THE CALCULATION OF ELECTRIC DIPOLE OSCILLATOR STRENGTH FOR POLYATOMIC MOLECULES USING FLOATING SPHERICAL GAUSSIAN ORBITALS

By

CHANDA ASHOK SAPRE

Bachelor of Science
Institute of Science
University of Bombay
Bombay, India
1967

Master of Science
Institute of Science
University of Bombay
Bombay, India
1969

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
MASTER OF SCIENCE
May, 1973

THE CALCULATION OF ELECTRIC DIPOLE OSCILLATOR STRENGTH FOR POLY-ATOMIC MOLECULES USING FLOATING SPHERICAL GAUSSIAN ORBITALS

Thesis Approved:

Paul W Westhaus
Thesis Adviser

E E Kohnke

Charles T Butler

N N Durham
Dean of the Graduate College

ACKNOWLEDGMENTS

I sincerely thank Dr. Paul A. Westhaus for his suggestion of the problem and his patient guidance throughout the course of this research work. I also acknowledge the help of Mr. E. G. Bradford.

I also thank the officials of the University Computer Center, Oklahoma State University, Stillwater, Oklahoma, for their cooperation and generous help. The financial support in the form of teaching assistantship provided by the Physics Department and the Bio-Medical Research Grant are also acknowledged. I would also like to thank Mrs. Danvers for typing the manuscript.

TABLE OF CONTENTS

APPENDIXES

LIST OF TABLES

CHAPTER I

INTRODUCTION TO THE PROBLEM

The object of the thesis is to calculate the transition probability, in terms of the oscillator strength, for the water molecule for its transition from one electronic state to another as a result of its interaction with an electromagnetic field. The Frank-Condon principle is assumed to govern the transitions. The wave functions are obtained using Configuration Interaction. These wave functions are not exact eigenfunctions of the Hamiltonian; however, they are exact eigenfunctions of the symmetry operators of the molecule of water which has C_{2V} symmetry.

A completely rigorous quantum mechanical treatment of the interaction of light with matter requires the introduction of the quantum mechanical principles which include quantization of the electromagnetic field. However, the same phenomenon can also be treated semiclassically by Einstein's phenomenological theory of radiation based on conservation of momentum and energy principles applied to the interaction between a quantized matter system and a perturbing classical electromagnetic field (1). Since the behavior of a quantum-mechanical system in a given electromagnetic field is a quantum-mechanics problem, it is possible to use the theory of quantum transitions to calculate the probability that a given system (an atom or a molecule) will undergo a transition into an excited state or will relax from an excited state to a lower level energy state. In the former case, the system absorbs energy from the field while in the latter, it loses energy to the field. Thus, in the absorption phenomenon, when the system goes from a state of lower energy, E_i, to a state of higher energy, E_f, the excess energy $(E_f - E_i)$ is derived from the electromagnetic field. The condition of absorption is satisfied if the spectrum of incident electromagnetic radiation includes the wave of frequency

$$\omega_{if} = \frac{E_f - E_i}{\hbar} \quad .$$

In the lowest order, the transition is entirely due to the perturbation which is a harmonic function of time and has the above frequency ω_{if}. Similarly, the transition of the system from an excited state to a lower energy state can be looked upon as an emission of excess energy $(E_f - E_i)$ from the system to the perturbing field, and in the induced reverse transition the incident electromagnetic spectrum again must include the wave with frequency ω_{if}.

Our system consists of a biological molecule placed in an electromagnetic field corresponding to regions of the spectrum of either visible ($\lambda \sim$ 4000 Å - 8000 Å) or near ultraviolet ($\lambda \sim$ 2000 Å - 4000 Å) or far ultraviolet ($\lambda \sim$ 100 Å - 2000 Å) radiation. At these wavelengths (corresponding to energies greater than 2 ev), the electronic states of the molecules are changed in the transitions. The classification of the initial and final states of the system should, of course, include all their quantum numbers; namely vibrational, and perhaps rotational as well as electronic labels. However, we can sum over the vibrational states of the initial levels and therby arrive at a definition of an "electronic oscillator strength". These electronic oscillator strengths correspond to "vertical" or Frank-Condon transitions. According to the Frank-Condon principle, the transition in a molecule involving an electron jump takes place so rapidly, in comparison to the vibrational motion of the nuclei, that immediately afterwards the nuclei still have very nearly the same relative positions and velocities as before the transition (2). We will concern ourselves solely with the electronic transition probabilities (see Appendix A).

The electronic transition probability - as will be proven later - is proportional to the square of the transition operator matrix elements between the initial and final electronic states. For electric-dipole transitions these matrix elements are non-zero only if there is a change in the symmetry of the electronic charge distribution from initial to final state. These so-called allowed transitions can occur only between the electronic states of different symmetry of charge distribution whereas, the transitions between the states of same symmetry are forbidden (3). In the forbidden case, the transition moment, therefore, would be equal to zero in the first order. However, the forbidden transitions often exhibit measurable intensities, though much smaller than those of allowed transitions, since the symmetry of charge distribution can also be disturbed by nuclear vibrations. Also, the transition may indeed by possible due to higher order couplings with the electromagnetic field.

In our work we are mainly interested in the effect of visible and ultraviolet radiation on the biological molecules. We are in the process of developing a set of computing programs to treat large electronic systems. In this research we have developed an electric-dipole transition program for electronic transitions. Since the water molecule is one of the simplest yet interesting molecules, we have tested our program on water. The program itself is very general and may be used for any system whose wave functions are described in terms of Gaussian orbitals.

CHAPTER II

ELECTRIC DIPOLE TRANSITION PROBABILITY

In this chapter the electronic probability for transition of a molecule from a given initial ground state to some excited state as a result of its interaction with electromagnetic radiation is calculated.

Let the system in question consist of N particles (electrons) and A point nuclei. The above system could be an atom, a molecule, a solid or more generally a system of identical particles in an external electromagnetic field. Our system, in particular, is a molecule. The calculation of the electric dipole transition probability requires finding the exact stationary states of the system, an extremely complex problem as will be seen in the later chapter. As mentioned in the introduction, since the mass of the electrons is much smaller than the mass of nuclei, the Frank-Condon principle can be invoked. In computing electronic transitions the dynamical state depicts the system of N electrons in the field of fixed nuclei in an external electromagnetic field.

The Hamiltonian of the above system in Schrödinger picture is described by both a scalar potential (ϕ) and a vector potential ($\vec{A}$) (4).

$$H_s(t) = H_{os} + H_s'(t)$$

i.e.,

$$H_s(t) = \Bigg[\Bigg\{ \sum_{j=1}^{N} \frac{-\hbar^2}{2m} \nabla_j^2 + \sum_{j=1}^{N} \sum_{\alpha=1}^{A} \frac{-Z_\alpha e^2}{|\vec{r}_j - \vec{R}_\alpha|} + \sum_{1 \le j < k}^{N} \frac{e^2}{r_{jk}} \Bigg\} + \sum_{j=1}^{N} \frac{e i\hbar}{2m} \Big\{ \vec{\nabla}_j \cdot \vec{A}(\vec{r}_j, t) + \vec{A}(\vec{r}_j, t) \cdot \vec{\nabla}_j \Big\} + \sum_{j=1}^{N} \frac{e^2}{2m} A^2(\vec{r}_j, t) + \sum_{j=1}^{N} e\, \phi(\vec{r}_j . t) \Bigg] \tag{2.1}$$

where

$$H_{os} = \sum_{j=1}^{N} \frac{-\hbar^2}{2m} \nabla_j^2 + \sum_{j=1}^{N} \sum_{\alpha=1}^{A} \frac{-Z_\alpha e^2}{r_{j\alpha}} + \sum_{1 \le j < k}^{N} \frac{e^2}{r_{jk}} \tag{2.2}$$

= time independent electrostatic Hamiltonian of the isolated system.

$|\vec{r}_j - \vec{r}_k|$ is written as r_{jk}

and

$$H_s'(t) = \Bigg[\sum_{j=1}^{N} \frac{i\hbar e}{2m} \Big\{ \vec{\nabla}_j \cdot \vec{A}(\vec{r}_j, t) + \vec{A}(\vec{r}_j, t) \cdot \vec{\nabla}_j \Big\} + \sum_{j=1}^{N} \frac{e^2}{2m} A^2(\vec{r}_j, t) + \sum_{j=1}^{N} e\, \phi(\vec{r}_j, t) \Bigg]$$

= time dependent perturbation part of the Hamiltonian.

The whole problem can, therefore, be expressed in terms of finding the solution to the equation:

$$i\hbar \frac{d}{dt} | \Psi_s(t)> = H_s(t) | \Psi_s(t)> \tag{2.4}$$

with a given initial state of the system.

The Hamiltonian describes the system at all the times, i.e., it unfolds the system with time. We shall discuss the system in Schrödinger picture and in Interaction picture. The subscript s denotes the quantities in Schrödinger picture and I in Interaction picture.

