. 1) .AND.
     $        (NFERM(ISTATE) .EQ. 1)) THEN
            AFOCK2 = AFOCK2 + REALWF(ISTATE)**2 + AIMWF(ISTATE)**2
          ENDIF
 30     CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
        WRITE (15,930) 100.*AFOCK1,100.*AFOCK2
 930    FORMAT (' '/' '/' '/' FOCK STATE DECOMPOSITION:',
     $          F6.2,'% 1 FERM,0 PHOT',F6.2,'% 1 FERM,1 PHOT')
C************

      ENDIF

C-------------------------------------------------------
C     DETERMINE STRUCTURE FUNCTIONS (PROBABILITY OF
C     FINDING FERMION WITH PLUS MOMENTUM FRACTION X).
C-------------------------------------------------------
C     FIGURE OUT WHAT VALUES OF FERMION KPLUS TO RUN OVER
      KBIG = 0
      DO 40 ISTATE = 1,NSTATES
        K = KFERM(ISTATE)
        IF (K .GT. KBIG) KBIG = K
 40   CONTINUE
      KSMALL = KPLUSTOT
      DO 41 ISTATE = 1,NSTATES
        K = KFERM(ISTATE)
        IF (K .LT. KSMALL) KSMALL = K
 41   CONTINUE

C     INITIALIZE STRUCTURE FUNCTION (WAVEFCN) TO ZERO
      DO 42 KPLUS = KSMALL,KBIG,2
        WAVEFCN(KPLUS) = 0.0D0
 42   CONTINUE

C     LOOP OVER FOCK STATE COMPONENTS
      DO 43 ISTATE = 1,NSTATES
        WAVEFCN(KFERM(ISTATE))
     $    = WAVEFCN(KFERM(ISTATE))
     $    + REALWF(ISTATE)**2 + AIMWF(ISTATE)**2
 43   CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
C     NOTE: IMAGEN PRINTS UP TO 80 CHARACTERS ACROSS.
C           NEED TO DECLARE LRECL 84 IN FILEDEF TO DO SO.
```

```
      WRITE (15,940)
     $  (INT(.5+10000.0*(DFLOAT(KPLUS)/DFLOAT(KPLUSTOT))),
     $   KPLUS=KSMALL,KBIG,2)
 940  FORMAT (' '/' '/' '/
     $' STRUCTURE FUNCTION:'/' '/
     $' NOTES: 1) VALUES SHOULD BE MULTIPLIED BY 1/10000 (***=10000)'/
     $' '/'      X=',14(1X,I4))
      WRITE (15,941)
 941  FORMAT (
     $'     ------------------------------------------------------------',
     $'-------------')
      WRITE (15,942)
     $  (INT(.5+10000.0*WAVEFCN(KPLUS)),KPLUS=KSMALL,KBIG,2)
 942  FORMAT (8X,14(1X,I4))
C************

C-------------------------------------------------------
C     PLOT WAVE FUNCTION.
C-------------------------------------------------------
C     FIGURE OUT WHAT VALUES FOR FERMION KX TO RUN OVER
      KXBIG = 0
      DO 50 ISTATE = 1,NSTATES
        KX = KXFERM(ISTATE)
        IF (KX .GT. KXBIG) KXBIG = KX
 50   CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
C     NOTE: IMAGEN PRINTS UP TO 80 CHARACTERS ACROSS.
C           NEED TO DECLARE LRECL 84 IN FILEDEF TO DO SO.
      WRITE (15,950)
 950  FORMAT (' '/' '/
     $' WAVE FUNCTION SQUARED AT KY=0:'/' '/
     $' NOTES: 1) VALUES SHOULD BE MULTIPLIED BY 1/10000 (***=10000)'/
     $'        2) KX IS IN UNITS OF ELECTRON MASS'/' ')
C************

C     LOOP OVER VALUES OF KX
      DO 53 KX = KXBIG,-KXBIG,-1

C       INITIALIZE WAVE FUNCTION TO ZERO
        DO 54 KPLUS = KSMALL,KBIG,2
          WAVEFCN(KPLUS) = 0.0D0
 54     CONTINUE

C       LOOP OVER FOCK STATE COMPONENTS
        DO 56 ISTATE = 1,NSTATES
          IF ((KXFERM(ISTATE) .EQ. KX) .AND.
     $        (KYFERM(ISTATE) .EQ. 0) ) THEN
            WAVEFCN(KFERM(ISTATE))
     $      = WAVEFCN(KFERM(ISTATE))
     $      + REALWF(ISTATE)**2 + AIMWF(ISTATE)**2
          ENDIF
 56     CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
```

```
        WRITE (15,952) INT(10000.0*DFLOAT(KX)*PI/ALPERP),
     $    (INT(.5+10000.0*WAVEFCN(KPLUS)),KPLUS=KSMALL,KBIG,2)
 952    FORMAT (1X,I5,' |',14(1X,I4))
C************

 53   CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
      WRITE (15,956)
     $  (INT(.5+10000.0*(DFLOAT(KPLUS)/DFLOAT(KPLUSTOT))),
     $   KPLUS=KSMALL,KBIG,2)
 956  FORMAT (
     $'       |-----------------------------------------------------',
     $'-------------'/
     $'   KX  X',14(1X,I4))
C************

      RETURN
      END
```

COULOMBI

```
C-----------------------------------------------------------
C     THIS ROUTINE HANDLES INPUT, VARIABLE PARMETERS FOR COULOMB.
C     IT GENERATES INPUT FILES FOR THE ROUTINE COULOMB.
C     INPUT PARAMETERS ARE READ FROM THE FILE COULOMB DATA.
C     MASSES ARE IN UNITS OF ELECMASS.
C     LENGTHS ARE IN UNITS OF 1/ELECMASS.
C
C     IBC = 1 MEANS KEEP ONLY ODD FERMION KPLUS
C         = 2 MEANS KEEP ONLY EVEN FERMION KPLUS
C
C     FERMIONS ASSUMED TO HAVE CHARGE -1.
C
C     CODES HAVE BEEN VECTORIZED ON AN IBM 3090 FORTRAN COMPILER.
C-----------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

C-----------------------------------------------------------
C     OPEN FILES.
C-----------------------------------------------------------
      OPEN (UNIT=13,FILE='qed.data',STATUS='UNKNOWN')
      OPEN (UNIT=20,FILE='file20.file',STATUS='UNKNOWN')

      PI = 3.141592653589793D0

C-----------------------------------------------------------
C     READ INPUT DATA.
C     NOTE: COVARIANT CUT-OFF SCHEME PRESENTLY ASSUMES KXTOT,KYTOT = 0.
C           CAN GET OTHER VALUES FOR KXTOT,KYTOT BY BOOSTING
C           (SEE NOTES ON BOUND STATES).
C     MASSES ARE IN UNITS OF ELECMASS.
C     LENGTHS ARE IN UNITS OF 1/ELECMASS.
C-----------------------------------------------------------
      READ (13,*) KPLUSBEG
      READ (13,*) KPLUSEND
      READ (13,*) KXTOT
      READ (13,*) KYTOT
      READ (13,*) ICHARGE
      READ (13,*) IBC
      READ (13,*) ALPHAG
      READ (13,*) ALAMBBEG
      READ (13,*) ALAMBEND
      READ (13,*) ALPERBEG
      READ (13,*) ALPEREND
      READ (13,*) EPSILBEG
      READ (13,*) EPSILEND
      READ (13,*) PARA1
      READ (13,*) PARA2
      READ (13,*) PARA3
      READ (13,*) PARA4
      READ (13,*) PARA5
      READ (13,*) PHOTMASS
```

```
      READ (13,*) RPHOMASS
      READ (13,*) FERMMASS
      READ (13,*) RFERMASS
      READ (13,*) IUVFERM
      READ (13,*) IPRIWF
      READ (13,*) IPRIOUT

C-------------------------------------------------------
C     CHECK IF VALUES OF INPUT DATA ARE O.K.
C-------------------------------------------------------
      IF (ICHARGE .NE. 0) THEN
        WRITE (6,919)
 919    FORMAT (' MSG FROM COULOMBI: ICHARGE MUST BE 0.')
        STOP
      ENDIF

      IF (KPLUSBEG .LE. 0) THEN
        WRITE (6,920)
 920    FORMAT (' MSG FROM COULOMBI: KPLUSBEG MUST BE .GT. 0')
        STOP
      ENDIF

      IF (KPLUSEND .LE. 0) THEN
        WRITE (6,921)
 921    FORMAT (' MSG FROM COULOMBI: KPLUSEND MUST BE .GT. 0')
        STOP
      ENDIF

      IF ((IBC .NE. 1) .AND.
     $    (IBC .NE. 2)) THEN
        WRITE (6,922)
 922    FORMAT (' MSG FROM COULOMBI: IBC MUST BE 1 OR 2')
        STOP
      ENDIF

      IF ((IBC .EQ. 2) .AND. (MOD(KPLUSBEG,2) .EQ. 1)) THEN
        WRITE (6,924)
 924    FORMAT(' MSG FROM COULOMBI: KPLUSBEG MUST BE EVEN IF IBC = 2')
        STOP
      ENDIF

      IF ((IBC .EQ. 2) .AND. (MOD(KPLUSEND,2) .EQ. 1)) THEN
        WRITE (6,925)
 925    FORMAT(' MSG FROM COULOMBI: KPLUSEND MUST BE EVEN IF IBC = 2')
        STOP
      ENDIF

      IF ((ICHARGE .EQ. 0) .AND. (MOD(KPLUSBEG,2) .EQ. 1)) THEN
        WRITE (6,926)
 926    FORMAT (' MSG FROM COULOMBI:',
     $          ' KPLUSBEG MUST BE EVEN IF ICHARGE = 0')
        STOP
      ENDIF

      IF ((ICHARGE .EQ. 0) .AND. (MOD(KPLUSEND,2) .EQ. 1)) THEN
```

```
        WRITE (6,927)
 927    FORMAT (' MSG FROM COULOMBI:',
     $          ' KPLUSEND MUST BE EVEN IF ICHARGE = 0')
        STOP
       ENDIF

       IF ((KXTOT .NE. 0) .OR.
     $    (KYTOT .NE. 0)) THEN
        WRITE (6,930)
 930    FORMAT (' MSG FROM COULOMBI: KXTOT AND KYTOT MUST BE'/
     $          ' EQUAL TO ZERO.  OTHER VALUES CAN BE OBTAINED'/
     $          ' BY LORENTZ BOOSTING.')
        STOP
       ENDIF

       IF (EPSILON .LT. 0.0D0) THEN
        WRITE (6,931)
 931    FORMAT (' MSG FROM COULOMBI: EPSIOON MUST BE .GE. 0.0D0')
        STOP
       ENDIF

       IF ((IUVFERM .NE. 0) .AND. (IUVFERM .NE. 1)) THEN
        WRITE (6,932)
 932    FORMAT (' MSG FROM COULOMBI: IUVFERM MUST BE 0 OR 1')
        STOP
       ENDIF

C------------------------------------------------------
C      GENERATE PARAMETERS FILE.
C------------------------------------------------------
       WRITE (20,954) KPLUSBEG,KPLUSEND,KXTOT,KYTOT,ICHARGE,IBC
       WRITE (20,955) ALPHAG
       WRITE (20,955) ALAMBBEG,ALAMBEND
       WRITE (20,955) ALPERBEG,ALPEREND
       WRITE (20,955) EPSILBEG,EPSILEND
       WRITE (20,955) PARA1
       WRITE (20,955) PARA2
       WRITE (20,955) PARA3
       WRITE (20,955) PARA4
       WRITE (20,955) PARA5
       WRITE (20,955) PHOTMASS
       WRITE (20,955) RPHOMASS
       WRITE (20,955) FERMMASS
       WRITE (20,955) RFERMASS
       WRITE (20,954) IUVFERM,IPRIWF,IPRIOUT
 954   FORMAT (6I8)
 955   FORMAT (2D30.22)

       STOP
       END
```

COULOMB

```
C-----------------------------------------------------------
C     THIS ROUTINE EVALUATES THE HAMILTONIAN MATRIX BETWEEN
C     VARIATIONAL STATES.
C
C     P_MINUS IS DEFINED TO BE L/PI*HAMILTONIAN.
C
C     MASSES ARE IN UNITS OF ELECMASS.
C     LENGTHS ARE IN UNITS OF 1/ELECMASS.
C
C     FERMIONS ASSUMED TO HAVE CHARGE -1.
C-----------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(222222),
     $  KFERM(222222),KXFERM(222222),KYFERM(222222)

C-----------------------------------------------------------
C     FOLLOWING ARRAY NEEDED IN SUBROUTINE PRINTC.
C-----------------------------------------------------------
      DIMENSION WAVEFCN(4444)

      LOGICAL TOOMANY

C-----------------------------------------------------------
C     OPEN FILES.
C-----------------------------------------------------------
      OPEN (UNIT=14,FILE='statesc.output',STATUS='UNKNOWN')
      OPEN (UNIT=15,FILE='coulomb.output',STATUS='UNKNOWN')
      OPEN (UNIT=16,FILE='coulomb.diagnose',STATUS='UNKNOWN')
      OPEN (UNIT=20,FILE='file20.file',STATUS='UNKNOWN')
      OPEN (UNIT=24,FILE='ke.file',STATUS='UNKNOWN')
      OPEN (UNIT=25,FILE='pe.file',STATUS='UNKNOWN')
      OPEN (UNIT=26,FILE='energy.file',STATUS='UNKNOWN')

C-----------------------------------------------------------
C     DIM(REALWF,KFERM,...) = NSIZE.  NSIZE SHOULD BE CHOSEN
C     .GE. THE NUMBER OF FOCK STATES (NSTATES).
C     DIM(WAVEFCN) = KPLUSMAX.  KPLUSMAX SHOULD BE CHOSEN
C     .GE. KPLUSTOT.
C-----------------------------------------------------------
      NSIZE = 222222
      KPLUSMAX = 4444
      IF (KPLUSEND .GT. KPLUSMAX) THEN
        WRITE (15,900) KPLUSMAX
 900    FORMAT(' KPLUSTOT .GT. KPLUSMAX =',I5/
     $         ' RE-COMPILE QEDVAR WITH LARGER VALUE OF KPLUSMAX.')
        STOP
      ENDIF