The correspondence between the two pictures is defined by the following relations:

$$|\Psi_I(t)> = e^{iH_{os} t/\hbar} |\Psi_s(t)> \tag{2.5}$$

= the state function correspondence,

and

$$\Omega_I(t) = e^{iH_{os} t/\hbar} \Omega_s e^{-iH_{os} t/\hbar} \tag{2.6}$$

= operator correspondence.

The Hamiltonian in the two pictures can be written as

$$H_s(t) = H_{os} + H_s'(t) \tag{2.7}$$

$$H_I(t) = H_{OI} + H_I'(t) \tag{2.8}$$

When there is no perturbation, the Hamiltonians in the two pictures are equivalent. Equation (2.4) is the equation of motion in Schrödinger picture. The equation of motion in Interaction picture can be derived as follows:

$$|\Psi_I(t)\rangle = e^{iH_{os}t/\hbar}\,|\Psi_s(t)\rangle$$

so that

$$i\hbar\frac{d}{dt}\,|\Psi_I(t)\rangle = i\hbar\frac{d}{dt}\left[e^{iH_{os}t/\hbar}\,|\Psi_s(t)\rangle\right].$$

Now consider the right-hand side of this equation.

$$i\hbar\frac{d}{dt}\left[e^{iH_{os}t/\hbar}\,|\Psi_s(t)\rangle\right]$$

$$= i\hbar\left[\frac{i}{\hbar}H_{os}\,e^{iH_{os}t/\hbar}\,|\Psi_s(t)\rangle + e^{iH_{os}t/\hbar}\frac{d}{dt}|\Psi_s(t)\rangle\right]$$

$$= -H_{os}\,e^{iH_{os}t/\hbar}|\Psi_s(t)\rangle + e^{iH_{os}t/\hbar}\left[i\hbar\frac{d}{dt}|\Psi_s(t)\rangle\right]$$

$$= -H_{os}\,e^{iH_{os}t/\hbar}|\Psi_s(t)\rangle + e^{iH_{os}t/\hbar}\left[H_{os} + H_s'(t)\right]|\Psi_s(t)\rangle$$

Since $e^{iH_{os}t/\hbar}$ and H_{os} commute, the right-hand side of the equation reduces to

$$e^{iH_{os}t/\hbar}\,H_s'(t)\,|\Psi_s(t)\rangle$$

$$= e^{iH_{os}t/\hbar}\,H_s'(t)\,e^{-iH_{os}t/\hbar}\,|\Psi_I(t)\rangle$$

$$= H_I'(t)\,|\Psi_I(t)\rangle \quad ,$$

and we get

$$i\hbar\frac{d}{dt}\,|\Psi_I(t)\rangle = H_I'(t)\,|\Psi_I(t)\rangle \quad . \tag{2.9}$$

Thus the solution of Equation (2.9) with the given initial condition unfolds the system in Interaction picture.

Next, we define a unitary operator $U_I(t, t_o)$ in the Interaction picture such that, given a state $|\Psi_I(t_o)>$ at time t_o, it can take the system to a state $|\Psi_I(t)>$ at time t, i. e. (5),

$$|\Psi_I(t)> \; = \; U_I(t, t_o)\,|\Psi_I(t_o)> \quad . \tag{2.10}$$

Obviously, at time $t = t_o$, we require that

$$U_I(t_o, t_o) \; = \; 1.$$

Substituting Equation (2. 10) into Equation (2. 9), we get

$$i\hbar \frac{d}{dt}\left\{U_I(t, t_o)\,|\Psi_I(t_o)>\right\} = H_I'(t)\; U_I(t, t_o)\,|\Psi_I(t_o)>$$

$$= \; i\hbar\left\{\frac{d}{dt}U_I(t, t_o)\right\}|\Psi_I(t_o)> \; = \; H_I'(t)\; U_I(t, t_o)\,|\Psi_I(t_o)> \quad .$$

Since $|\Psi_I(t_o)>$ is arbitrary, it must follow that

$$i\hbar \frac{d}{dt}U_I(t, t_o) \; = \; H_I'(t)\; U_I(t, t_o) \quad . \tag{2.11}$$

This is an operator equation. The formal solution of Equation (2. 11) subject to the initial condition is

$$U_I(t, t_o) \; = 1 - \frac{i}{\hbar}\int_{t_o}^{t} dt_1\, H_I'(t_1)\; U_I(t_1, t_o) \quad . \tag{2.12}$$

It is proven in the following steps that Equation (2. 12) is a solution of Equation (2. 11) subject to the initial condition. Differentiating Equation (2. 12) with respect to t, we find

$$\frac{d}{dt}\left[U_I(t, t_o)\right] \; = \; 0 - \frac{d}{dt}\left[\frac{i}{\hbar}\int_{t_o}^{t} dt_1 H_I'(t_1)\; U_I(t_1, t_o)\right] \quad . \tag{2.13}$$

Now, the derivative of an operator F is defined as

$$\frac{d}{dt}F(t) = \lim_{\delta t \to 0} \frac{F(t + \delta t) - F(t)}{\delta t} . \tag{2.14}$$

Let

$$F(t) = \int_{t_o}^{t} dt_1\, H_I'(t_1)\, U_I(t_1, t_o) \;;$$

$$F(t + \delta t) = \int_{t_o}^{t+\delta t} dt_1\, H_I'(t_1)\, U_I(t_1, t_o) \tag{2.15}$$

$$= \int_{t_o}^{t} dt_1\, H_I'(t_1)\, U_I(t_1, t_o) + \int_{t}^{t+\delta t} dt_1\, H_I'(t_1)\, U_I(t_1, t_o)$$

Substituting from Equation (2.15) into Equation (2.14)

$$\frac{d}{dt}\int_{t_o}^{t} dt_1\, H_I'(t_1)\, U_I(t_1, t_o) = \lim_{\delta t \to 0}\left\{\int_{t}^{t+\delta t} dt_1\, H_I'(t_1)\, U_I(t_1, t_o)/\delta t\right\}$$

$$= \lim_{\delta t \to 0}\left\{H_I'(t)\, U_I(t, t_o)\delta t/\delta t\right\} = H_I'(t)\, U_I(t, t_o) . \tag{2.16}$$

Substituting Equation (2.16) into Equation (2.13), we get

$$\frac{d}{dt}U_I(t, t_o) = \frac{-i}{\hbar} H_I'(t)\, U_I(t, t_o)$$

or

$$i\hbar \frac{d}{dt}U_I(t, t_o) = H_I'(t)\, U_I(t, t_o) ,$$

which is the same as Equation (2.11). Thus we have proven that the differential Equation (2.11) along with Equation (2.14) is equivalent to the Equation (2.12).

A perturbation series for $U_I(t,t_o)$ may be obtained by substituting for $U_I(t_1,t_o)$ in Equation (2.12),

$$U_I(t,t_o) = \left[1 - \frac{i}{\hbar}\int_{t_o}^{t} dt_1 H_I'(t_1)\left\{1 - \frac{i}{\hbar}\int_{t_o}^{t_1} dt_2\, H_I'(t_2)\, U_I(t_2,t_o)\right\}\right] ,$$

and continuing the substitution for $U_I(t_n,t_o)$ in an iterative fashion, the result is that

$$U_I(t,t_o) = \left[1 + \left\{\sum_{n=1}^{\infty} \left(\frac{-i}{\hbar}\right)^n \int_{t_o}^{t} dt_1 \int_{t_o}^{t_1} dt_2 \int_{t_o}^{t_2} dt_3 \; \ldots\ldots \right.\right.$$

$$\left.\left. \ldots\ldots \int_{t_o}^{t_{n-1}} dt_n\, H_I'(t_1)\, H_I'(t_2)\, H_I'(t_3) \ldots\ldots H_I'(t_n)\right\}\right] \quad (2.17)$$

and

$$U_I(t_o,t_o) = 1 \quad . \qquad (2.18)$$

With the above background in mind, we can proceed to calculate the electric dipole transition probability between two states of the system. Let $\{|k>\}$ be a complete orthonormal basis set where k stands for a particular set of eigenvalues for a complete set of observables, among which is H_{os}. Consider two states $|a>$ and $|b>$ which are eigenstates of H_{os}. Then

$$\begin{aligned} H_{os}|a> &= E_a|a> \quad , \\ H_{os}|b> &= E_b|b> \quad , \end{aligned} \qquad (2.19)$$

or, in general, if F is a function of H_{os},

$$F(H_{os})|a\rangle = F(E_a)|a\rangle \; ,$$
$$F(H_{os})|b\rangle = F(E_b)|b\rangle \; . \qquad (2.20)$$

Now $U_I(t,t_o)|a\rangle$ is the state to which $|a\rangle$ would evolve in time interval $(t - t_o)$ in the Interaction picture. And the probability that the system would be found in the state $|b\rangle$ is

$$P_{b\leftarrow a}(t,t_o) = |\langle b|U_I(t,t_o)|a\rangle|^2$$
$$= |C_{b\leftarrow a}(t,t_o)|^2 \; , \qquad (2.21)$$

where

$$C_{b\leftarrow a}(t,t_o) = \langle b|U_I(t,t_o)|a\rangle \qquad (2.22)$$

= the probability amplitude in the Interaction picture.

Retaining only the first two terms in the perturbation expansion, Equation (2.17), of $U_I(t,t_o)$ in Equation (2.22) for the short time interval $(t-t_o)$, we get

$$\langle b|U_I(t,t_o)|a\rangle \approx \langle b|1 - \frac{i}{\hbar}\int_{t_o}^{t} dt_1 H_I'(t_1)\, U_I(t_1,t_o)|a\rangle \; .$$

Substituting for $H_I'(t_1)$ from correspondence relation, Equation (2.6), we find

$$P_{b\leftarrow a}(t,t_o) \simeq P^1_{b\leftarrow a}(t,t_o) = |C^1_{b\leftarrow a}(t,t_o)|^2 \; .$$

$$P^1_{b\leftarrow a}(t,t_o) = |\langle b|1 - \frac{i}{\hbar}\int_{t_o}^{t} dt_1\, e^{iH_{os}t_1/\hbar} H_s'(t_1)\, e^{-iH_{os}t_1/\hbar} |a\rangle|^2$$

$$= |-\frac{i}{\hbar}\int_{t_o}^{t} dt_1 \langle b| e^{iH_{os}t_1/\hbar} H_s'(t_1)\, e^{-iH_{os}t_1/\hbar} |a\rangle|^2 ,$$

and then applying the relation in Equation (2.20), we get

$$P^1_{b\leftarrow a}(t,t_o) = |\frac{-i}{\hbar}\int_{t_o}^{t} dt_1 \langle b| e^{iE_b t_1/\hbar} H_s'(t_1)\, e^{-iE_a t_1/\hbar} |a\rangle|^2 .$$

Thus, we get

$$P^1_{b\leftarrow a}(t,t_o) = |\frac{-i}{\hbar}\int_{t_o}^{t} dt_1\, e^{i(E_b-E_a)t_1/\hbar} \langle b|H_s'(t_1)|a\rangle|^2 . \qquad (2.23)$$

Before we evaluate the above time integral, let us look at $H_s'(t)$ from Equation (2.3). Let ψ be an arbitrary state of the system.

$$H_s'(t)\psi = \left\{ \sum_{j=1}^{N} \frac{ei\hbar}{2m}\left[\vec{\nabla}_j \cdot \vec{A}(\vec{r}_j,t_1) + \vec{A}(\vec{r}_j,t_1)\cdot\vec{\nabla}_j\right]\psi \right.$$

$$\left. + \sum_{j=1}^{N} \frac{e^2}{2m} A^2(\vec{r}_j,t_1)\psi + \sum_{j=1}^{N} e\,\emptyset(\vec{r}_j,t_1)\psi \right\}$$

Since

$$[\vec{\nabla}_j\cdot\vec{A} + \vec{A}\cdot\vec{\nabla}_j]\psi = \vec{\nabla}_j\cdot(\vec{A}\psi) + \vec{A}\cdot(\vec{\nabla}_j\psi)$$

$$= (\vec{\nabla}_j\cdot\vec{A})\psi + \vec{A}\cdot(\vec{\nabla}_j\psi) + \vec{A}\cdot(\vec{\nabla}_j\psi)$$

$$= (\vec{\nabla}_j\cdot\vec{A})\psi + 2\vec{A}\cdot(\vec{\nabla}_j\psi)$$

we get

$$H'_s(t_1)\psi = \left[\sum_{j=1}^{N} \frac{ei\hbar}{2m} (\vec{\nabla}_j \cdot \vec{A})\psi + \sum_{j=1}^{N} \frac{ei\hbar}{m} \vec{A} \cdot (\vec{\nabla}_j \psi) + \sum_{j=1}^{N} \frac{e^2}{2m} A^2 \psi + \sum_{j=1}^{N} e\,\phi\,\psi \right] .$$

In empty space, the charge and current densities are zero; so it is possible to use the Coulomb guage where $\vec{\nabla} \cdot \vec{A} = 0$ and since the charge density is zero, it follows that $\phi = 0$. Under these conditions,

$$H'_s(t_1)\psi = \left[\sum_{j=1}^{N} \frac{ei\hbar}{m} \left\{ \vec{A}(\vec{r}_j, t_1) \cdot (\vec{\nabla}_j \psi) \right\} + \sum_{j=1}^{N} \frac{e^2}{2m} A^2 (\vec{r}_j, t_1)\psi \right] . \tag{2.24}$$

In Equation (2.24) the term $(\frac{e^2}{2m} A^2)$ is generally very small and hence, we can omit it. Thus, we get

$$H'_s(t_1)\psi = \frac{ei\hbar}{m} \vec{A}(\vec{r}_j, t_1) \cdot \vec{\nabla}_j \psi .$$

Since ψ is an arbitrary state, the operator $H'_s(t)$ can be written simply as

$$H'_s(t_1) = \sum_{j=1}^{N} \frac{ei\hbar}{m} \left[\vec{A}(\vec{r}_j, t_1) \cdot \vec{\nabla}_j \right] . \tag{2.25}$$

Let the perturbing field consist of a monochromatic linearly polarized plane wave. Since $\vec{\nabla} \times \vec{A} = \vec{\mathcal{H}}$, $\vec{A}$ is perpendicular to $\vec{\mathcal{H}}$, the magnetic field. Also, $\vec{A}$ is perpendicular to the propagation vector $\vec{k}$. Since $\vec{\mathcal{H}}$, $\vec{E}$ and $\vec{k}$ are mutually perpendicular to one another, it follows that $\vec{A}$ must be parallel to the electric vector $\vec{E}$ of the perturbing monochromatic radiation. Let $\hat{\epsilon}$ be a unit polarization vector parallel to $\vec{E}$. Then $\vec{A}(\vec{r}_j, t)$ can be represented as

$$\vec{A}_{\vec{k},\omega}(\vec{r}_j, t) = A\hat{\epsilon}\left[e^{i(\omega t-\vec{k}\cdot\vec{r}_j)} + e^{-i(\omega t-\vec{k}\cdot\vec{r}_j)}\right] . \qquad (2.26)$$

$\vec{k}$ and ω are, of course, related by the fact that, in free space, $\omega = |\vec{k}|c$ where c is the speed of light in vacuum. The magnitude of $\vec{k}$ is related to the wavelength λ of the monochromatic radiation by $|\vec{k}| = 2\pi/\lambda$. Then $H_s'(t_1)$ in Equation (2.25) becomes

$$H_s'(t_1) = \left[\frac{Aei\hbar}{m}\sum_{j=1}^{N}\left\{e^{i(\omega t_1-\vec{k}\cdot\vec{r}_j)}\,\hat{\epsilon}\cdot\vec{\nabla}_j + e^{-i(\omega t_1-\vec{k}\cdot\vec{r}_j)}\,\hat{\epsilon}\cdot\vec{\nabla}_j\right\}\right]$$

$$= \frac{Aei\hbar}{m}\sum_{j=1}^{N}\hat{\epsilon}\cdot\vec{\nabla}_j\left[e^{i(\omega t_1-\vec{k}\cdot\vec{r}_j)} + e^{-i(\omega t_1-\vec{k}\cdot\vec{r}_j)}\right] . \qquad (2.27)$$

Substituting $H_s'(t_1)$ from Equation (2.27) into Equation (2.23) for the probability, we get

$$P^1_{b\leftarrow a}(t,t_o) = \left|\frac{-i}{\hbar}\int_{t_o}^{t} dt_1 e^{i(E_b-E_a)t_1/\hbar}\cdot\frac{Aei\hbar}{m}\cdot\right.$$

$$\left.\left[\langle b|\sum_{j=1}^{N}\left\{e^{i(\omega t_1-\vec{k}\cdot\vec{r}_j)} + e^{-i(\omega t_1-\vec{k}\cdot\vec{r}_j)}\right\}\hat{\epsilon}\cdot\vec{\nabla}_j|a\rangle\right]\right|^2$$

$$= \left|\left[\frac{Ae}{m}\int_{t_o}^{t} dt_1 e^{i(E_b-E_a+\hbar\omega)t_1/\hbar}\cdot\langle b|\sum_{j=1}^{N}e^{-i\vec{k}\cdot\vec{r}_j}\hat{\epsilon}\cdot\vec{\nabla}_j|a\rangle\right.\right.$$

$$\left.\left.+\frac{Ae}{m}\int_{t_o}^{t} dt_1 e^{i(E_b-E_a-\hbar\omega)t_1/\hbar}\cdot\langle b|\sum_{j=1}^{N}e^{i\vec{k}\cdot\vec{r}_j}\hat{\epsilon}\cdot\vec{\nabla}_j|a\rangle\right]\right|^2 .$$