      PI = 3.14159265358979 3D0
```

```
C-------------------------------------------------------
C     READ PARAMETERS FILE (FILE20).
C-------------------------------------------------------
      READ (20,904) KPLUSBEG,KPLUSEND,KXTOT,KYTOT,ICHARGE,IBC
      READ (20,905) ALPHAG
      READ (20,905) ALAMBBEG,ALAMBEND
      READ (20,905) ALPERBEG,ALPEREND
      READ (20,905) EPSILBEG,EPSILEND
      READ (20,905) PARA1
      READ (20,905) PARA2
      READ (20,905) PARA3
      READ (20,905) PARA4
      READ (20,905) PARA5
      READ (20,905) PHOTMASS
      READ (20,905) RPHOMASS
      READ (20,905) FERMMASS
      READ (20,905) RFERMASS
      READ (20,904) IUVFERM,IPRIWF,IPRIOUT
 904  FORMAT (6I8)
 905  FORMAT (2D30.22)

      WRITE (24,914) ALPHAG
 914  FORMAT (' ALPHAG =',F9.4/
     $    ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON        KE    '/
     $    ' ----------------------------------------------------------')
      WRITE (25,915) ALPHAG
 915  FORMAT (' ALPHAG =',F9.4/
     $    ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON        PE    '/
     $    ' ----------------------------------------------------------')
      WRITE (26,916) ALPHAG
 916  FORMAT (' ALPHAG =',F9.4/
     $    ' KPLUSTOT    ALAMBDA    ALPERP     EPSILON      ENERGY  '/
     $    ' ----------------------------------------------------------')

      DO 100 KPLUSTOT = KPLUSBEG,KPLUSEND,8
      DO 100 ALAMBDA = ALAMBBEG,ALAMBEND,.05
      DO 100 ALPERP = ALPERBEG,ALPEREND, 4.0
      DO 100 EPSILON = EPSILBEG,EPSILEND,.01

C-------------------------------------------------------
C     DETERMINE START CPU TIME.
C-------------------------------------------------------
        CALL VTTIME(IVIRTIME,ITOTTIME)
        START = DFLOAT(ITOTTIME)/100.0D0

C-------------------------------------------------------
C     CALL SUBROUTINE STATESC TO GENERATE STATES CONSISTENT
C     WITH K, ALPERP, ALAMBDA.
C-------------------------------------------------------
        CALL STATESC(NSIZE,KPLUSTOT,IBC,
     $               ALAMBDA,ALPERP,
     $               RFERMASS,
     $               NSTATES,TOOMANY,
     $               KFERM,KXFERM,KYFERM)
        IF (TOOMANY) THEN
```

```
          WRITE (15,918) NSIZE
 918      FORMAT (' NUMBER OF STATES GENERATED BY SUBROUTINE',
     $            ' STATESC .GT. NSIZE =',I5/
     $            ' RE-COMPILE COULOMB WITH LARGER VALUE OF NSIZE')
          STOP
        ENDIF

        CALL VTTIME(IVIRTIME,ITOTTIME)
        TIME1 = DFLOAT(ITOTTIME)/100.0D0-START

C-------------------------------------------------------
C     CALL SUBROUTINE WF TO WORK OUT VARIATIONAL WAVE-FUNCTION.
C-------------------------------------------------------
        CALL WF(REALWF,NSIZE,NSTATES,
     $          ALPHAG,KPLUSTOT,ALPERP,
     $          FERMMASS,
     $          PARA1,PARA2,PARA3,PARA4,PARA5,
     $          KFERM,KXFERM,KYFERM)

C************DIAGNOSTICS: PRINT OUT RESULTS FROM WF
        RPERP = ALAMBDA*ALPERP/PI
        XALPERP = ALPERP*ALPHAG

        WRITE (14,920) KPLUSTOT,ICHARGE,IBC,
     $    ALPHAG,ALAMBDA,ALPERP,XALPERP,RPERP,EPSILON,PHOTMASS,RPHOMASS,
     $    FERMMASS,RFERMASS,NSTATES
 920    FORMAT (' INPUT (MASSES ARE IN UNITS OF ELECMASS):'/
     $          ' KPLUSTOT ICHARGE IBC =',3I4/
     $          ' ALPHAG   =',F11.4/
     $          ' ALAMBDA  =',F11.4/
     $          ' ALPERP   =',F11.4,'*1/ELECMASS =',F8.4,
     $                        '*BOHR  (RPERP =',F8.4,')'/
     $          ' EPSILON  =',F11.4/
     $          ' PHOTMASS =',F11.4/
     $          ' RPHOMASS =',F11.4/
     $          ' FERMMASS =',F11.4/
     $          ' RFERMASS =',F11.4/' '/
     $          ' # OF FOCK STATES                =',I9)

        IF (IPRIWF .EQ. 1) THEN
          WRITE (14,921)
 921      FORMAT (' '/
     $    ' STATE |   FERMION    |  REALWF '/
     $    '       |  K+  KX  KY  |         '/
     $    ' ------------------------------')
          DO 20 ISTATE=1,NSTATES
             WRITE (14,925) ISTATE,
     $       KFERM(ISTATE),KXFERM(ISTATE),KYFERM(ISTATE),REALWF(ISTATE)
 925         FORMAT (I6,3X,I3,1X,I3,1X,I3,4X,F7.4)
 20       CONTINUE
        ENDIF
C************

        CALL VTTIME(IVIRTIME,ITOTTIME)
        TIME2 = DFLOAT(ITOTTIME)/100.0D0-START-TIME1
```

```
C-------------------------------------------------------
C     CALL SUBROUTINES KE, PE
C     TO FIND HAMILTONIAN MATRIX ELEMENT.
C-------------------------------------------------------
        CALL KE(REMSQHO,NSIZE,NSTATES,REALWF,
     $          KPLUSTOT,ALPERP,FERMMASS,
     $          KFERM,KXFERM,KYFERM)

        CALL VTTIME(IVIRTIME,ITOTTIME)
        TIME3 = DFLOAT(ITOTTIME)/100.0D0-START-TIME1-TIME2

        CALL PE(REMSQH2,IUVFERM,
     $          NSIZE,NSTATES,REALWF,
     $          ALPHAG,KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $          PHOTMASS,FERMMASS,RPHOMASS,RFERMASS,
     $          KFERM,KXFERM,KYFERM)

        CALL VTTIME(IVIRTIME,ITOTTIME)
        TIME4 = DFLOAT(ITOTTIME)/100.0D0-START-TIME1-TIME2-TIME3-TIME4

        REMSQHO = DFLOAT(KPLUSTOT)*REMSQHO
        REMSQH2 = DFLOAT(KPLUSTOT)*REMSQH2

C-------------------------------------------------------
C     CALL SUBROUTINE PRINTC TO PRINT OUT RESULTS.
C-------------------------------------------------------
        IF (IPRIOUT .EQ. 1)
     $    CALL PRINTC(WAVEFCN,TIME1,TIME2,TIME3,TIME4,
     $                KPLUSTOT,ICHARGE,IBC,NSTATES,NSIZE,
     $                KPLUSMAX,ALPHAG,ALAMBDA,ALPERP,EPSILON,
     $                PARA1,PARA2,PARA3,PARA4,PARA5,
     $                PHOTMASS,RPHOMASS,FERMMASS,RFERMASS,
     $                REMSQHO,REMSQH2,
     $                REALWF,
     $                KFERM,KXFERM,KYFERM)

        WRITE (24,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,REMSQHO-4.0D0
        WRITE (25,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,REMSQH2
        WRITE (26,930) KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $                 REMSQHO+REMSQH2-4.0D0
 930    FORMAT (3X,I4,2X,3F11.4,F15.10)

 100  CONTINUE

      STOP
      END


      SUBROUTINE STATESC(NSIZE,KPLUSTOT,IBC,
     $                   ALAMBDA,ALPERP,
     $                   RFERMASS,
     $                   NSTATES,TOOMANY,
     $                   KFERM,KXFERM,KYFERM)
C-------------------------------------------------------
```

```
C     THIS SUBROUTINE GENERATES THE FOCK STATES CONSISTENT WITH
C     KPLUSTOT, ALAMBDA, ALPERP.
C     FOR Q=-1 ONLY KEEP STATES WITH 1 FERMION AND 0 PHOTONS.
C     FOR Q=0 ONLY KEEP STATES WITH 1 FERMION PAIR AND 0 PHOTONS.
C
C
C     OUTPUT VARIABLES:
C     NSTATES   NUMBER OF FOCK STATES.
C
C     TOOMANY   LOGICAL VARIABLE.  TOOMANY=TRUE  IF NSTATES .GT. NSIZE.
C                                  TOOMANY=FALSE IF NSTATES .LE. NSIZE.
C
C     KFERM,    KPLUS, KX, KY OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM,
C
C
C     INPUT VARIABLES:
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C               TO NSTATES.
C
C     KPLUSTOT  TOTAL KPLUS OF INCOMING, OUTGOING STATES.
C
C     IBC       ONLY HAVE EVEN FERMION KPLUS IF IBC=2,
C               ODD FERMION KPLUS IF IBC=1.
C
C     ALAMBDA   VALUE OF CUT-OFF MASS.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     RFERMASS  FERMION MASS TO BE USED IN COVARIANT CUT-OFF.
C
C
C     USAGE NOTES:
C     1) THIS ROUTINE ASSUMES THAT THE TOTAL KPERP OF THE INCOMING
C        AND OUTGOING FOCK STATES IS ZERO.  THIS IS NECESSARY IN THE
C        CALCULATION OF INVARIANT MASSES.
C     2) FOCK STATES ARE GENERATED WITH KPERP=0.
C     3) FERMION CHARGE IS ASSUMED TO BE -1.
C     4) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     5) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)
      DIMENSION
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE)

      LOGICAL TOOMANY

      PI = 3.141592653589793DO
      SMALL = 1.OD-13

C-------------------------------------------------------------
```

```
C     INITIALIZE VARIOUS ARRAYS TO ZERO.
C-------------------------------------------------------
      DO 10 I = 1,NSIZE
        KFERM(I) = 0
        KXFERM(I) = 0
        KYFERM(I) = 0
 10   CONTINUE

      TOOMANY = .FALSE.

C-------------------------------------------------------
C     GENERATE STATES WITH 1 FERMION PAIR, 0 PHOTONS (ICHARGE=0).
C     RECALL THAT KPLUSTOT MUST BE EVEN FOR ICHARGE = 0.
C-------------------------------------------------------
      ISTATE = 0

C************DIAGNOSTICS
C     WRITE (16,920) ALAMBDA**2
C920  FORMAT (' '/' 1 PAIR STATES  LAMBDA**2 =',F8.3/
C    $'  KPLUSF KPLUSA KXF  KYF  KXA  KYA  INVMASS'/
C    $'  ----------------------------------------')
C************

C     FIGURE OUT WHAT VALUES OF FERMION KPLUS TO RUN OVER.
      IF (IBC .EQ. 1) THEN
        IFIRSTKF = 1
        ILASTKF = KPLUSTOT - 1
       ELSEIF (IBC .EQ. 2) THEN
        IFIRSTKF = 2
        ILASTKF = KPLUSTOT - 2
      ENDIF

      DO 20 KPLUSF=IFIRSTKF,ILASTKF,2
        KPLUSA = KPLUSTOT-KPLUSF
        XF = DFLOAT(KPLUSF)/DFLOAT(KPLUSTOT)
        XA = DFLOAT(KPLUSA)/DFLOAT(KPLUSTOT)

C       ONLY CONTINUE IF SUM(MASS**2/X) .LE. LAMBDA**2.
        AINVMASS = RFERMASS**2/XF + RFERMASS**2/XA

C************DIAGNOSTICS
C       WRITE (16,921) KPLUSF,KPLUSA,AINVMASS
C921    FORMAT (3X,I4,3X,I4,21X,F8.3)
C************

        IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 20

C       FIGURE OUT WHAT LARGEST ALLOWED FERMION KX, KY IS.
        ARGUMENT = ALAMBDA**2*XF*XA - RFERMASS**2
        IF (ARGUMENT .LT. SMALL) THEN
          KPFMAX = 0
         ELSE
          KPFMAX = INT(ALPERP/PI*DSQRT(ARGUMENT) + SMALL)
        ENDIF
```

```
        DO 22 KXF=-KPFMAX,KPFMAX
        DO 22 KYF=-KPFMAX,KPFMAX
          KXA = -KXF
          KYA = -KYF
          AKPFSQ = (PI/ALPERP)**2*(DFLOAT(KXF**2) + DFLOAT(KYF**2))

C         KEEP STATE IF INVMASS**2 .LE. LAMBDA**2.
          AINVMASS = (AKPFSQ + RFERMASS**2)/XF
     $             + (AKPFSQ + RFERMASS**2)/XA

C************DIAGNOSTICS
C         WRITE (16,922) KXF,KYF,KXA,KYA,AINVMASS
C922      FORMAT (15X,I4,1X,I4,1X,I4,1X,I4,1X,F8.3)
C************

          IF (AINVMASS .GT. ALAMBDA**2+SMALL) GOTO 22

          ISTATE = ISTATE + 1
          IF (ISTATE .GT. NSIZE) THEN
            TOOMANY = .TRUE.
            RETURN
          ENDIF
          KFERM(ISTATE) = KPLUSF
          KXFERM(ISTATE) = KXF
          KYFERM(ISTATE) = KYF
 22     CONTINUE
 20   CONTINUE

      NSTATES = ISTATE
      RETURN
      END


      SUBROUTINE WF(REALWF,NSIZE,NSTATES,
     $              ALPHAG,KPLUSTOT,ALPERP,
     $              FERMMASS,
     $              PARA1,PARA2,PARA3,PARA4,PARA5,
     $              KFERM,KXFERM,KYFERM)
C-------------------------------------------------------------
C     THIS ROUTINE RETURNS THE VARIATIONAL WAVE FUNCTION
C     FOR THE INPUTTED FOCK STATES.
C
C
C     OUTPUT VARIABLES:
C     REALWF    REAL PART OF NORMALIZED FOCK STATE WAVE FUNCTIONS.
C
C
C     INPUT VARIABLES:
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C               TO NSTATES.
C
C     NSTATES   NUMBER OF FOCK STATES.
C
C     ALPHAG    VALUE OF COUPLING CONSTANT (=G**2/4PI).
```

```
C
C     KPLUSTOT  TOTAL KPLUS OF INCOMING, OUTGOING STATES.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     FERMMASS  FERMION MASS IN LAGRANGIAN.
C
C     PARA1,
C     PARA2,    VARIATIONAL PARAMETERS (PARA4, PARA5 NOT USED).
C     PARA3,
C     PARA4,
C     PARA5
C
C     KFERM,    KPLUS, KX, KY OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM,
C
C
C     USAGE NOTES:
C     1) FERMION CHARGE IS ASSUMED TO BE -1.
C     2) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     3) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE)

      PI = 3.141592653589793DO
      PERPFACT = PI**2/ALPERP**2
      AMSQARED = FERMMASS**2*(2.ODO - .25*ALPHAG**2)**2
C     AMSQARED = FERMMASS**2
C    $  *(2.ODO + .5DO*(DSQRT(1.ODO - ALPHAG**2) - 1.ODO))**2

C-------------------------------------------------------
C     INITIALIZE TO ZERO.
C-------------------------------------------------------
      DO 50 ISTATE = 1,NSTATES
       REALWF(ISTATE) = 0.ODO
 50   CONTINUE

C-------------------------------------------------------
C     WORK OUT VARIATIONAL WAVE FUNC FOR ICHARGE=0.
C-------------------------------------------------------
      DO 100 ISTATE = 1,NSTATES
        AKPERPSQ = PERPFACT*( DFLOAT(KXFERM(ISTATE))**2
     $                      +DFLOAT(KYFERM(ISTATE))**2)
        X = DFLOAT(KFERM(ISTATE))/DFLOAT(KPLUSTOT)

        REALWF(ISTATE) = +1.ODO/
     $    (( PARA1*AMSQARED
     $      -(AKPERPSQ+PARA2*FERMMASS**2)/(X*(1.ODO-X)))**2)**PARA3
```

```
 100  CONTINUE

C-------------------------------------------------------
C     NORMALIZE VARIATIONAL WAVE FUNCTION.
C-------------------------------------------------------
      WFNORM = 0.0D0
      DO 300 ISTATE = 1,NSTATES
       WFNORM = WFNORM + REALWF(ISTATE)**2
 300  CONTINUE
      WFNORM = DSQRT(WFNORM)
      DO 301 ISTATE = 1,NSTATES
       REALWF(ISTATE) = REALWF(ISTATE)/WFNORM
 301  CONTINUE

      RETURN
      END


      SUBROUTINE KE(REMSQHO,NSIZE,NSTATES,REALWF,
     $              KPLUSTOT,ALPERP,FERMMASS,
     $              KFERM,KXFERM,KYFERM)
C-------------------------------------------------------
C     THIS ROUTINE RETURNS THE VALUE OF THE HAMILTONIAN
C     BETWEEN THE INPUTTED VARIATIONAL STATES.
C
C
C     OUTPUT VARIABLES:
C     REMSQHO   CONTRIBUTION TO MATRIX ELEMENT FROM HO.
C
C
C     INPUT VARIABLES:
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C               TO NSTATES.
C
C     NSTATES   NUMBER OF FOCK STATES.
C
C     REALWF    REAL PART OF FOCK STATE WAVE FUNCTIONS.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     FERMMASS  FERMION MASS IN LAGRANGIAN.
C
C     KFERM,    KPLUS, KX, KY OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM
C
C
C     USAGE NOTES:
C     1) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     2) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)
```

```
      DIMENSION REALWF(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE)

      PI = 3.141592653589793DO
      PERPFACT = PI**2/ALPERP**2

C-------------------------------------------------------
C     INITIALIZE TO ZERO.
C-------------------------------------------------------
      REMSQHO = 0.0DO

C-------------------------------------------------------
C     LOOP OVER DIAGONAL MATRIX ELEMENTS.
C-------------------------------------------------------
      DO 10 ISTATE = 1,NSTATES
       IF (REALWF(ISTATE) .NE. 0.0DO) THEN

C        INITIALIZE TO ZERO.
         REALHO = 0.0DO

         AKPERPSQ = PERPFACT*DFLOAT(KXFERM(ISTATE)**2+KYFERM(ISTATE)**2)
         KAFER = KPLUSTOT - KFERM(ISTATE)
         REALHO = REALHO + (FERMMASS**2 + AKPERPSQ)
     $          *(1.0DO/DFLOAT(KFERM(ISTATE)) + 1.0DO/DFLOAT(KAFER))

C************DIAGNOSTICS
C        WRITE (16,910) ISTATE,REALHO
C910     FORMAT (' ISTATE REALHO =',I7,F11.4)
C************

         REMSQHO  = REMSQHO
     $   +REALHO*REALWF(ISTATE)**2

C      FINISHED WITH THIS FOCK STATE.  GO TO NEXT FOCK STATE.
       ENDIF
 10   CONTINUE

      RETURN
      END


      SUBROUTINE PE(REMSQH2,IUVFERM,
     $              NSIZE,NSTATES,REALWF,
     $              ALPHAG,KPLUSTOT,ALAMBDA,ALPERP,EPSILON,
     $              PHOTMASS,FERMMASS,RPHOMASS,RFERMASS,
     $              KFERM,KXFERM,KYFERM)
C-------------------------------------------------------
C     THIS ROUTINE RETURNS THE VALUE OF THE HAMILTONIAN
C     BETWEEN THE INPUTTED VARIATIONAL STATES.
C
C
C     OUTPUT VARIABLES:
C     REMSQH2   CONTRIBUTION TO MATRIX ELEMENT FROM INST. PHOTON.
C
```

```
C
C
C     INPUT VARIABLES:
C     IUVFERM   UV CUT-OFF APPLIED TO FERMIONS, ANTI-FERMIONS ONLY.
C
C     NSIZE     ARRAY DIMENSION OF NPHOT, NFERM,... AS DEFINED IN
C               CALLING ROUTINE.  IT SHOULD BE GREATER THAN OR EQUAL
C               TO NSTATES.
C
C     NSTATES   NUMBER OF FOCK STATES.
C
C     REALWF    REAL PART OF FOCK STATE WAVE FUNCTIONS.
C
C     ALPHAG    VALUE OF COUPLING CONSTANT (=G**2/4PI).
C
C     KPLUSTOT  TOTAL KPLUS OF INCOMING, OUTGOING STATES.
C
C     ALAMBDA   VALUE OF CUT-OFF MASS.
C
C     ALPERP    SIZE OF KPERP GRID.
C
C     EPSILON   MINIMUM PHOTON INVMASS**2.
C
C     PHOTMASS  PHOTON MASS IN LAGRANGIAN.
C
C     FERMMASS  FERMION MASS IN LAGRANGIAN.
C
C     RPHOMASS  PHOTON MASS TO BE USED IN COVARIANT CUT-OFF.
C
C     RFERMASS  FERMION MASS TO BE USED IN COVARIANT CUT-OFF.
C
C     KFERM,    KPLUS, KX, KY OF FERMION IN THE FOCK STATES.
C     KXFERM,   ARRAYS OF DIMENSION NSIZE.
C     KYFERM
C
C
C     USAGE NOTES:
C     1) THIS ROUTINE ASSUMES THAT THE TOTAL KPERP OF THE INCOMING
C        AND OUTGOING FOCK STATES IS ZERO.  THIS IS NECESSARY IN THE
C        CALCULATION OF INVARIANT MASSES.
C     2) FERMION CHARGE IS ASSUMED TO BE -1.
C     3) MASSES ARE IN UNITS OF ELECMASS.
C        LENGTHS ARE IN UNITS OF 1/ELECMASS.
C     4) REAL VARIABLES ARE DEFINED TO BE REAL*8 (DOUBLE PRECISION).
C-------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION REALWF(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE)

      SMALL = 1.0D-13
      PI = 3.141592653589793D0
      PERPFACT = (PI/ALPERP)**2
      PPLUS = 2.0D0*FERMMASS
```

```
C-------------------------------------------------------
C     INITIALIZE TO ZERO.
C-------------------------------------------------------
      REMSQH2 = 0.0D0

C-------------------------------------------------------
C     LOOP OVER OUTGOING FOCK STATES WITH 0 PHOTONS.
C-------------------------------------------------------
       DO 20 IOUTSTAT = 1,NSTATES
        IF (REALWF(IOUTSTAT) .EQ. 0.0D0) GOTO 20

C-------------------------------------------------------
C     LOOP OVER INCOMING FOCK STATES WITH 0 PHOTONS.
C     ASSUME HAMILTONIAN IS HERMITIAN, SO ONLY NEED TO CONSIDER
C     IOUTSTAT .GE. INSTATE.
C-------------------------------------------------------
        DO 30 INSTATE = 1,IOUTSTAT
         IF (REALWF(INSTATE) .EQ. 0.0D0) GOTO 30

C        INITIALIZE TO ZERO.
         REALH2 = 0.0D0

         KFERMI = KFERM(INSTATE)
         KXFERMI = KXFERM(INSTATE)
         KYFERMI = KYFERM(INSTATE)
         KAFERI = KPLUSTOT - KFERM(INSTATE)
         KXAFERI = -KXFERM(INSTATE)
         KYAFERI = -KYFERM(INSTATE)
         KFERMO = KFERM(IOUTSTAT)
         KXFERMO = KXFERM(IOUTSTAT)
         KYFERMO = KYFERM(IOUTSTAT)
         KAFERO = KPLUSTOT - KFERM(IOUTSTAT)
         KXAFERO = -KXFERM(IOUTSTAT)
         KYAFERO = -KYFERM(IOUTSTAT)

         AKPERPSQ = PERPFACT*DFLOAT(KXFERMI**2 + KYFERMI**2)
         ALPERPSQ = PERPFACT*DFLOAT(KXFERMO**2 + KYFERMO**2)
         X = DFLOAT(KFERMI)/DFLOAT(KPLUSTOT)
         Y = DFLOAT(KFERMO)/DFLOAT(KPLUSTOT)

         IQX = KXFERMI - KXFERMO
         IQY = KYFERMI - KYFERMO
         QPERPSQ = PERPFACT*DFLOAT(IQX**2 + IQY**2)

C-------------------------------------------------------
C     CALCULATE POTENTIAL.
C-------------------------------------------------------
C        NO MATRIX ELEMENT IF KFERMI = KFERMO.
         IF (KFERMI .EQ. KFERMO) GOTO 30

         IF (X .GT. Y) THEN

C         CHECK IF INVMASS**2 OF INTERMEDIATE STATE IS
C         .LE. LAMBDA**2.
```

```
            IF (IUVFERM .EQ. 0) THEN
             IF (( (ALPERPSQ + RFERMASS**2)/Y
     $           +(AKPERPSQ + RFERMASS**2)/(1-X)
     $           +(QPERPSQ  + RPHOMASS**2)/(X-Y))
     $          .GT. ALAMBDA**2 + SMALL) GOTO 30
            ENDIF
            IF (IUVFERM .EQ. 1) THEN
             IF (( (ALPERPSQ + RFERMASS**2)/Y
     $           +(AKPERPSQ + RFERMASS**2)/(1-X))
     $          .GT. ALAMBDA**2 + SMALL) GOTO 30
            ENDIF

            QPLUS = (X-Y)*PPLUS
            QMINUS = (AKPERPSQ + FERMMASS**2)/(X*PPLUS)
     $           - (ALPERPSQ + FERMMASS**2)/(Y*PPLUS)
            QFEYNSQ = QPLUS*QMINUS - QPERPSQ
            Q3 = .5*(QPLUS - QMINUS)

C       CHOOSE ONE OF FOLLOWING CUT-OFFS:
C       1) NO MATRIX ELEMENT IF INVMASS**2 OF INSTANTANEOUS PHOTON
C          .LT. EPSILON.
           IF ((QPERPSQ+RPHOMASS**2)/(X-Y) .LT. EPSILON-SMALL) GOTO 30

C       2) NO MATRIX ELEMENT IF Q_FR**2 .LT. EPSILON
C          IF (ABS(QFEYNSQ) .LT. EPSILON-SMALL) GOTO 30

C       3) NO MATRIX ELEMENT IF Q_VEC**2 .LT. EPSILON
C          IF ((QPERPSQ + Q3**2) .LT. EPSILON-SMALL) GOTO 30

C       CHOOSE ONE OF FOLLOWING QSQ:
           QSQ = -QFEYNSQ
C          QSQ = QPERPSQ + Q3**2

           REALH2 = REALH2 - 1.0D0/(QSQ + PHOTMASS**2)

          ENDIF

          IF (X .LT. Y) THEN

C          CHECK IF INVMASS**2 OF INTERMEDIATE STATE IS
C          .LE. LAMBDA**2.
           IF (IUVFERM .EQ. 0) THEN
            IF (( (AKPERPSQ + RFERMASS**2)/X
     $           +(ALPERPSQ + RFERMASS**2)/(1-Y)
     $           +(QPERPSQ  + RPHOMASS**2)/(Y-X))
     $          .GT. ALAMBDA**2 + SMALL) GOTO 30
           ENDIF
           IF (IUVFERM .EQ. 1) THEN
            IF (( (AKPERPSQ + RFERMASS**2)/X
     $           +(ALPERPSQ + RFERMASS**2)/(1-Y))
     $          .GT. ALAMBDA**2 + SMALL) GOTO 30
           ENDIF

           QPLUS = (Y-X)*PPLUS
           QMINUS = (AKPERPSQ + FERMMASS**2)/((1.0-X)*PPLUS)
```

```
     $           - (ALPERPSQ + FERMMASS**2)/((1.0-Y)*PPLUS)
          QFEYNSQ = QPLUS*QMINUS - QPERPSQ
          Q3 = .5*(QPLUS - QMINUS)

C       CHOOSE ONE OF FOLLOWING CUT-OFFS:
C       1) NO MATRIX ELEMENT IF INVMASS**2 OF INSTANTANEOUS PHOTON
C          .LT. EPSILON.
           IF ((QPERPSQ+RPHOMASS**2)/(Y-X) .LT. EPSILON-SMALL) GOTO 30

C       2) NO MATRIX ELEMENT IF Q_FR**2 .LT. EPSILON
C          IF (ABS(QFEYNSQ) .LT. EPSILON-SMALL) GOTO 30

C       3) NO MATRIX ELEMENT IF Q_VEC**2 .LT. EPSILON
C          IF ((QPERPSQ + Q3**2) .LT. EPSILON-SMALL) GOTO 30

C       CHOOSE ONE OF FOLLOWING QSQ:
           QSQ = -QFEYNSQ
C          QSQ = QPERPSQ + Q3**2

           REALH2 = REALH2 - 1.0D0/(QSQ + PHOTMASS**2)

         ENDIF

         IF (REALH2 .NE. 0.0D0) THEN
          REALH2 = (2.0D0*FERMMASS/DFLOAT(KPLUSTOT))**2*REALH2
          REALH2 = 2.0*REALH2/ALPERP**2

C************DIAGNOSTICS
C         WRITE (16,920) INSTATE,IOUTSTAT,QFEYNSQ,REALH2
C920      FORMAT (' INSTATE IOUTSTAT QFEYNSQ REALH2 =',2I6,2F12.7)
C************

          FACT = 2.0D0
          IF (INSTATE .EQ. IOUTSTAT) FACT = 1.0D0
          REMSQH2 = REMSQH2
     $     +FACT*REALH2*ALPHAG*REALWF(IOUTSTAT)*REALWF(INSTATE)
         ENDIF