We consider our electromagnetic radiation to be monochromatic with electric vector $\vec{E}(x, t)$ and magnetic vector $\vec{\mathcal{H}}(x, t)$. Let us next consider a case where the wavelength λ of the radiation is much greater than the size a of the system (6). If the center of the system is taken as an origin of a coordinate system, then the phase of the wave as it sweeps over the system will change by $(\pm\ 2\pi a/\lambda)$. Since the dimensions of the system are much smaller than the wavelength, the phase change within the system can be overlooked and the field within the system can be assumed to be the same at all the points. The states $|a>$ and $|b>$ are eigenkets of the unperturbed electronic Hamiltonian H_{os} and correspond to bound states of N-electrons (we shall not consider polarization process in this thesis). Thus, the wave functions

$$<\vec{r}_1 \xi_1 \vec{r}_2 \xi_2 \ldots\ldots \vec{r}_N \xi_N|a> \; = \; \psi_a(\vec{r}_1 \xi_1 \vec{r}_2 \xi_2 \ldots\ldots \vec{r}_N \xi_N)$$

and

$$<\vec{r}_1 \xi_1 \vec{r}_2 \xi_2 \ldots\ldots \vec{r}_N \xi_N|b> \; = \; \psi_b(\vec{r}_1 \xi_1 \vec{r}_2 \xi_2 \ldots\ldots \vec{r}_N \xi_N)$$

for these bound states vanish exponentially as any electronic coordinate $\vec{r}_j$ is taken far from the fixed nuclear framework, and contribution to the matrix elements

$$<b|\sum_j e^{\pm i\vec{k}\cdot\vec{r}_j}\hat{\epsilon}\cdot\vec{\nabla}_j|a>$$

from these regions of configuration space are, therefore, insignificant. The major contribution to the matrix elements comes from those regions of configuration space where the wave functions are large, i.e., the wave functions are quite appreciable for the electrons lying within the characteristic size "a" of the system. Since we are considering the total system in which $\lambda \gg a \geq r_j$, the magnitude of the term $(i\vec{k}\cdot\vec{r}_j)$ is always much smaller than unity. In the exponent,

$$|i\vec{k}\cdot\vec{r}_j| < \frac{2\pi a}{\lambda} \quad ,$$

we need to retain only the first term in the expansion of $e^{i\vec{k}\cdot\vec{r}_j}$.

$$\begin{aligned} e^{\pm i\vec{k}\circ\vec{r}_j} &= 1 \pm i\vec{k}\circ\vec{r}_j + \ldots\ldots \\ &\approx 1 \end{aligned}$$

This corresponds to the electric dipole term.

Thus to the approximations discussed above, the transition probability becomes

$$P^1_{b\leftarrow a}(t,t_o) = \left| \left[\frac{Ae}{m} \langle b| \sum_{j=1}^{N} \hat{\varepsilon}\cdot\vec{\nabla}_j |a\rangle \cdot \int_{t_o}^{t} dt_1 \left\{ e^{i(E_b-E_a+\hbar\omega)t_1/\hbar} + e^{i(E_b-E_a-\hbar\omega)t_1/\hbar} \right\} \right] \right|^2 .$$

Let $t_o = 0$ and t a short time later at which we are calculating the probability of transition (7). Then

$$P^1_{b\leftarrow a}(t,0) = \left| \left[\frac{Ae}{m} \langle b| \sum_{j=1}^{N} \hat{\varepsilon}\cdot\vec{\nabla}_j |a\rangle \cdot \frac{\hbar}{i} \left\{ \frac{e^{i(E_b-E_a+\hbar\omega)t/\hbar} - 1}{E_b-E_a+\hbar\omega} + \frac{e^{i(E_b-E_a-\hbar\omega)t/\hbar} - 1}{E_b-E_a-\hbar\omega} \right\} \right] \right|^2 .$$

Now the transition probability is

$$
\begin{aligned}
P^1_{b\leftarrow a}(t,0) = \Bigg[& \frac{A^2 e^2 \hbar^2}{m^2} \left| \langle b \right| \sum_{j=1}^{N} \hat{\varepsilon} \cdot \vec{\nabla}_j \left| a \rangle \right|^2 \cdot \\
& \Bigg\{ \frac{2-2\cos[(E_b-E_a+\hbar\omega)t/\hbar]}{(E_b-E_a+\hbar\omega)^2} + \frac{2-2\cos[(E_b-E_a-\hbar\omega)t/\hbar]}{(E_b-E_a-\hbar\omega)^2} \\
& + \frac{2+2\cos 2\omega t - 2\cos[(E_b-E_a+\hbar\omega)t/\hbar]}{(E_b-E_a)^2-(\hbar\omega)^2} \\
& - \frac{2\cos[(E_b-E_a-\hbar\omega)t/\hbar]}{(E_b-E_a)^2-(\hbar\omega)^2} \Bigg\} \Bigg] \quad . \qquad (2.28)
\end{aligned}
$$

The formula given in Equation (2.28) is derived on the basis that the electromagnetic radiation is monochromatic. In practice, however, even the best monochromatic radiation includes a narrow band of wavelength (i.e., frequency). Let our radiation cover a narrow frequency band ranging from ω_1 to ω_2 including the frequency $(E_b - E_a)/\hbar = \omega_o$. In this case A^2 can be replaced by $8\pi I$ where $I(\omega)$ is the energy density of radiation per unit frequency interval. Then the transition probability $P^1_{b\leftarrow a}(t,0)$ is obtained by integrating Equation (2.28) from ω_1 to ω_2 and neglecting small terms, we get

$$P_{b\leftarrow a}(t,0) = \int_{\omega_1}^{\omega_2} P^1_{b\leftarrow a}(t,0)\,d\omega$$

$$= \Bigg[\frac{16\,e^2\hbar^2 I(\omega_o)}{m^2}\,\Big|\langle b\Big|\sum_{j=1}^{N}\hat{\varepsilon}\cdot\vec{\nabla}_j\Big|a\rangle\Big|^2 \cdot$$

$$\int_{\omega_1}^{\omega_2}\frac{1-\cos\{(|E_b-E_a|-\hbar\omega)t/\hbar\}}{(|E_b-E_a|-\hbar\omega)^2}\,d\omega\Bigg]$$

$$= \Bigg[\frac{16\,e^2\hbar^2 I}{m^2}\,\Big|\langle b\Big|\sum_{j=1}^{N}\hat{\varepsilon}\cdot\vec{\nabla}_j\Big|a\rangle\Big|^2 \cdot \int_{\xi(\omega_1)}^{\xi(\omega_2)} d\xi\left(\frac{1-\cos\xi t/\hbar}{\xi^2}\right) \qquad (2.29)$$

where

$$\xi = [(|E_b-E_a| - \hbar\omega)] \quad .$$

If now the frequency band is wide compared to the natural width of the emission line, no generality is lost if $\xi(\omega_1)$ is replaced by $(-\infty)$ and $\xi(\omega_2)$ by $(+\infty)$. Integration gives

$$P_{b\leftarrow a}(t,0) = \frac{8\,e^2\hbar^2 It}{m^2}\,\Big|\langle b\Big|\sum_{j=1}^{N}\hat{\varepsilon}\cdot\vec{\nabla}_j\Big|a\rangle\Big|^2$$

and the transition probability rate is

$$R_{b\leftarrow a} = \frac{8\,e^2\hbar^2 I}{m^2}\,\Big|\langle b\Big|\sum_{j=1}^{N}\hat{\varepsilon}\cdot\vec{\nabla}_j\Big|a\rangle\Big|^2 \quad . \qquad (2.30)$$

In the transition phenomena, absorption takes place if $E_b > E_a$ and emission takes place if $E_a > E_b$.

The matrix element can be reexpressed in terms of other operators and so the electric dipole transition probability can be shown also to be proportional to

$$R_{b\leftarrow a} \propto \left|\langle \Psi_f | \sum_{j=1}^{N} \vec{r}_j | \Psi_i \rangle\right|^2$$

$$\propto \left|\text{dipole length moment}\right|^2 \quad ,$$

and

$$R_{b\leftarrow a} \propto \left|\langle \Psi_f | \sum_{j=1}^{N} \vec{a}_j | \Psi_i \rangle\right|^2$$

$$\propto \left|\text{dipole acceleration moment}\right|^2 \quad .$$

The wavelength λ enters these alternate formulae in various ways. This is proven in the following section.

We will show the equivalence of

(1) Dipole Length Moment $S_R = \left|\langle \Psi_f | \sum_{j=1}^{N} \vec{r}_j \, | \Psi_i \rangle\right|^2$,

(2) Dipole Velocity Moment $S_\nabla = \left|\langle \Psi_f | \sum_{j=1}^{N} \vec{\nabla}_j | \Psi_i \rangle\right|^2$

where Ψ_i and Ψ_f are the eigenfunctions of the Hamiltonian of the system. The electronic Hamiltonian of the system is

$$H_{os} = \left[\sum_{j=1}^{N} \frac{-\hbar^2}{2m} \nabla_j^2 + \sum_{j=1}^{N} \sum_{\alpha=1}^{A} \frac{-Z_\alpha e^2}{r_{j\alpha}} + \sum_{1\le j<k}^{N} \frac{e^2}{r_{jk}} \right] \quad . \tag{2.31}$$

That is

$$H_{os} = T + V \quad , \tag{2.32}$$

where

$$T = \sum_{j=1}^{N} \frac{-\hbar^2}{2m} \nabla_j^2 \quad . \tag{2.33}$$

and

$$V = \sum_{j=1}^{N} \sum_{\alpha=1}^{A} \frac{-Z_\alpha e^2}{r_{j\alpha}} + \sum_{1 \le j < k}^{N} \frac{e^2}{r_{jk}} \quad . \tag{2.34}$$

Since V and $\vec{r}_j$ commute,

$$\left[V, \sum_{j=1}^{N} \vec{r}_j\right] = 0 \quad , \tag{2.35}$$

and the commutator

$$\begin{aligned} \left[H_{os}, \sum_j \vec{r}_j\right] &= \left[(T+V), \sum_j \vec{r}_j\right] \\ &= \left[T, \sum_j \vec{r}_j\right] + \left[V, \sum_j \vec{r}_j\right] \\ &= \left[T, \sum_j \vec{r}_j\right] \\ &= -\frac{\hbar^2}{2m}\left[\sum_{k=1}^{N} \nabla_k^2 \,, \sum_{j=1}^{N} \vec{r}_j\right] \quad . \end{aligned} \tag{2.36}$$

We have that

$$\begin{aligned} \left[H_{os}, \sum_j \vec{r}_j\right] &= \left[T, \sum_j \vec{r}_j\right] \\ &= -\frac{\hbar^2}{2m} \sum_{k=1}^{N} \sum_{j=1}^{N} [\nabla_k^2, \vec{r}_j] \quad . \end{aligned} \tag{2.37}$$

Consider $[\nabla_k^2, \vec{r}_j]\, f\}$ component by component where $f = f(\vec{r}_1, \vec{r}_2 \ldots \vec{r}_N)$ is an arbitrary wave function.

The x component, for example, is

$$\begin{aligned}
[\nabla_k^2, x_j]f &= (\nabla_k^2 x_j - x_j \nabla_k^2)f \\
&= \nabla_k^2(x_j f) - x_j(\nabla_k^2 f) \\
&= \vec{\nabla}_k \cdot \vec{\nabla}_k(x_j f) - x_j \nabla_k^2 f \\
&= \vec{\nabla}_k \cdot [(\vec{\nabla}_k x_j)f + x_j(\vec{\nabla}_k f)] - x_j \nabla_k^2 f \\
&= (\nabla_k^2 x_j)f + \vec{\nabla}_k x_j \cdot \vec{\nabla}_k f + \vec{\nabla}_k x_j \cdot \vec{\nabla}_k f \\
&= 2\, \vec{\nabla}_k x_j \cdot \vec{\nabla}_k f + (\nabla_k^2 x_j)f \ ,
\end{aligned}$$

that is,

$$[\nabla_k^2, \vec{r}_j]_x\, f = 2\vec{\nabla}_k x_j \cdot \vec{\nabla}_k f + (\nabla_k^2 x_j)f \quad . \tag{2.38}$$

Now, we see that

$$\begin{aligned}
\vec{\nabla}_k x_j &= \frac{\partial x_j}{\partial x_k}\hat{x} + \frac{\partial x_j}{\partial y_k}\hat{y} + \frac{\partial x_j}{\partial z_k}\hat{z} \\
&= \delta_{jk}\hat{x} + 0 + 0 \\
&= \delta_{jk}\hat{x} \qquad ,
\end{aligned} \tag{2.39a}$$

and since

$$\vec{\nabla}_k f = \frac{\partial f}{\partial x_k}\hat{x} + \frac{\partial f}{\partial y_k}\hat{y} + \frac{\partial f}{\partial z_k}\hat{z} \qquad ,$$

we conclude that

$$\vec{\nabla}_k x_j \cdot \vec{\nabla}_k f = \frac{\partial f}{\partial x_k}\delta_{jk}[\hat{x}\cdot\hat{x}] + 0 + 0 = \frac{\partial f}{\partial x_k}\delta_{jk} \quad . \tag{2.39b}$$

Furthermore,

$$\nabla_k^2 x_j = \frac{\partial^2 x_j}{\partial x_k^2} + \frac{\partial^2 x_j}{\partial y_k^2} + \frac{\partial^2 x_j}{\partial z_k^2} = 0 \quad . \tag{2.39c}$$

Thus, using Equation (2.39) into Equation (2.38), we get

$$[\nabla_k^2, \vec{r}_j]_x f = 2\,\delta_{jk} \frac{\partial f}{\partial x_k} \quad . \tag{2.40a}$$

Similarly, we find

$$[\nabla_k^2, \vec{r}_j]_y f = 2\,\delta_{jk} \frac{\partial f}{\partial y_k} \tag{2.40b}$$

and

$$[\nabla_k^2, \vec{r}_j]_z f = 2\,\delta_{jk} \frac{\partial f}{\partial z_k} \tag{2.40c}$$

or writing these three equations in vector notation,

$$[\nabla_k^2, \vec{r}_j] f = 2\,\delta_{jk} \vec{\nabla}_k f \quad .$$

Since f is arbitrary, it follows that

$$[\nabla_k^2, \vec{r}_j] = 2\,\delta_{jk} \vec{\nabla}_k \tag{2.41}$$

and

$$\begin{aligned} [T, \sum_j \vec{r}_j] &= \frac{-\hbar^2}{2m} \sum_j \sum_k [\nabla_k^2, \vec{r}_j] \\ &= -\frac{\hbar^2}{2m} \sum_j \sum_k 2\,\delta_{jk} \vec{\nabla}_k \\ &= \frac{(i\hbar)(i\hbar)}{m} \sum_j \vec{\nabla}_j = \frac{-i\hbar}{m} \sum_j \vec{P}_j \quad . \end{aligned}$$

Thus we get

$$[T, \sum_j \vec{r}_j] = [H_{os}, \sum_j \vec{r}_j] = \frac{-i\hbar}{m} \sum_j \vec{P}_j \quad . \tag{2.42}$$

Since Ψ_i and Ψ_f are the eigenfunctions of the Hamiltonian H_{os}

$$H_{os} | \Psi_i > = E_i | \Psi_i > \quad ,$$

$$H_{os} | \Psi_f > = E_f | \Psi_f > \quad . \tag{2.43}$$

Then,

$$< \Psi_f | [H_{os}, \sum_j \vec{r}_j] | \Psi_i > = < \Psi_f | H_{os} (\sum_j \vec{r}_j) - (\sum_j \vec{r}_j) H_{os} | \Psi_i > .$$

Since H_{os} is hermitian, $H^+_{os} = H_{os}$

$$= < \Psi_f | H^+_{os} (\sum_j \vec{r}_j) - (\sum_j \vec{r}_j) H_{os} | \Psi_i >$$

$$= < \Psi_f | E_f (\sum_j \vec{r}_j) - (\sum_j \vec{r}_j) E_i | \Psi_i >$$

$$= (E_f - E_i) < \Psi_f | \sum_j \vec{r}_j | \Psi_i >$$

i.e.,

$$\frac{-i\hbar}{m} < \Psi_f | \sum_j \vec{P}_j | \Psi_i > = (E_f - E_i) < \Psi_f | \sum_j \vec{r}_j | \Psi_i > \quad . \tag{2.44}$$

There is a third formulation in terms of dipole acceleration operator S_A ($S_A = |< \Psi_f | \sum_j \vec{a}_j | \Psi_i > |^2$). This formulation can also be proven equivalent to dipole length or dipole velocity.

The above proof requires that the wave functions $| \Psi_i >$ and $| \Psi_f >$ be exact eigenfunctions of the Hamiltonian. Since the eigenfunctions obtained in our calculations are only approximate eigenstates of the Hamiltonian, the numerical results obtained for the transition probability using the three equivalent expressions, in general, show discrepancies. However, neither dipole length nor dipole velocity nor dipole acceleration formula is more fundamental (8). Each, in fact, has a classical analogue and we should consider the results obtained with S_R, S_∇ and S_A as having equal significance. In practice, the divergence in their results must be expected because the three operators emphasize different spatial portions of the wave functions in their integral process. Therefore, different credence must be given to the results obtained with different operators depending upon the accuracy of these wave functions in the regions where they are most sensitive to the particular operator. The velocity and length operators give more emphasis to the regions of the wave functions close to the atomic shells where the charge density of the reshuffling electrons is large. The dipole velocity and the dipole length formulae, respectively, favor the inner and outer portions of the shell. The acceleration operator gives more stress to the spatial region of the wave function close to the nuclei since it contains terms behaving as r^{-3}. In this region, our electron wave functions, computed with Gaussian orbitals, are generally the poorest, but since S_A is most sensitive to the accuracy of the wave function in the immediate neighborhood of the nuclei, discrepancies in the results obtained with S_A would be large. Hence, we do not attempt to calculate S_A.