C       FINISHED WITH THIS FOCK STATE.  GO TO NEXT FOCK STATE.
 30     CONTINUE
 20    CONTINUE

      RETURN
      END


      SUBROUTINE PRINTC(WAVEFCN,TIME1,TIME2,TIME3,TIME4,
     $                  KPLUSTOT,ICHARGE,IBC,NSTATES,NSIZE,
     $                  KPLUSMAX,ALPHAG,ALAMBDA,ALPERP,EPSILON,
     $                  PARA1,PARA2,PARA3,PARA4,PARA5,
     $                  PHOTMASS,RPHOMASS,FERMMASS,RFERMASS,
     $                  REMSQHO,REMSQH2,
     $                  REALWF,
     $                  KFERM,KXFERM,KYFERM)
C-------------------------------------------------------
```

```
C     THIS SUBROUTINE PRINTS OUT RESULTS OF PROGRAM COULOMB.
C-------------------------------------------------------------
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      IMPLICIT INTEGER (I-N)

      DIMENSION WAVEFCN(KPLUSMAX)

      DIMENSION REALWF(NSIZE),
     $  KFERM(NSIZE),KXFERM(NSIZE),KYFERM(NSIZE)

      LOGICAL ODDKPLUS

      PI = 3.141592653589793DO

C************DIAGNOSTICS: RESULTS FROM QEDVAR
      RPERP = ALAMBDA*ALPERP/PI
      XALPERP = ALPERP*ALPHAG

      WRITE (15,910) KPLUSTOT,ICHARGE,IBC,
     $  ALPHAG,ALAMBDA,ALPERP,XALPERP,RPERP,EPSILON,
     $  PHOTMASS,RPHOMASS,FERMMASS,RFERMASS,
     $  PARA1,PARA2,PARA3,PARA4,PARA5,NSTATES,
     $  TIME1,TIME2,TIME3,TIME4,
     $  TIME1+TIME2+TIME3+TIME4,
     $  REMSQHO-4.ODO,REMSQH2,REMSQHO+REMSQH2-4.ODO
 910  FORMAT (' INPUT (MASSES ARE IN UNITS OF ELECMASS):'/
     $        ' KPLUSTOT ICHARGE IBC =',3I4/
     $        ' ALPHAG   =',F11.4/
     $        ' ALAMBDA  =',F11.4/
     $        ' ALPERP   =',F11.4,'*1/ELECMASS =',F8.4,
     $                        '*BOHR  (RPERP =',F8.4,')'/
     $        ' EPSILON  =',F11.4/
     $        ' PHOTMASS =',F11.4/
     $        ' RPHOMASS =',F11.4/
     $        ' FERMMASS =',F11.4/
     $        ' RFERMASS =',F11.4/
     $        ' VAR PARAMETERS =',5F8.4/' '/
     $        ' # OF FOCK STATES             =',I9/' '/
     $        ' CPU TIME TO FIND FOCK STATES =',F8.2,' SEC'/
     $        ' CPU TIME TO WORK OUT WF      =',F8.2,' SEC'/
     $        ' CPU TIME TO FIND KE          =',F8.2,' SEC'/
     $        ' CPU TIME TO FIND PE          =',F8.2,' SEC'/
     $        ' TOTAL CPU TIME USED          =',F8.2,' SEC'/' '/
     $        '                           KE =',F15.10/
     $        '                           PE =',F15.10/
     $        '                       ENERGY =',F15.10)
C************

C-------------------------------------------------------------
C     PLOT WAVE FUNCTION.
C-------------------------------------------------------------
C     FIGURE OUT WHAT VALUES OF FERMION KPLUS TO RUN OVER
      KBIG = 0
      DO 40 ISTATE = 1,NSTATES
        K = KFERM(ISTATE)
```

```
        IF (K .GT. KBIG) KBIG = K
 40   CONTINUE
      KSMALL = KPLUSTOT
      DO 41 ISTATE = 1,NSTATES
        K = KFERM(ISTATE)
        IF (K .LT. KSMALL) KSMALL = K
 41   CONTINUE

C     FIGURE OUT WHAT VALUES FOR FERMION KX TO RUN OVER
      KXBIG = 0
      DO 50 ISTATE = 1,NSTATES
        KX = KXFERM(ISTATE)
        IF (KX .GT. KXBIG) KXBIG = KX
 50   CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
C     NOTE: IMAGEN PRINTS UP TO 80 CHARACTERS ACROSS.
C           NEED TO DECLARE LRECL 84 IN FILEDEF TO DO SO.
      WRITE (15,950)
 950  FORMAT (' '/' '/
     $' WAVE FUNCTION SQUARED AT KY=0:'/' '/
     $' NOTES: 1) VALUES SHOULD BE MULTIPLIED BY 1/10000 (***=10000)'/
     $'        2) KX IS IN UNITS OF ELECTRON MASS'/' ')
C************

C     LOOP OVER VALUES OF KX
      DO 53 KX = KXBIG,-KXBIG,-1

C       INITIALIZE WAVE FUNCTION TO ZERO
        DO 54 KPLUS = KSMALL,KBIG,2
          WAVEFCN(KPLUS) = 0.0D0
 54     CONTINUE

C       LOOP OVER FOCK STATE COMPONENTS
        DO 56 ISTATE = 1,NSTATES
          IF ((KXFERM(ISTATE) .EQ. KX) .AND.
     $        (KYFERM(ISTATE) .EQ. 0) ) THEN
            WAVEFCN(KFERM(ISTATE))
     $      = WAVEFCN(KFERM(ISTATE))
     $      + REALWF(ISTATE)**2
          ENDIF
 56     CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
        WRITE (15,952) INT(10000.0*DFLOAT(KX)*PI/ALPERP),
     $    (INT(.5+10000.0*WAVEFCN(KPLUS)),KPLUS=KSMALL,KBIG,2)
 952    FORMAT (1X,I5,' |',14(1X,I4))
C************

 53   CONTINUE

C************DIAGNOSTICS: RESULTS FROM QEDVAR
      WRITE (15,956)
     $  (INT(.5+10000.0*(DFLOAT(KPLUS)/DFLOAT(KPLUSTOT))),
     $   KPLUS=KSMALL,KBIG,2)
```

```
 956  FORMAT (
     $'       |-------------------------------------------------------',
     $'-------------'/
     $'   KX  X',14(1X,I4))
C************