CHAPTER III

CALCULATIONS OF APPROXIMATE WAVE FUNCTIONS

In our calculations we have used the Hartree-Fock method to evaluate the best possible orbital functions. The quantum-mechanical Hartree-Fock treatment of a complex system (atom or molecule) assumes an independent particle model, i.e., in an N-particle system each occupies a definite single particle state and moves independently of the presence of others in a potential representing the attraction of nucleus and the average repulsive effect of the remaining particles (5). The single particle energy operator then should be the sum of bare nucleus Hamiltonian and the potential interaction operator which depends upon the states occupied by the rest of the particles. In this approximation, the wave function of the system is represented by a single Slater determinant made up from an antisymmetrized product of symmetry-adapted occupied spin orbitals. The best choice of the orbitals is, in fact, obtained self-consistently by the variational method using an arbitrary Slater determinant as a trial function (10). The two conditions, namely

1. the spin functions of different spins are orthogonal and

2. the spatial parts of the spin orbitals are orthonormal

imply that this symmetry-adapted spin orbital basis set is orthonormal (11). Most generally, we have the spin orbital

$$\Gamma_{\underset{\sim}{\mu}}(x) = \Phi_{\underset{\sim}{\mu}\frac{1}{2}}(\vec{r})\,\lambda_{\frac{1}{2}}(\xi) + \Phi_{\underset{\sim}{\mu}-\frac{1}{2}}(\vec{r})\,\lambda_{-\frac{1}{2}}(\xi) \tag{3.1a}$$

where

$$\sum_{\xi} \lambda^{*}_{m_s}(\xi)\,\lambda_{m_s'}(\xi) = \delta_{m_s m_s'} \tag{3.1b}$$

and

$$\sum_{m_s=-\frac{1}{2}}^{\frac{1}{2}} \int \Phi^{*}_{\underset{\sim}{\mu} m_s}(\vec{r})\,\Phi_{\underset{\sim}{\mu}' m_s}(\vec{r})\,d\vec{r} = \delta_{\underset{\sim}{\mu}\underset{\sim}{\mu}'} \quad . \tag{3.1c}$$

In the usual notation $\alpha(\xi) = \lambda_{\frac{1}{2}}(\xi)$ and $\beta(\xi) = \lambda_{-\frac{1}{2}}(\xi)$. Hence,

$$\int \Gamma^{*}_{\underset{\sim}{\mu}}(x)\,\Gamma_{\underset{\sim}{\mu}'}(x)\,dx = \delta_{\underset{\sim}{\mu}\underset{\sim}{\mu}'} \quad . \tag{3.2}$$

The integration is carried out over both the spin variable and the space variable. The orthonormal symmetry-adapted spin orbital basis set assures the orthonormality of the Slater determinant constructed from the former. Thus,

$$< \Delta | \Delta > = 1 \tag{3.3}$$

where

$$\Delta = \mathcal{A}\{\Gamma_1(x_1)\,\Gamma_2(x_2)\,\ldots\,\Gamma_N(x_N)\} \quad . \tag{3.4}$$

The Hamiltonian of the N-particle system is

$$H_{os} = \sum_{j=1}^{N} H^{bare}(j) + \sum_{1\leq j<k}^{N} \frac{e^2}{|\vec{r}_j - \vec{r}_k|} \tag{3.5}$$

where

$$\sum_{j=1}^{N} H^{\text{bare}}(j) = \sum_{j=1}^{N} \left[-\frac{\hbar^2}{2m} \nabla_j^2 + \sum_{\alpha=1}^{A} \frac{-Z_\alpha e^2}{|\vec{r}_j - \vec{R}_\alpha|} \right] \tag{3.6}$$

= sum of bare nucleus Hamiltonian

and "A" is the total number of nuclei.

$$\sum_{1 \le j < k}^{N} \frac{e^2}{|\vec{r}_j - \vec{r}_k|} = \sum_{1 \le j < k}^{N} \frac{e^2}{r_{jk}}$$

= interelectronic repulsion.

In the Hartree-Fock method the expectation value of Hamiltonian is minimized, i.e., $E = \langle \Delta | H_{os} | \Delta \rangle$ is minimized subject to the constraint in Equation (3.2) (12). It can be shown that the variational procedure implies that the best (i.e., Hartree-Fock) orbitals satisfy a quasi-eigenvalue equation

$$H^{HF}(x_1)\, \Gamma_{\underset{\sim}{\mu}}(x_1) = \epsilon_{\underset{\sim}{\mu}}\, \Gamma_{\underset{\sim}{\mu}}(x_1) \tag{3.7}$$

where the Hartree-Fock operator

$$H^{HF}(1) = H^{\text{bare}}(1) + V^{HF}(1) \tag{3.8}$$

contains the non-local HF potential

$$V^{HF}(1) = \sum_{\underset{\sim}{\mu}=1}^{N} \int dx_2\, \Gamma^*_{\underset{\sim}{\mu}}(x_2) \frac{(1 - P_{12})}{r_{12}} \Gamma_{\underset{\sim}{\mu}}(x_2) \quad . \tag{3.9}$$

The sum in Equation (3.9) is over the orbitals occupied in the HF determinant; these are the first N solutions to Equation (3.7). Though we were originally interested in finding just the first N (occupied) orbitals, we recognize that formally $(H^{HF})^{+} = H^{HF}$ should have a complete set of solutions. The unoccupied or so-called "virtual" orbitals are labeled $\Gamma_{\underset{\sim}{\mu}}$, $\underset{\sim}{\mu} = N+1, N+2, \ldots$, and we have implicitly assumed that the complete set of solutions is denumerable. We have termed the problem posed in Equation (3.7) a "quasi-eigenvalue" problem since the operator V^{HF}, and thus H^{HF}, explicitly depend on the first N solutions. The equation must be solved "self-consistently", i.e., in an iterative fashion with the first N solutions of one iteration used to construct V^{HF} for the next iteration.

Let us prepare a background before we go to the main problem. We assume that we deal with systems with closed shell ground state for which $\Gamma_{\underset{\sim}{\mu}}(x)$, the symmetry-adapted spin orbitals, can be written as a product of a spatial and a spin function. In particular we assume that the Hartree-Fock determinant is composed of (N/2) doubly occupied spatial orbitals. Thus,

$$\begin{aligned}\Gamma_{\underset{\sim}{\mu}}(x) &= \Phi_{\mu}(\vec{r})\,\lambda_{m_s}(\xi)\\ &= \Phi_{\mu}(\vec{r})\,\alpha(\xi) \quad \text{or} \quad \Phi_{\mu}(\vec{r})\,\beta(\xi)\end{aligned}$$

where for the occupied orbitals

$$\underset{\sim}{\mu} = 1, 2, 3, \ldots, N \quad ,$$

$$\mu = 1, 2, 3, \ldots, N/2 \quad .$$

We now expand the solutions

$$\Gamma_{\underset{\sim}{\mu}}(\vec{r}, \xi) = \sum_{j=1}^{\infty} \sum_{m_s=-\frac{1}{2}}^{\frac{1}{2}} b_{jm_s;\underset{\sim}{\mu}}\, \chi_j(\vec{r})\, \lambda_{m_s}(\xi) \qquad (3.10)$$

where $\chi_j(\vec{r})$ is in principle a complete normalized but not necessarily an orthogonal set of spatial orbitals. The Hartree-Fock equation for orbital $\Gamma_{\underset{\sim}{\mu}}$ can be written as

$$\sum_{j,m_s} \left[H^{HF} - \epsilon_{\underset{\sim}{\mu}}\right] b_{jm_s;\underset{\sim}{\mu}}\, \chi_j(\vec{r})\, \lambda_{ms}(\xi) = 0 \tag{3.11}$$

Multiplying Equation (3.11) by $\{\chi_i^*(\vec{r})\, \lambda^*_{m_s'}(\xi)\}$ and integrating over the space and spin coordinates, we get

$$\sum_{j,m_s} \left[H^{HF}_{im_s',jm_s} - \epsilon_{\underset{\sim}{\mu}} S_{im_s',jm_s}\right] b_{jm_s;\underset{\sim}{\mu}} = 0 \tag{3.12}$$

for $i = 1, 2, \ldots, \infty$, $m_s' = -\frac{1}{2}, \frac{1}{2}$.

Equivalently in matrix form, the $\underset{\sim}{\mu}$th solution to the Hartree-Fock equation satisfies

$$\left[\underset{\sim}{H}^{HF} - \epsilon_{\underset{\sim}{\mu}} \underset{\sim}{S}\right] \underset{\sim}{b}_{\underset{\sim}{\mu}} = 0 \tag{3.13}$$

where the matrix elements are

$$H^{HF}_{im_s',jm_s} = \int d\vec{r} \sum_{\xi} \chi_i^*(\vec{r})\, \lambda^*_{m_s'}(\xi)\, H^{HF} \chi_j(\vec{r})\lambda_{m_s}(\xi) \tag{3.14}$$

and

$$S_{im_s',jm_s} = \int d\vec{r} \sum_{\xi} \chi_i^*(\vec{r})\, \lambda_{m_s'}(\xi)\, \chi_j(\vec{r})\, \lambda_{m_s}(\xi) \quad . \tag{3.15}$$

It is obvious that $\underset{\sim}{S}$ is diagonal in the spin indices and a little effort shows that $\underset{\sim}{H}^{HF}$ is also diagonal in the spin indices. Thus,

$$S_{im_s',jm_s} = \int \chi_i^*(\vec{r})\, \chi_j(\vec{r})\, d\vec{r}\ \delta_{m_s'm_s} \tag{3.16}$$

and

$$H^{HF}_{im_s', jm_s} = \begin{cases} H^{HF}_{i\frac{1}{2}, j\frac{1}{2}} = H^{HF}_{i-\frac{1}{2}, j-\frac{1}{2}} & \text{for } m_s = m_s' \\ 0 & \text{for } m_s' \neq m_s \end{cases} \quad . \tag{3.17}$$

We can therefore simplify the matrix indexing and define

$$\begin{aligned} \overline{S}_{ij} &= \int \chi_i^*(\vec{r})\, \chi_j(\vec{r})\, d\vec{r} \quad , \\ \overline{H}_{ij} &= H^{HF}_{i\frac{1}{2}, j\frac{1}{2}} = H^{HF}_{i-\frac{1}{2}, j-\frac{1}{2}} \quad , \\ b_{j\mu} &= b_{j\frac{1}{2}, \mu\frac{1}{2}} = b_{j-\frac{1}{2}, \mu-\frac{1}{2}} \quad . \end{aligned} \tag{3.18}$$

We can rewrite the Equation (3.12) in explicit block diagonal form

$$\left\{ \left[\begin{array}{ccc|ccc} \overline{H}_{11} & \overline{H}_{12} & \cdots & & & \\ \overline{H}_{21} & \overline{H}_{22} & & & 0 & \\ \vdots & & \ddots & & & \\ \hline & & & \overline{H}_{11} & \overline{H}_{12} & \cdots \\ & 0 & & \overline{H}_{21} & \overline{H}_{22} & \\ & & & \vdots & & \ddots \end{array} \right] - \epsilon_{\underset{\sim}{\mu}} \left[\begin{array}{ccc|ccc} \overline{S}_{11} & \overline{S}_{12} & \cdots & & & \\ \overline{S}_{21} & \overline{S}_{22} & & & 0 & \\ \vdots & & \ddots & & & \\ \hline & & & \overline{S}_{11} & \overline{S}_{12} & \cdots \\ & 0 & & \overline{S}_{21} & \overline{S}_{22} & \\ & & & \vdots & & \ddots \end{array} \right] \right\} b_{\underset{\sim}{\mu}} = 0 \tag{3.19}$$

Clearly the solutions are spin degenerate, and the Hartree-Fock eigenvalues are the roots of the characteristic Equation (3.20).

$$\det \left[\underset{\sim}{\mathcal{H}}^{HF} - \epsilon_{\underset{\sim}{\mu}} \underset{\sim}{\mathcal{S}}^{HF} \right] = 0 \tag{3.20}$$

or explicitly noting the identity of the diagonal blocks, the Hartree-Fock eigenvalues (now noted to be doubly degenerate) are found as the roots of

$$\det[\underset{\sim}{\overline{H}} - \varepsilon_{\underset{\sim}{\mu}} \underset{\sim}{\overline{S}}] = 0 \quad . \tag{3.21}$$

Once the roots are obtained they are substituted into Equation (3.12), the solution of which for given $\varepsilon_{\underset{\sim}{\mu}}$ yields the corresponding $b_{\underset{\sim}{\mu}}$'s, one for the α-spin and one for the β-spin.

To solve Equation (3.21) we need to know

$$\begin{aligned}\overline{H}_{ij} &= \langle \chi_i(\vec{r}_1)\alpha(\xi_1)|H(1)|\chi_j(\vec{r}_1)\alpha(\xi_1)\rangle \\ &= \int d\vec{r}_1 \sum_{\xi_1} \alpha^*(\xi_1)\chi_i^*(\vec{r}_1)[H^{\text{bare}}(1) + V^{HF}(1)]\alpha(\xi_1)\chi_j(\vec{r}_1) \ .\end{aligned}$$

Since $H^{\text{bare}}(1)$ does not depend upon spin

$$\begin{aligned}\overline{H}_{ij} &= \int d\vec{r}_1 \chi_i^*(\vec{r}_1) H^{\text{bare}}(1) \chi_j(\vec{r}_1) \sum_{\xi_1} \alpha^*(\xi_1)\alpha(\xi_1) \\ &\quad + \int d\vec{r}_1 \sum_{\xi_1} \chi_i^*(\vec{r}_1)\alpha^*(\xi_1)\, V^{HF}(1)\chi_j(\vec{r}_1)\alpha(\xi_1) \\ &= \int d\vec{r}_1 \chi_i^*(\vec{r}_1)\, H^{\text{bare}}(1)\chi_j(\vec{r}_1) \\ &\quad + \int d\vec{r}_1 \sum_{\xi_1} \chi_i^*(\vec{r}_1)\alpha^*(\xi_1)\, V^{HF}(1)\chi_j(\vec{r}_1)\alpha(\xi_1) \\ &= H_{ij}^{\text{bare}} + \int d\vec{r}_1 \sum_{\xi_1} \chi_i^*(\vec{r}_1)\alpha^*(\xi_1)\, V^{HF}(1)\chi_j(\vec{r}_1)\alpha(\xi_1) \ .\end{aligned} \tag{3.22}$$

Let us evaluate the second term in Equation (3.22). Substituting for V^{HF}, we get

$$\begin{aligned} I = \Big[\int d\vec{r}_1 \sum_{\xi_1} \chi_i^*(1)\, \alpha^*(1) \sum_{\underset{\sim}{\mu}=1}^{N} \int d\vec{r}_2 \sum_{\xi_2} \Gamma_{\underset{\sim}{\mu}}^*(x_2) \cdot \\ \frac{(1 - P_{12})}{r_{12}}\, \Gamma_{\underset{\sim}{\mu}}(x_2)\, \chi_j(\vec{r}_1)\, \alpha(\xi_1) \Big] \end{aligned}$$

$$\begin{aligned} = \Big[\int d\vec{r}_1 \sum_{\xi_1} \sum_{\xi_2} \sum_{\mu=1}^{N/2} \int d\vec{r}_2\, \chi_i^*(1)\alpha^*(1) \Big\{ \Phi_\mu^*(\vec{r}_2)\alpha^*(\xi_2) \cdot \\ \frac{(1 - P_{12})}{r_{12}}\, \Phi_\mu(\vec{r}_2)\, \alpha(\xi_2) + \Phi_\mu^*(\vec{r}_2)\beta^*(\xi_2) \cdot \\ \frac{(1 - P_{12})}{r_{12}}\, \Phi_\mu(\vec{r}_2)\beta(\xi_2) \Big\}\, \chi_j(1)\alpha(1) \Big] \end{aligned}$$

$$\begin{aligned} = \Big[\int d\vec{r}_1 \sum_{\xi_1} \sum_{\xi_2} \sum_{\mu=1}^{N/2} \Big\{ \int d\vec{r}_2\, \chi_i^*(1)\alpha^*(1)\Phi_\mu^*(2)\alpha^*(2)\, \frac{1}{r_{12}} \cdot \\ \chi_j(1)\alpha(1)\Phi_\mu(2)\alpha(2) - \int d\vec{r}_2\, \chi_i^*(1)\alpha^*(1)\Phi_\mu^*(2)\alpha^*(2)\, \frac{1}{r_{12}} \cdot \\ \Phi_\mu(1)\alpha(1)\chi_j(2)\alpha(2) + \int d\vec{r}_2\, \chi_i^*(1)\alpha^*(1)\Phi_\mu^*(2)\beta^*(2)\, \frac{1}{r_{12}} \cdot \\ \chi_j(1)\alpha(1)\, \Phi_\mu(2)\beta(2) - \int d\vec{r}_2\, \chi_i^*(1)\alpha^*(1)\Phi_\mu^*(2)\beta^*(2)\, \frac{1}{r_{12}} \cdot \\ \Phi_\mu(1)\beta(1)\, \chi_j(2)\alpha(2) \Big\} \Big] \quad . \end{aligned}$$

$$\begin{aligned} I = \Big[2 \int d\vec{r}_1 \sum_{\mu=1}^{N/2} \int d\vec{r}_2\, \chi_i^*(1)\Phi_\mu^*(2)\, \frac{1}{r_{12}}\, \chi_j(1)\Phi_\mu(2) \\ - \int d\vec{r}_1 \sum_{\mu=1}^{N/2} \int d\vec{r}_2\, \chi_i^*(1)\Phi_\mu^*(2)\, \frac{1}{r_{12}}\, \Phi_\mu(1)\chi_j(2) \Big] \quad . \end{aligned}$$