      RETURN
      END
```

REFERENCES

1. H. C. Pauli and S. J. Brodsky, Phys. Rev. **D32**, 1993 (1985); Phys. Rev. **D32**, 2001 (1985).
2. T. Eller, H. C. Pauli and S. J. Brodsky, Phys. Rev. **D35**, 1493 (1987).
3. A. Harindranath and J. P. Vary, Phys. Rev. **D36**, 1141 (1987).
4. K. Hornbostel, S. J. Brodsky and H. C. Pauli, SLAC-PUB-4678 (1988); Phys. Rev. D (1990), in proof.
5. P. A. M. Dirac, Rev. Mod. Phys. **21**, 392 (1949).
6. G. McCartor, Z. Phys. **C41**, 271 (1988).
7. S. Weinberg, Phys. Rev. **150**, 1313 (1966).
8. S.D. Drell, D. Levy, T.M. Yan, Phys. Rev. **187**, 2159 (1969); Phys. Rev. **D1**, 1035 (1970); Phys. Rev. **D1**, 1617 (1970).
9. L. Susskind, Phys. Rev. **165**, 1535 (1968); L. Susskind, G. Frye, Phys. Rev. **164**, 2003 (1967).
10. J.D. Bjorken, J.B. Kogut, D.E. Soper, Phys. Rev. **D3**, 1382 (1971); J.B. Kogut, D.E. Soper, Phys. Rev. **D1**, 2901 (1970).
11. S.J. Brodsky, R. Roskies, R. Suaya, Phys. Rev. **D8**, 4574 (1973).
12. A. Casher, Phys. Rev. **D14**, 452 (1976).
13. S.J. Chang, R.G. Root, T.M. Yan, Phys. Rev. **D7**, 1133 (1973); S.J. Chang, T.M. Yan, Phys. Rev. **D7**, 1147 (1973).
14. G.P. Lepage, S.J. Brodsky, Phys. Rev. **D22**, 2157 (1980).
15. S.J. Brodsky, C.R. Ji, SLAC-PUB-3747, (1985).
16. G.P. Lepage, S.J. Brodsky, T. Huang, P.B. Mackenzie, published in the *Proceedings of the Banff Summer Institute*, 1981.

17. S. J. Brodsky, in *Trends in Theoretical Physics, vol. 1*, edited by P. J. Ellis and Y. C. Tang (Addison-Wesley, Redwood City, California, 1990).

18. S. J. Brodsky, G. P. Lepage, in *Perturbative Quantum Chromodynamics*, edited by A. H. Mueller (World Scientific, Singapore, 1989).

19. C. Hamer, private communications with author.

20. A. Hanson, T. Regge, C. Teitelboim, RX-748, Lincei, 1976.

21. S.J. Brodsky, C.R. Ji, SLAC-PUB-3756, (1985).

22. H. C. Pauli, private communication.

23. P.A.M. Dirac, *Lectures on Quantum Mechanics*, Belfer Graduate School, 1964.

24. P. J. Steinhardt, Ann. Phys. **128**, 425 (1980).

TABLE CAPTIONS

1: A comparison of light-cone and equal-time quantization.

2: Definitions in light-cone quantization.

TABLE 1

	Instant Form	**Front Form**
Hamiltonian	$H = \sqrt{\vec{P}^2 + m^2} + V$	$P^- = \frac{P_\perp^2 + m^2}{P^+} + V$
Conserved quantities	$E, \quad \vec{P}$	$P^-, \quad P^+, \quad \vec{P}_\perp$
Momenta	$P_z <> 0$	$P^+ > 0$
Bound state equation	$H\psi = E\psi$	$P^+P^-\psi = M^2\psi$
Vacuum	Complicated	Trivial

TABLE 2

Variables	τ = light-cone time = $x^+ = x^0 + x^3$ x^- = light-cone position = $x^0 - x^3$ $\vec{x}_\perp = (x^1, x^2)$
Covariant notation	$A^\mu = \left(A^+, A^-, \vec{A}_\perp\right)$
Metric	$(g^{\mu\nu}) = \begin{pmatrix} 0 & 2 & 0 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$
Dot product	$x \cdot y = x^\mu g_{\mu\nu} y^\nu = \frac{1}{2}\left(x^+ y^- + x^- y^+\right) - \vec{x}_\perp \cdot \vec{y}_\perp$
Mass shell condition	$P^+ P^- = \vec{P}_\perp^2 + M^2$
Derivative	$\partial_+ = \frac{\partial}{\partial x^+}$, $\partial_- = \frac{\partial}{\partial x^-}$, $\partial_i = \frac{\partial}{\partial x^i}$ $\partial^+ = 2\partial_-$, $\partial^- = 2\partial_+$, $\partial^i = -\partial_i$
Underscore notation	$\underline{x} = (x^-, \vec{x}_\perp)$, $\underline{k} = \left(k^-, \vec{k}_\perp\right)$ $\underline{k} \cdot \underline{x} = \frac{1}{2} k^+ x^- - \vec{k}_\perp \cdot \vec{x}_\perp$

FIGURE CAPTIONS

1) The left diagram schematically describes the traditional equal-time form of dynamics, the right Dirac's generalized form.

2) These three diagrams compare the instant form, point form, and front form.

3) The pion is expanded in a Fock basis $|n\rangle$. P^+, $\vec{P}_\perp$ are the pion's plus and perpendicular momenta. k_i^+, $\tilde{\vec{k}}_{\perp i}$ are the ith constituent's plus and perpendicular momenta.

4) V_{flip} light-cone diagrams.

5) V_{noflip} light-cone diagrams.

6) $V_{instphot}$ light-cone diagrams.

7) $V_{instferm}$ light-cone diagrams.

8) Decomposition of positronium into Fock states.

9) Light-cone perturbation theory graphs contributing to Møller scattering. k_1^+ is assumed to be larger than k_3^+.

10) Comparison of ground state energy with (Y) and without (N) infrared cut-off.

11) Intermediate states that may need Fermi statistics.

12) More intermediate states that may need Fermi statistics.

13) Light by light scattering contribution.

14) Contribution to the Lamb shift.

15) One-loop LCPTh radiative corrections to fermion line.

16) One-loop fermion self-energy.

17) Fermion self-energy in time-ordered perturbation theory. The right graph is typically referred to as the Z-graph.

18) Fermion structure function from diagonalization. $\alpha = .3$, $K = 17$, $L_\perp = 10\frac{1}{m_e}$, $\Lambda = 2.3m_e$.

19) Fermion structure function from diagonalization. $\alpha = .6$, $K = 15$, $L_\perp = 8\frac{1}{m_e}$, $\Lambda = 2.7m_e$.

20) Mass squared spectrum from diagonalizing $\alpha = .6$, $K = 18$, $L_\perp = 14\frac{1}{m_e}$, $\Lambda = 2.3m_e$. The left column of states are majority (e^+e^-) and the right column of states are majority $(e^+e^-\gamma)$.

21) Ground state mass as a function of K for $\alpha = .6$, $L_\perp = 14\frac{1}{m_e}$, $\Lambda = 2.3m_e$.

22) Ground state mass as a function of $L_\perp$ for $\alpha = .6$, $K = 14$, $\Lambda = 2.3m_e$.

23) Physical electron's wavefunction for $\alpha = .6$, $K = 25$, $L_\perp = 12\frac{1}{m_e}$, $\Lambda = 3.5m_e$. Values shown are the absolute value of the amplitude squared, and should be divided by 10,000.

24) M^2, KE, PE versus K for $\alpha = .6$, $L_\perp = 20\frac{1}{m_e}$, $\Lambda = 2.4m_e$ with infrared cut-off. Points labelled E, K, P correspond to M^2, KE, PE, respectively.

25) M^2, KE, PE versus K for $\alpha = .6$, $L_\perp = 20\frac{1}{m_e}$, $\Lambda = 2.4m_e$ without infrared cut-off. Points labelled E, K, P correspond to M^2, KE, PE, respectively.

26) M^2 (E), kinetic energy (K), potential energy (P), spin-flip (F), no spin-flip (N), instantaneous photon (I) interaction contributions to M^2 as a function of $L_\perp$ for $\alpha = .6$, $K = 26$, $\Lambda = 2.5m_e$.

27) M^2 (E), kinetic energy (K), potential energy (P), spin-flip (F), no spin-flip (N), instantaneous photon (I) interaction contributions to M^2 as a function of K for $\alpha = .6$, $L_\perp = 20\frac{1}{m_e}$, $\Lambda = 2.5m_e$.

28) M^2 (E), kinetic energy (K), potential energy (P), spin-flip (F), no spin-flip (N), instantaneous photon (I) interaction contributions to M^2 as a function of Λ for $\alpha = .6$, $K = 26$, $L_\perp = 20\frac{1}{m_e}$.

29) Positronium wavefunction for $\alpha = .6$, $K = 42$, $L_\perp = 32\frac{1}{m_e}$, $\Lambda = 2.5m_e$.

Values shown are the absolute value of the amplitude squared, and should be divided by 10,000.

30) Light-cone data (V) compared with Coulomb data (C).

31) M^2 as a function of $\frac{1}{L_\perp^2}$ using Coulomb data.

32) M^2 as a function of $\frac{1}{K}$ using Coulomb data.

33) Some fermion mass counterterms needed to include the $(e^+e^-\gamma\gamma)$ Fock state.

34) Photon mass counterterms needed to include the $(e^+e^-e^+e^-)$ Fock state.

35) Representative instantaneous fermion interaction. The incoming particles are on the left, the outgoing on the right.

36) Representative instantaneous photon interaction. The incoming particles are on the left, the outgoing on the right.

37) Three graphs that occur in LCPTh for tree-level Møller scattering.

38) One-loop fermion self-mass.

39) One-loop fermion self-mass diagrams joined by instantaneous fermion.

40) N one-loop fermion self-mass pieces chained by $N-1$ instantaneous fermions.

41) Sum of N chained one-loop fermion self-mass diagrams.

42) One-loop photon self-mass.

43) One-loop fermion self-energy.

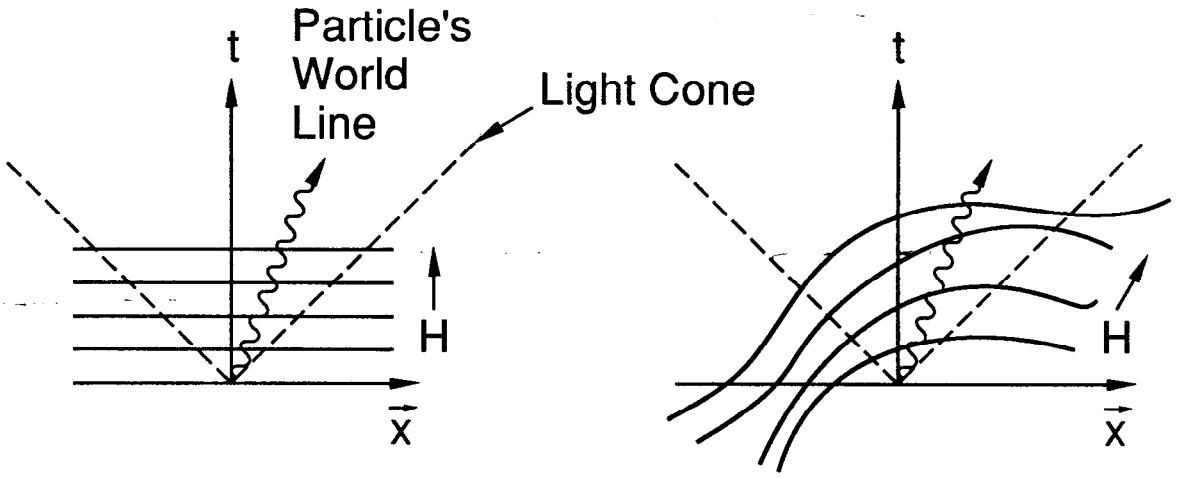
44) Two one-loop fermion self-energy contributions in TOPTh.

45) Usual time-ordering and Z-graph contributions to one-loop vacuum polarization in TOPTh.

46) One-loop vacuum polarization graph.

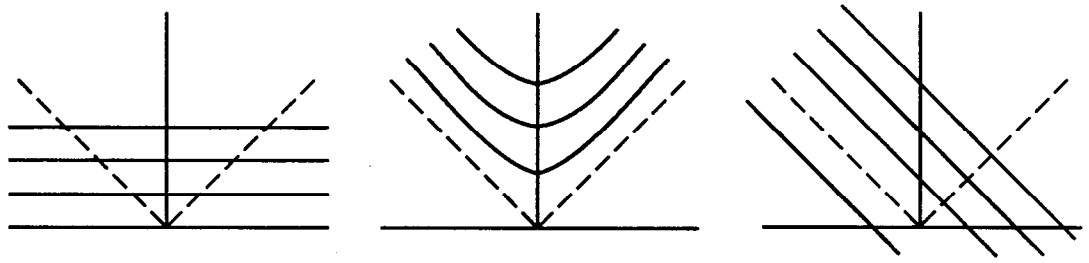
47) Three-point vertex in DLCQ, LCPTh.

48) Four-point instantaneous photon interaction in DLCQ, LCPTh.



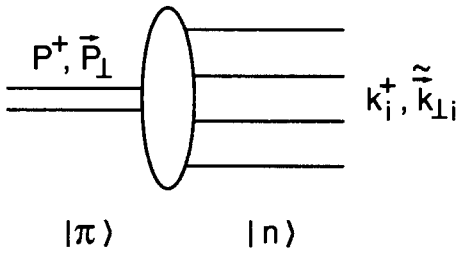
10-89 6482A1

Fig. 1



10-89 6482A2

Fig. 2



10-89 6482A3

Fig. 3

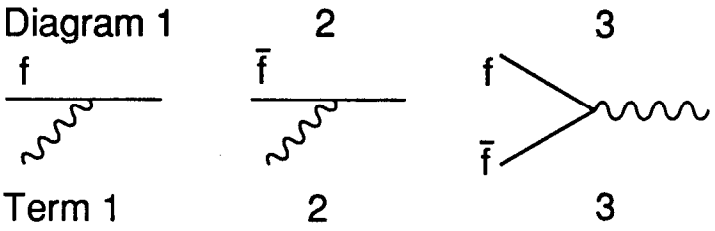


Fig. 4

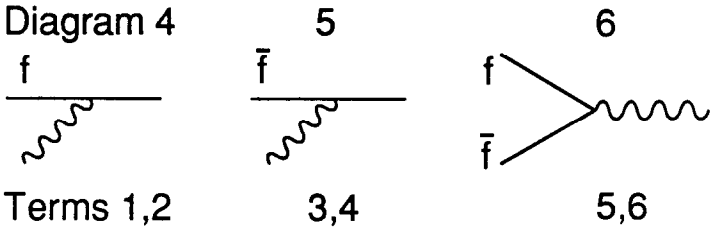


Fig. 5

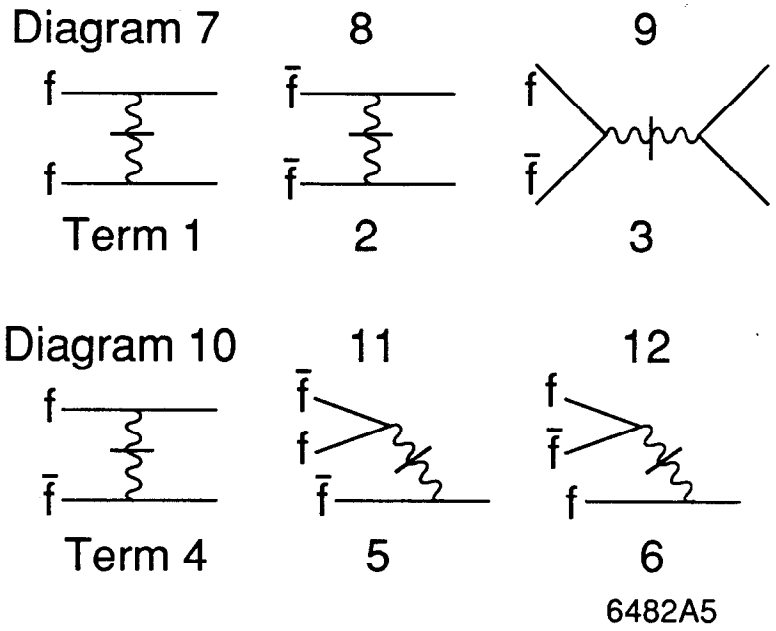


Fig. 6

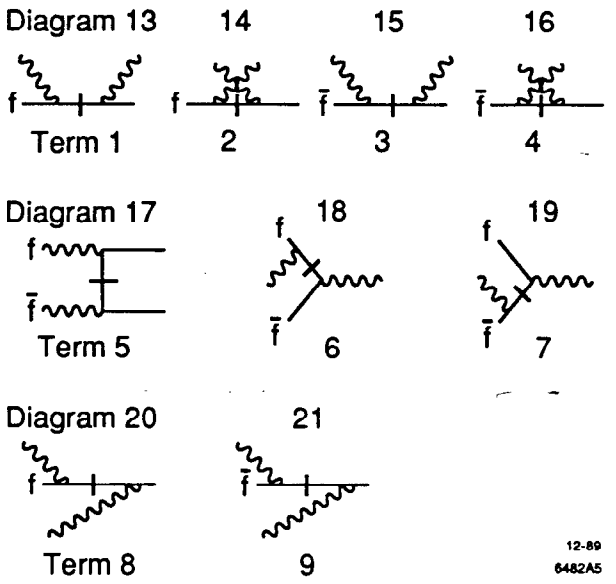


Fig. 7

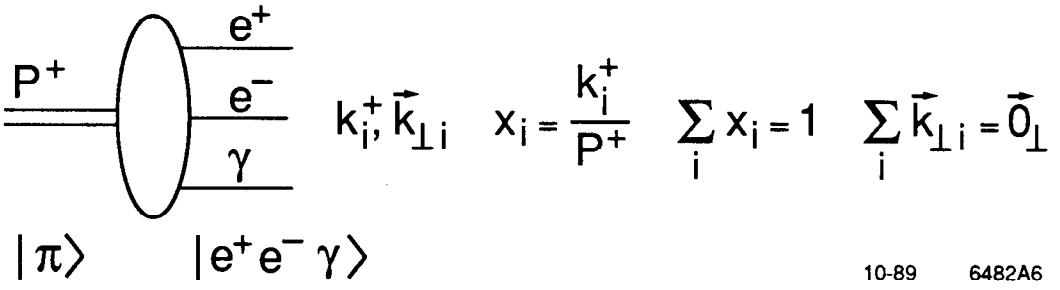


Fig. 8

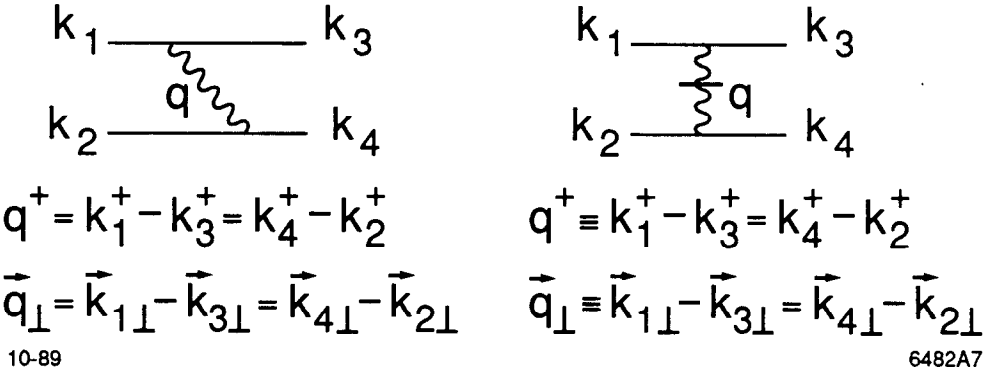


Fig. 9

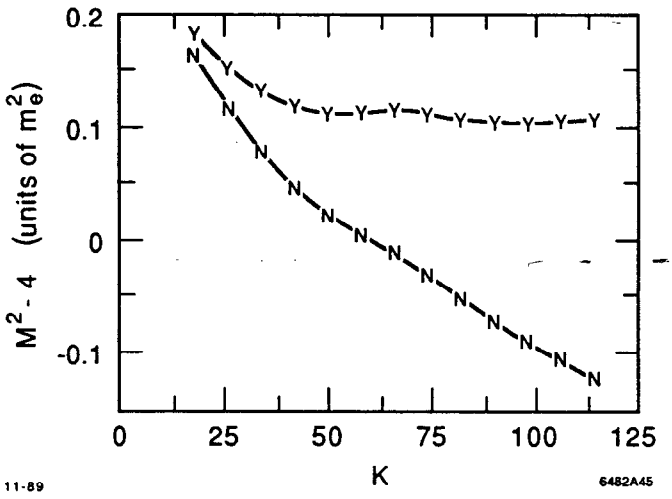


Fig. 10

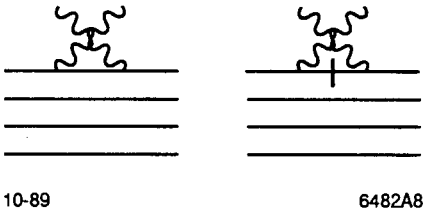


Fig. 11

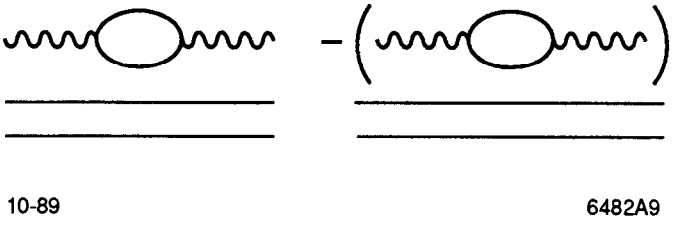
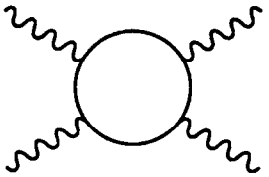
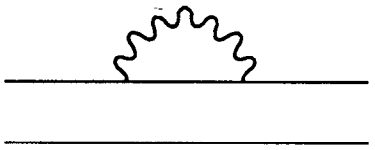


Fig. 12



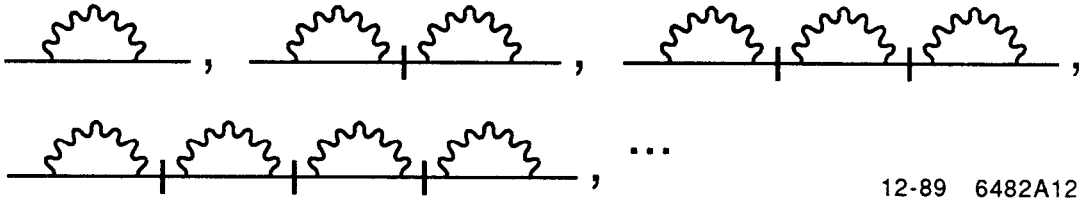
10-89 6482A10

Fig. 13



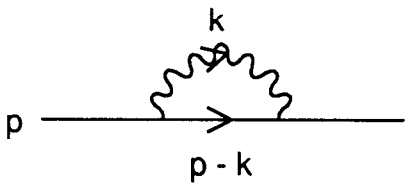
10-89 6482A11

Fig. 14



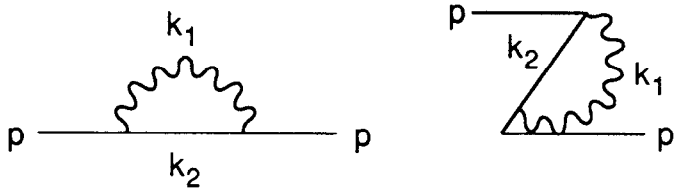
12-89 6482A12

Fig. 15



10-89 6482A13

Fig. 16



10-89 6482A14

Fig. 17

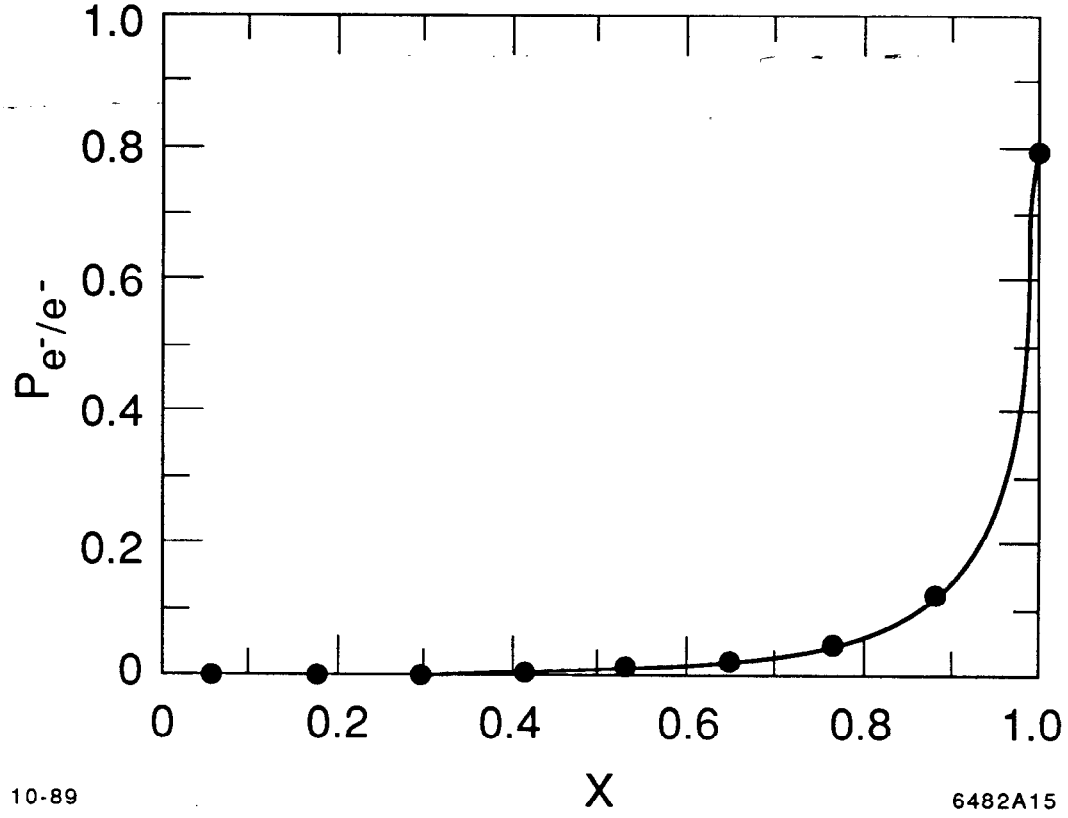


Fig. 18

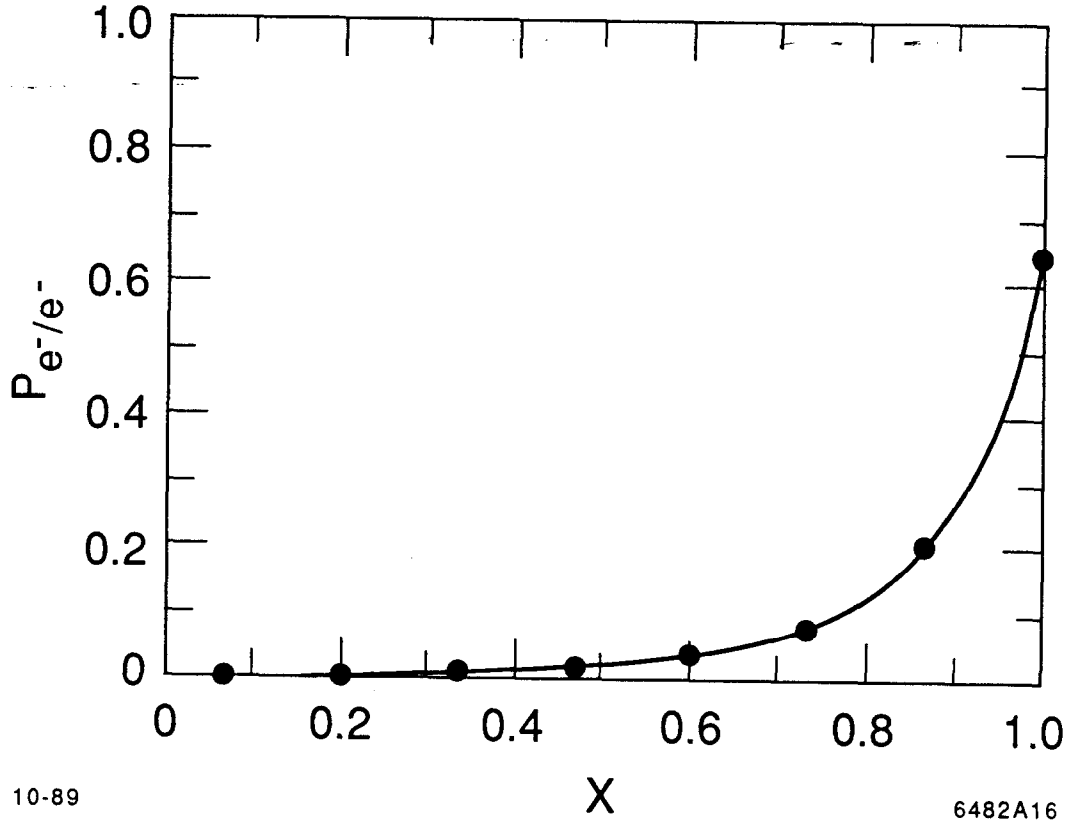


Fig. 19

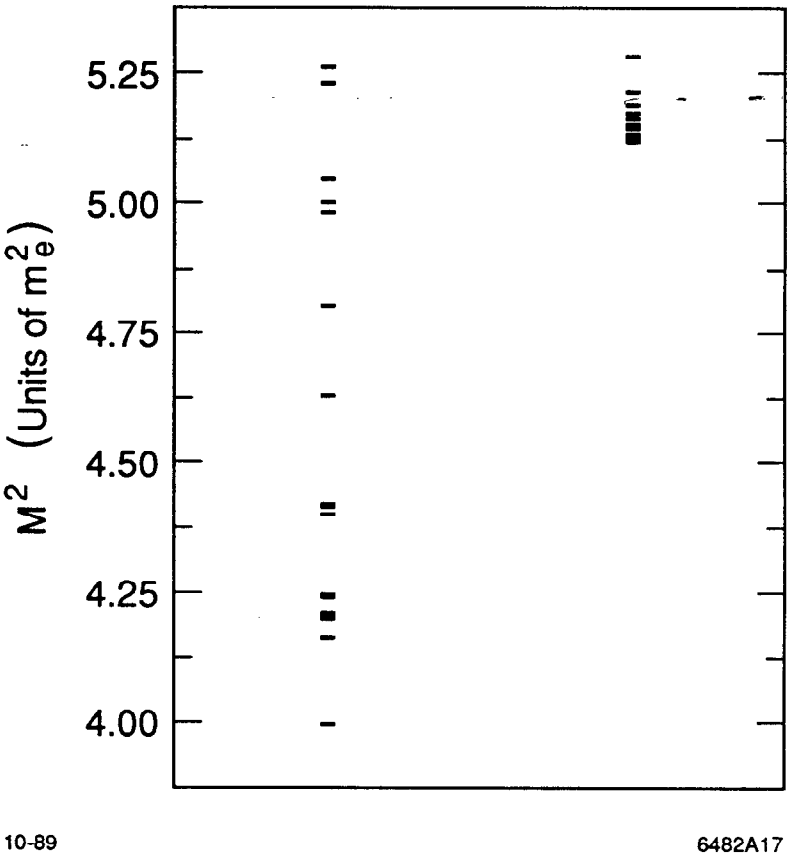


Fig. 20

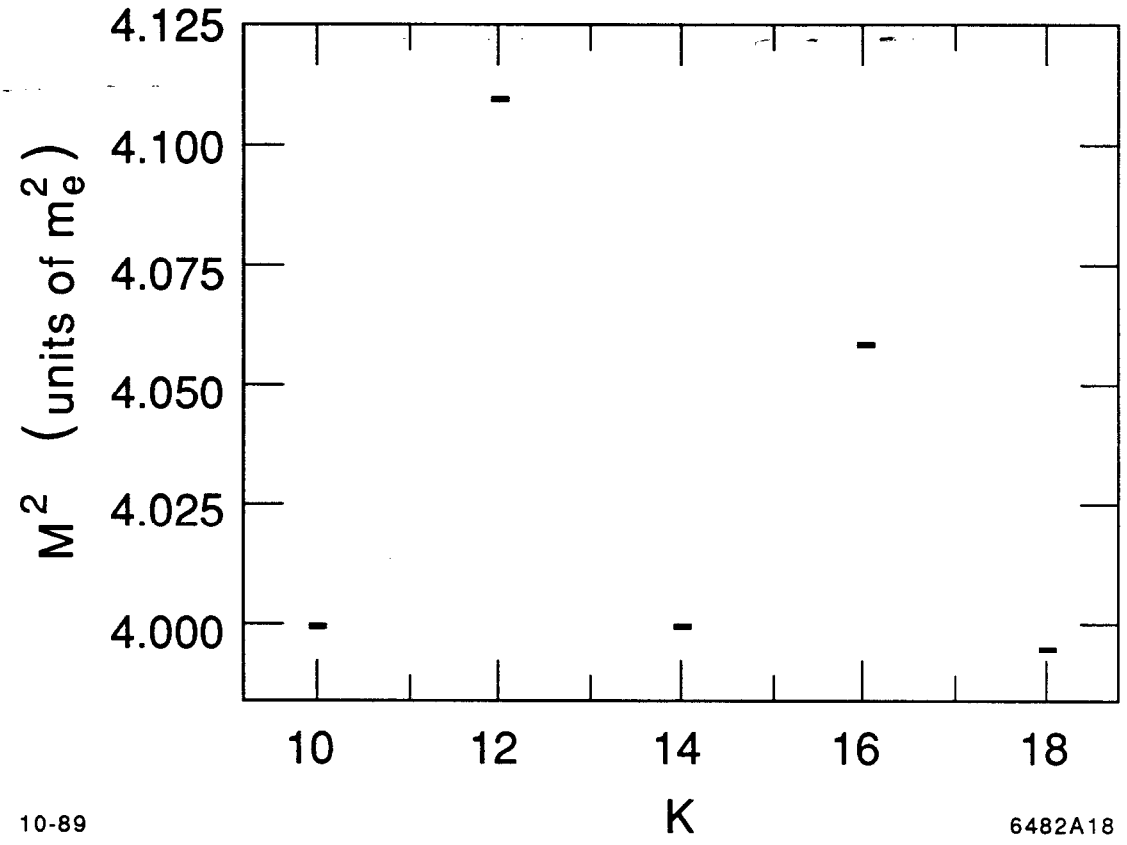

4.125
4.100
4.075
4.050
4.025
4.000
M^2 (units of m_e^2)
10
12
14
16
18
K
10-89
6482A18

Fig. 21

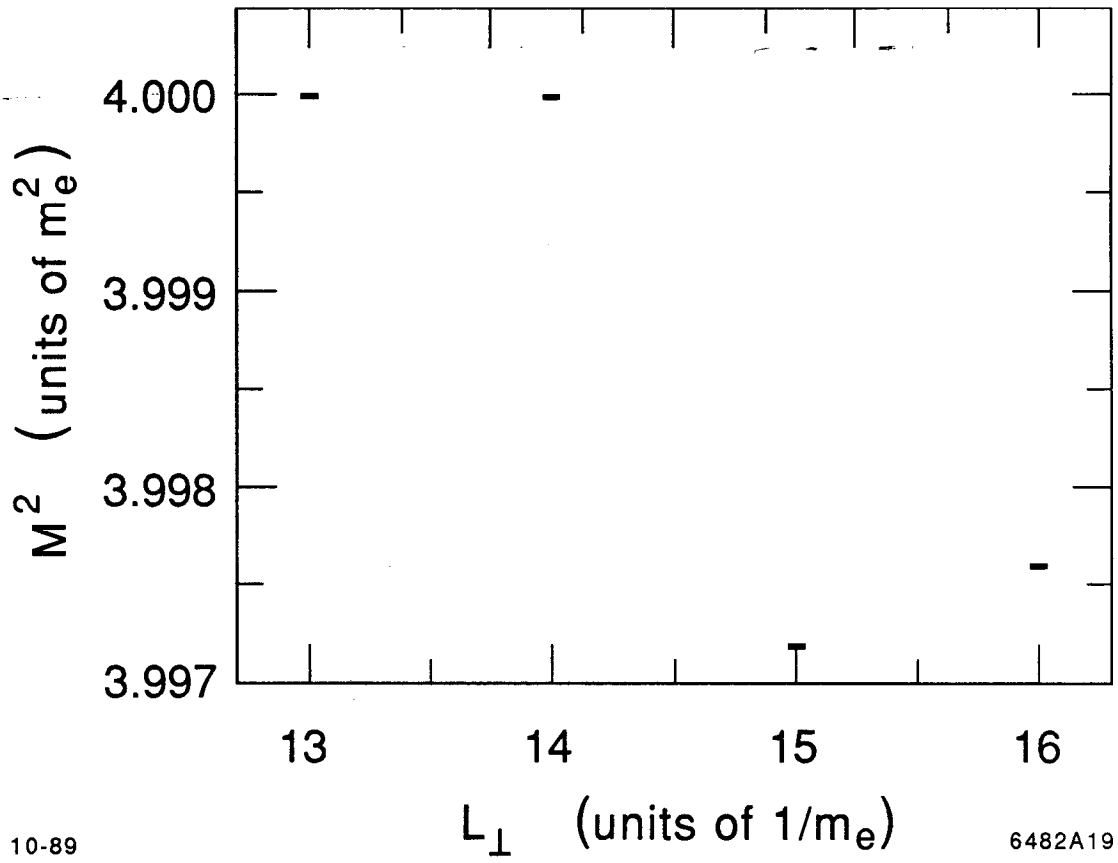

4.000
3.999
3.998
3.997
M^2 (units of m_e^2)
13
14
15
16
$L_\perp$ (units of $1/m_e$)
10-89
6482A19

Fig. 22

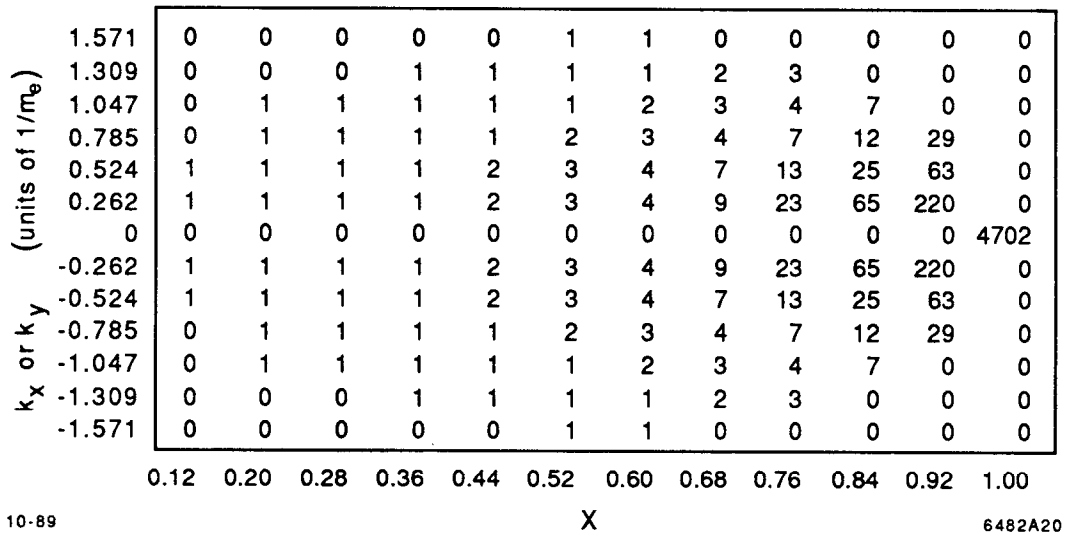

k_x or k_y (units of $1/m_e$)
1.571 0 0 0 0 0 1 1 0 0 0 0 0
1.309 0 0 0 1 1 1 1 2 3 0 0 0
1.047 0 1 1 1 1 1 2 3 4 7 0 0
0.785 0 1 1 1 1 2 3 4 7 12 29 0
0.524 1 1 1 1 2 3 4 7 13 25 63 0
0.262 1 1 1 1 2 3 4 9 23 65 220 0
0 0 0 0 0 0 0 0 0 0 0 0 4702
-0.262 1 1 1 1 2 3 4 9 23 65 220 0
-0.524 1 1 1 1 2 3 4 7 13 25 63 0
-0.785 0 1 1 1 1 2 3 4 7 12 29 0
-1.047 0 1 1 1 1 1 2 3 4 7 0 0
-1.309 0 0 0 1 1 1 1 2 3 0 0 0
-1.571 0 0 0 0 0 1 1 0 0 0 0 0
0.12 0.20 0.28 0.36 0.44 0.52 0.60 0.68 0.76 0.84 0.92 1.00
X
10-89
6482A20

Fig. 23

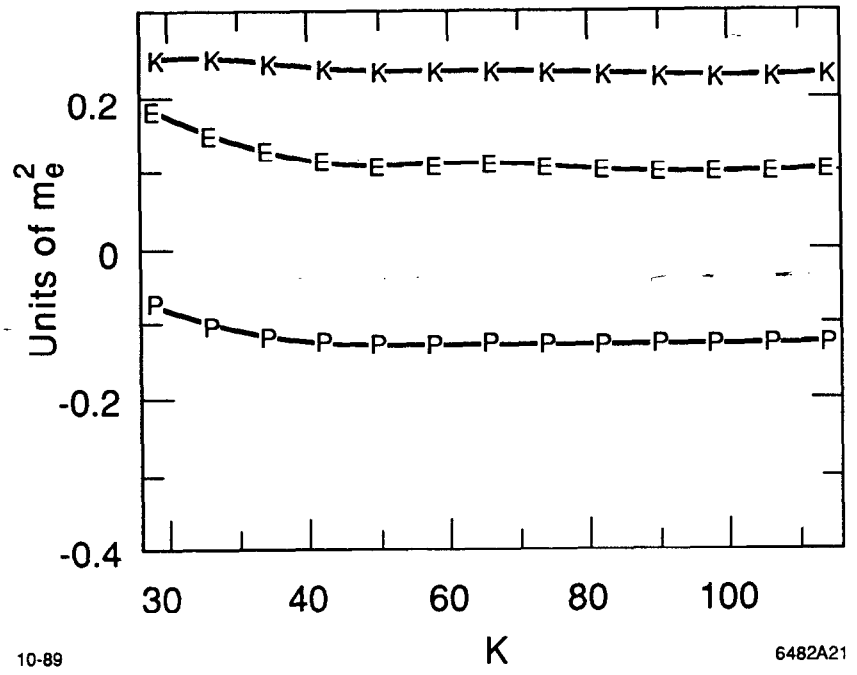

0.2
0
-0.4
Units of m_e^2
30
40
60
80
100
K
10-89
6482A21

Fig. 24

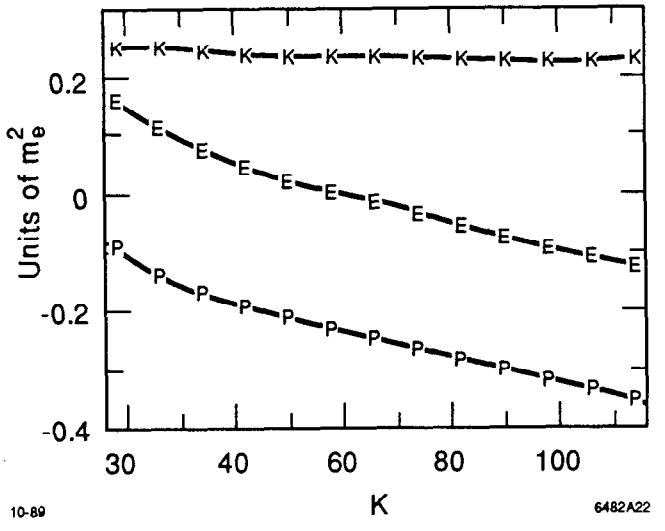

0.2
0
-0.2
-0.4
Units of m_e^2
30
40
60
80
100
K
10-89
6482A22

Fig. 25

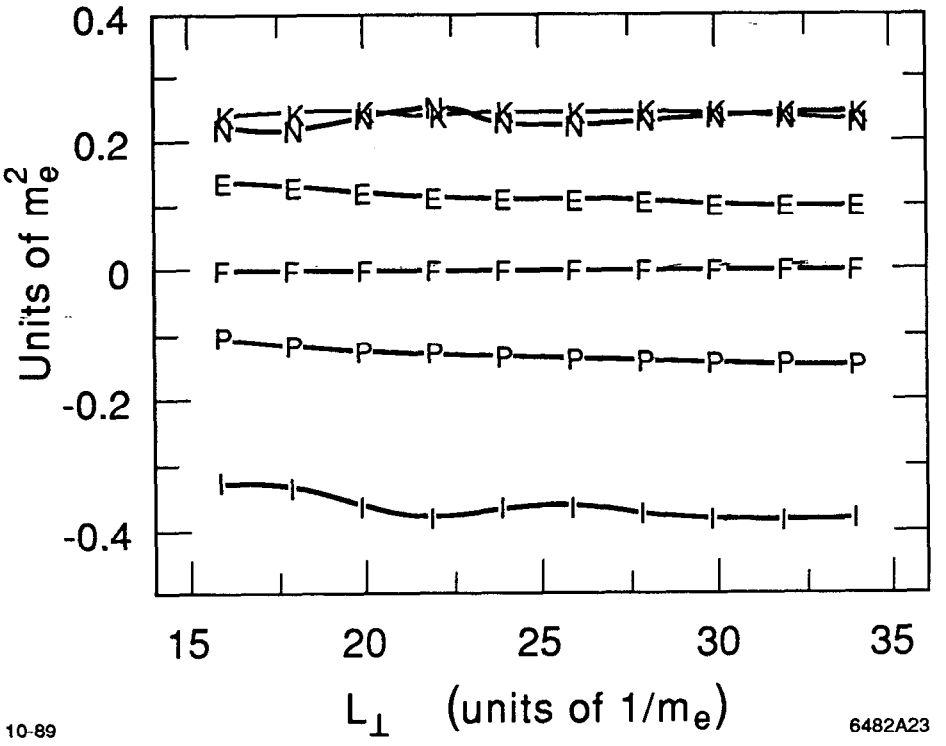

0.4
0.2
0
-0.2
-0.4
Units of m_e^2
15
20
25
30
35
$L_\perp$ (units of $1/m_e$)
10-89
6482A23

Fig. 26

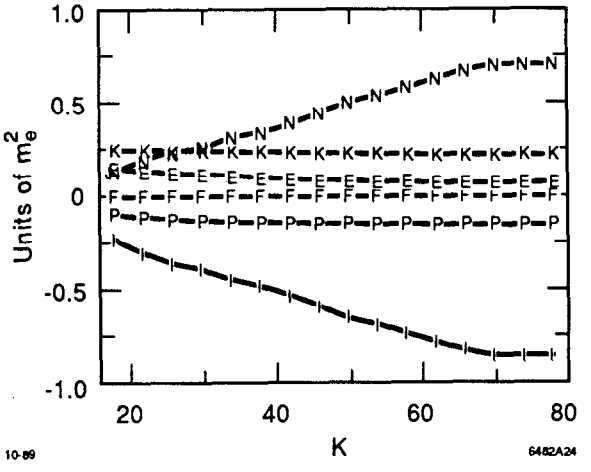

1.0
0.5
0
-0.5
-1.0
Units of m_e^2
20
40
60
80
K
10-89
6482A24

Fig. 27

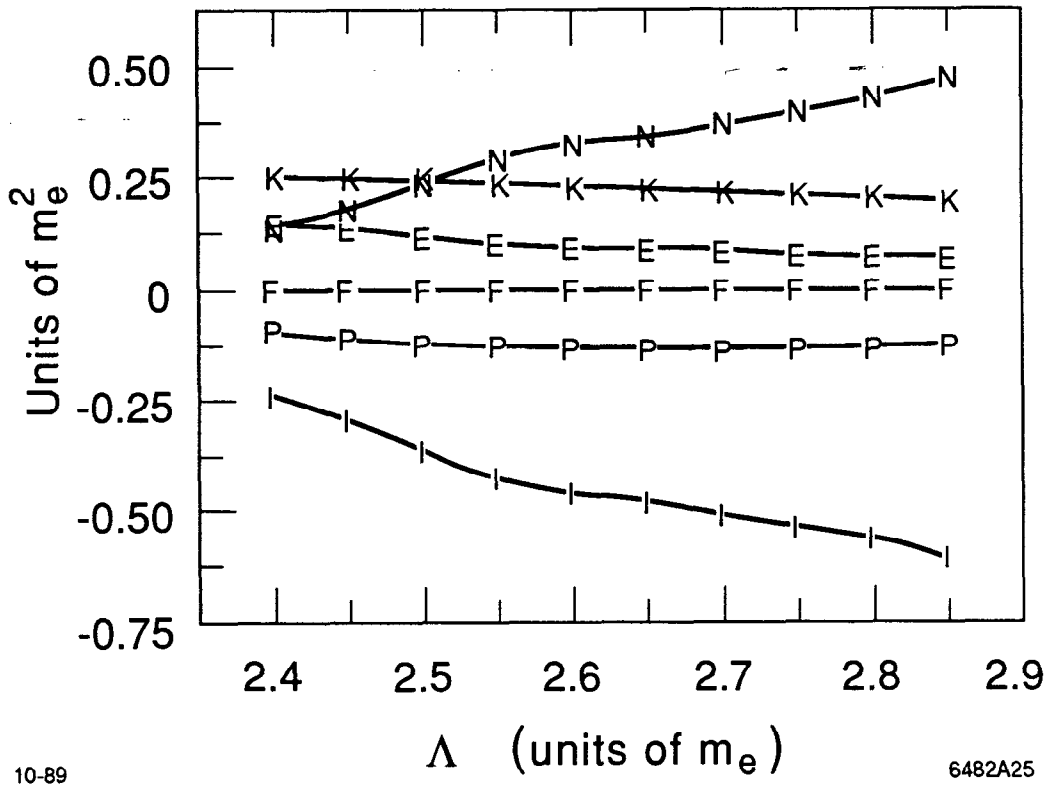

0.50
0.25
0
-0.25
-0.50
-0.75
Units of m_e^2
2.4
2.5
2.6
2.7
2.8
2.9
Λ (units of m_e)
10-89
6482A25

Fig. 28

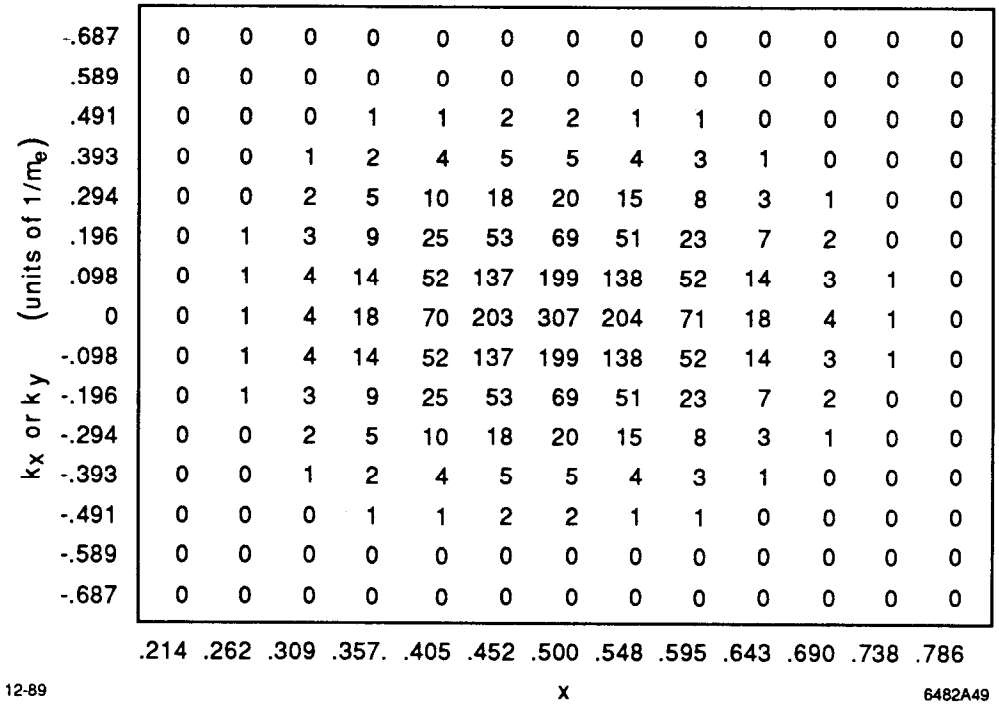

-.687 0 0 0 0 0 0 0 0 0 0 0 0 0
.589 0 0 0 0 0 0 0 0 0 0 0 0 0
.491 0 0 0 1 1 2 2 1 1 0 0 0 0
.393 0 0 1 2 4 5 5 4 3 1 0 0 0
.294 0 0 2 5 10 18 20 15 8 3 1 0 0
.196 0 1 3 9 25 53 69 51 23 7 2 0 0
.098 0 1 4 14 52 137 199 138 52 14 3 1 0
0 0 1 4 18 70 203 307 204 71 18 4 1 0
-.098 0 1 4 14 52 137 199 138 52 14 3 1 0
-.196 0 1 3 9 25 53 69 51 23 7 2 0 0
-.294 0 0 2 5 10 18 20 15 8 3 1 0 0
-.393 0 0 1 2 4 5 5 4 3 1 0 0 0
-.491 0 0 0 1 1 2 2 1 1 0 0 0 0
-.589 0 0 0 0 0 0 0 0 0 0 0 0 0
-.687 0 0 0 0 0 0 0 0 0 0 0 0 0
kx or ky (units of 1/me)
.214 .262 .309 .357. .405 .452 .500 .548 .595 .643 .690 .738 .786
X
12-89
6482A49

Fig. 29

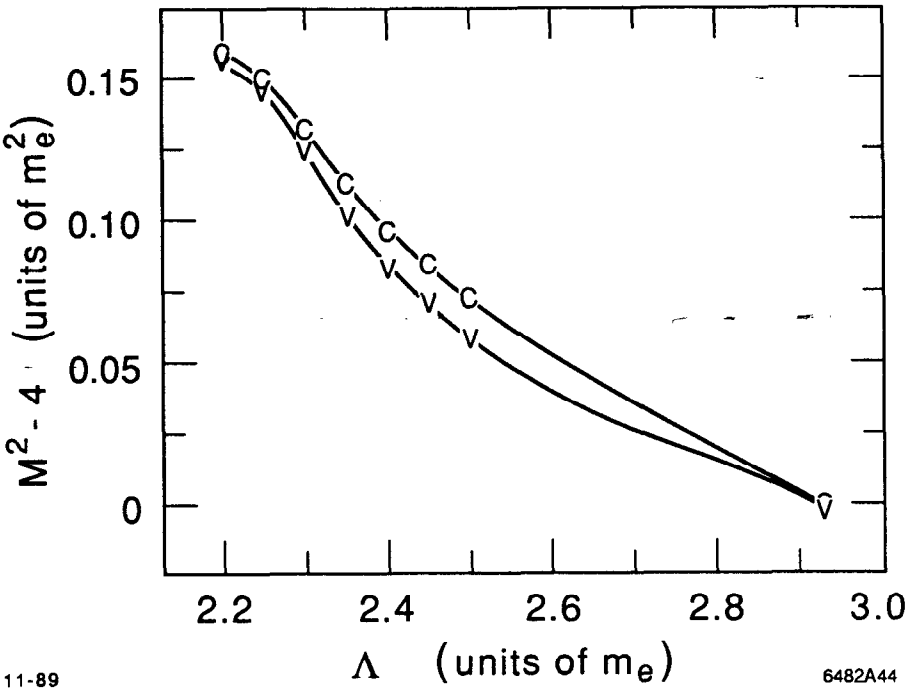

0.15
0.10
0.05
0
$M^2 - 4$ (units of m_e^2)
2.2
2.4
2.6
2.8
3.0
Λ (units of m_e)
11-89
6482A44

Fig. 30

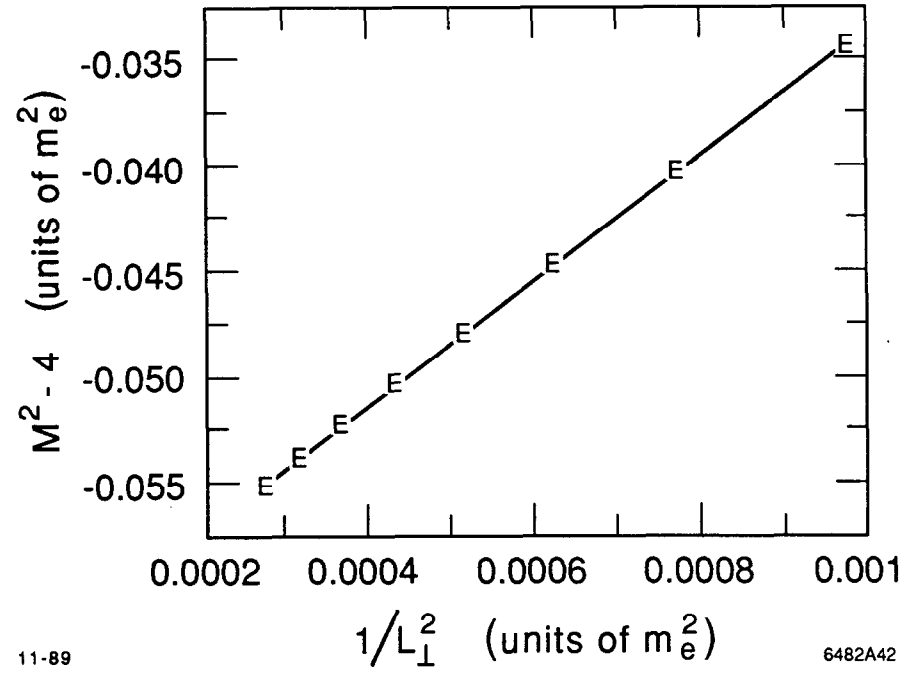

-0.035
-0.040
-0.045
-0.050
-0.055
$M^2 - 4$ (units of m_e^2)
0.0002
0.0004
0.0006
0.0008
0.001
$1/L_\perp^2$ (units of m_e^2)
11-89
6482A42

Fig. 31

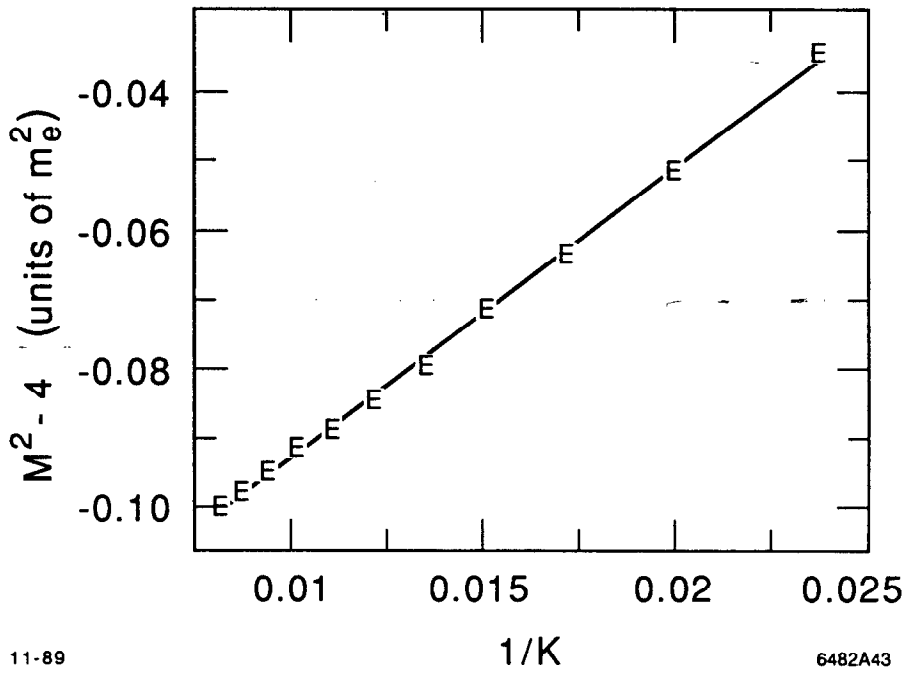


Fig. 32

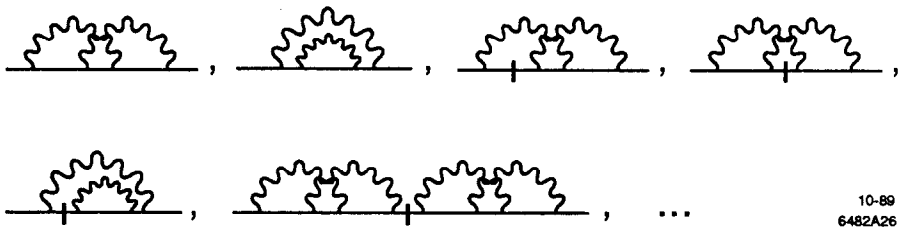


Fig. 33

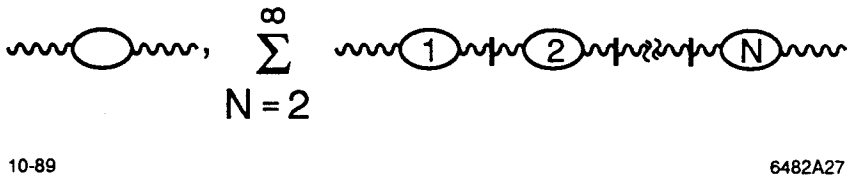


Fig. 34

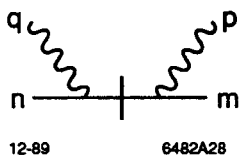


Fig. 35

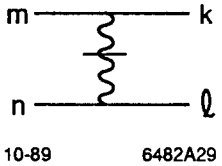


Fig. 36

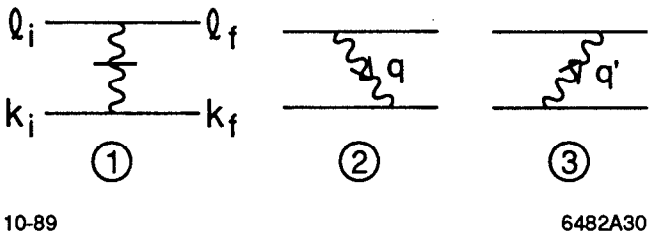


Fig. 37

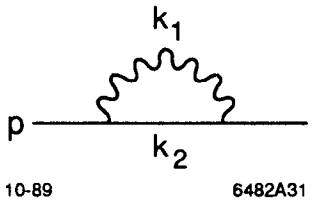


Fig. 38

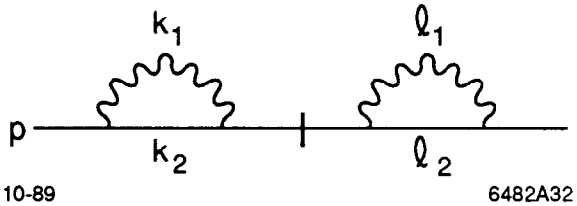


Fig. 39

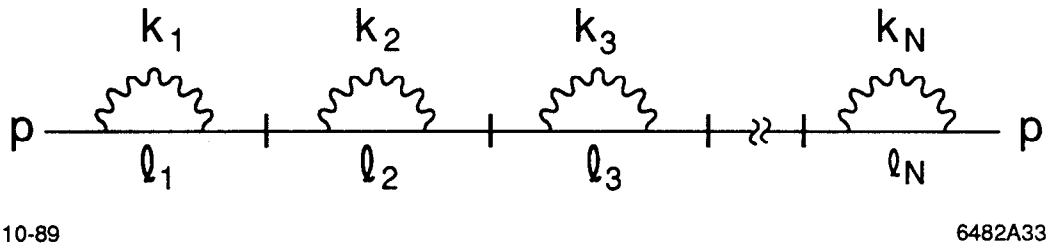
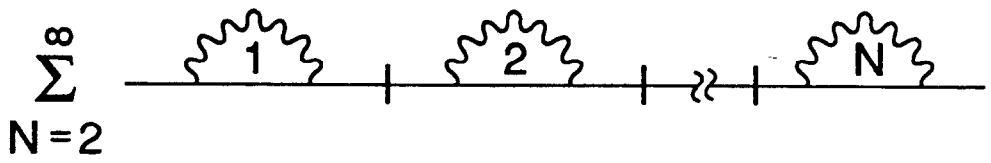
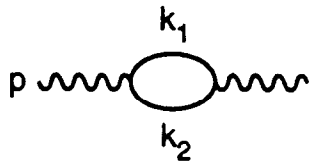


Fig. 40



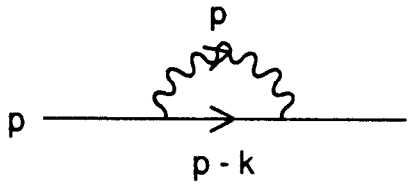
10-89 6482A34

Fig. 41



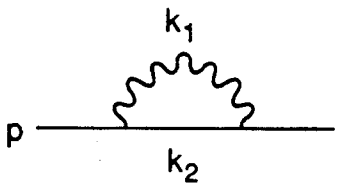
10-89 6482A35

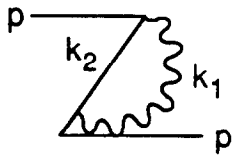
Fig. 42



10-89 6482A36

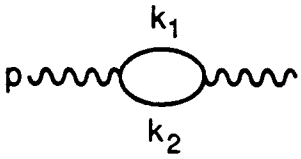
Fig. 43

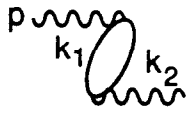




10-89 6482A37

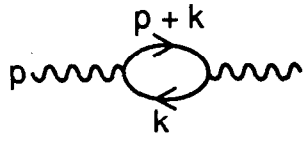
Fig. 44





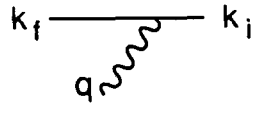
10-89 6482A38

Fig. 45



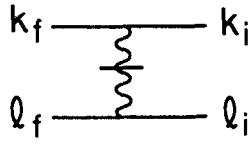
10-89 6482A39

Fig. 46



10-89 6482A40

Fig. 47



10-89 6482A41

Fig. 48