Substituting for Φ_μ in terms of atomic orbitals $\{\chi_k(\vec{r})\}$,

$$I = \Big[\Big\{ 2 \sum_{\mu=1}^{N/2} \int d\vec{r}_1 d\vec{r}_2 \chi_i^*(1) \sum_{k=1}^{\infty} b_{k\mu}^* \chi_k^*(2) \frac{1}{r_{12}} \chi_j(1) \sum_{\ell=1}^{\infty} b_{\ell\mu} \chi_\ell(2) \Big\}$$

$$- \Big\{ \sum_{\mu=1}^{N/2} \int d\vec{r}_1 d\vec{r}_2 \chi_i^*(1) \sum_{k=1}^{\infty} b_{k\mu}^* \chi_k^*(2) \frac{1}{r_{12}} \sum_{\ell=1}^{\infty} b_{\ell\mu} \chi_\ell(1) \chi_j(2) \Big\} \Big]$$

$$= \Big[\sum_{k=1}^{\infty} \sum_{\ell=1}^{\infty} \Big(\sum_{\mu=1}^{N/2} b_{k\mu}^* b_{\ell\mu} \Big) \Big\{ 2 \int d\vec{r}_1 d\vec{r}_2 \chi_i^*(1) \chi_k^*(2) \frac{1}{r_{12}} \chi_j(1) \chi_\ell(2)$$

$$- \int d\vec{r}_1 d\vec{r}_2 \chi_i^*(1) \chi_k^*(2) \frac{1}{r_{12}} \chi_\ell(1) \chi_j(2) \Big\} \Big]$$

$$= 2 \sum_k \sum_\ell \rho_{k\ell} <ik|V|j\ell> - \sum_k \sum_\ell \rho_{k\ell} <ik|V|\ell j>$$

where

$$\sum_{\mu=1}^{N/2} b_{k\mu}^* b_{\ell\mu} = \rho_{k\ell}$$

and

$$\int d\vec{r}_1 d\vec{r}_2 \chi_i^*(1) \chi_k^*(2) \frac{1}{r_{12}} \chi_j(1) \chi_\ell(2) = <ik|V|j\ell> \quad .$$

Thus, Equation (3.22) can be rewritten as

$$<i|H^{HF}|j> = [<i|H^{bare}|j> + 2\sum_k \sum_\ell \rho_{k\ell} <ik|V|j\ell>$$

$$- \sum_k \sum_\ell \rho_{k\ell} <ik|V|\ell j>] \quad . \qquad (3.23)$$

Now, Equation (3.12) can be expressed as

$$[\overline{H}_{ij} - \epsilon_{\underset{\sim}{\mu}} \overline{S}_{ij}] \underset{\sim}{b} = 0 \quad . \qquad (3.24)$$

In practice the normalized atomic basis set is a truncated set. We shall use a set of M spatial Gaussian orbitals which are normalized, but, of course, not orthogonal. Needless to say, $M \geq N/2$. Thus we have a finite matrix diagonalization problem. We proceed as follows.

Since $\overline{S}^{+} = \overline{S}$, there exists a unitary matrix V which diagonalizes $\overline{S}$ and

$$\underset{\sim}{V}^{+}\overline{\underset{\sim}{S}}\,\underset{\sim}{V} = \begin{bmatrix} \lambda_1 & & & & \\ & \lambda_2 & & \mathcal{O} & \\ & & \lambda_3 & & \\ & \mathcal{O} & & \ddots & \\ & & & & \lambda_M \end{bmatrix} \tag{3.25}$$

with $\lambda_i > 0$, $i = 1, 2, \ldots, M$.

Define a matrix

$$\underset{\sim}{T} = \begin{bmatrix} \frac{1}{\sqrt{\lambda_1}} & & & & \\ & \frac{1}{\sqrt{\lambda_2}} & & \mathcal{O} & \\ & & \frac{1}{\sqrt{\lambda_3}} & & \\ & \mathcal{O} & & \ddots & \\ & & & & \frac{1}{\sqrt{\lambda_M}} \end{bmatrix} . \tag{3.26}$$

$(\underset{\sim}{V}\underset{\sim}{T})$ is not a unitary matrix in general, but

$$(\underset{\sim}{V}\underset{\sim}{T})^{+}\,\overline{\underset{\sim}{S}}\,(\underset{\sim}{V}\underset{\sim}{T}) = 1 \quad . \tag{3.27}$$

We define an orthonormal basis set $\{L_i\}$

$$L_i = \sum_{j=1}^{M} (\underset{\sim}{VT})_{ji} \chi_j(\vec{r}) \quad . \tag{3.28}$$

The representation of H^{HF} in $\{L_i\}$ basis is

$$\begin{aligned}(L_i, H^{HF}, L_j) &= \int dx \sum_p (\underset{\sim}{VT})^*_{pi} \chi^*_p H^{HF} \sum_q (\underset{\sim}{VT})_{qj} \chi_j \\ &= \sum_p \sum_q (\underset{\sim}{VT})^*_{pi} (\underset{\sim}{VT})_{qj} \int dx \chi^*_p H^{HF} \chi_q \\ &= \sum_p \sum_q (\underset{\sim}{VT})^*_{pi} \overline{H}_{pq} (\underset{\sim}{VT})_{qj} \\ &= [(\underset{\sim}{VT})^+ \underset{\sim}{\overline{H}} (\underset{\sim}{VT})]_{ij} \quad .\end{aligned} \tag{3.29}$$

Since the matrix corresponding to Equation (3.29) is hermitian, there exists a unitary matrix U which diagonalizes it, i.e.,

$$\underset{\sim}{U}^+ [(\underset{\sim}{VT})^+ \underset{\sim}{\overline{H}} (\underset{\sim}{VT})] \underset{\sim}{U} = \begin{bmatrix} \epsilon_1 & & & & \\ & \epsilon_2 & & O & \\ & & \epsilon_3 & & \\ & O & & \ddots & \\ & & & & \epsilon_M \end{bmatrix} \quad . \tag{3.30}$$

Thus the matrix (VTU) brings $\overline{H}$ into a diagonal form and

$$(\underset{\sim}{VTU})^+ \underset{\sim}{\overline{S}} (\underset{\sim}{VTU}) = 1 \quad . \tag{3.31}$$

Thus, the formal calculation of the expansion coefficients, b's, in Equation (3.24) becomes quite straightforward. However, we are required to know V^{HF} which in turn requires knowledge of the orbitals and hence, the b's. The problem is, therefore, solved self-consistently. First, N/2 functions are guessed and $\underset{\sim}{\overline{H}}$ is constructed (implies V^{HF} known); Equation (3.24) is then solved, the b's and hence, the new orbitals are obtained. Now, if these functions coincide to a specified accuracy with the guessed ones, we have evaluated the symmetry-adapted eigenfunctions of Equation (3.24) and if not, the iteration is continued until the symmetry-adapted HF eigenfunctions are obtained. Then the total wave function of the system is approximated by a single antisymmetrized product, Δ, of N electron spin orbitals, to each of which is assigned an electron of the system.

In the above treatment we have assumed that the electrons move independently of one another with average electronic repulsion acting as an effective nuclear shielding. Actually, the repulsion between the electrons is instantaneous and as a result, their motions are correlated which is quite contrary to our model of their independent motion. Thus the wave function, Δ, obtained above is definitely not an eigenfunction of the Hamiltonian of the system. In order to include correlation of the electrons, it is necessary to determine other configuration functions which are linearly independent of the first wave function Δ. We make use of the method "Configuration Interaction" (CI) which provides the eigenfunctions of the many-electron system in the theoretical limit (13). In our CI procedure the N-electron basis functions are Slater determinants constructed as antisymmetrized products of N spin orbitals chosen from among the 2M solutions to the Hartree-Fock equations. Thus, the determinantal basis functions are written as

$$\Delta_K(x_1 x_2 x_3 \ldots x_N) = \mathcal{A}[\Gamma_{k_1}(x_1)\,\Gamma_{k_2}(x_2)\ldots\Gamma_{k_N}(x_N)]$$

where K stands for an ordered set $k_1, k_2, \ldots, k_N$ of orbital indices. Though there are $\binom{2M}{N}$ possible Slater determinants which could be constructed, we limit ourselves to those configurations which, on the basis of intuition and past calculations, are thought to be the most significant. Let the total number of Slater determinants be Q. Now, in the set $\{\Gamma_i\}$, $i = 1, 2, \ldots, 2M$ where $i = 1, 2, \ldots, N$ are the N occupied orbitals and $i = N+1, N+2, \ldots, 2M$ are the (2M-N) virtual orbitals fixed by the HF Hamiltonian, let

$$\begin{aligned}\Delta_o &= \text{Hartree-Fock determinant constructed from occupied orbitals}\\ &= \mathcal{A}\{\Gamma_1(1)\,\Gamma_2(2)\ldots\Gamma_N(N)\}\end{aligned}$$

We next construct excited configurations by removing one or more electrons from the occupied orbitals and placing them in virtual orbitals. We then classify the excited configurations according to the number of such one electron excitations. The first case is that of a one electron excitation, for example, from $\Gamma_k (k \leq N)$ to $\Gamma_p (p > N)$

$$\Delta_{k\to p} = \mathcal{A}\{\Gamma_1(1)\,\Gamma_2(2)\ldots\Gamma_p(k)\ldots\Gamma_\ell(\ell)\ldots\Gamma_N(N)\} \tag{3.32}$$

In a two electron excitation, the determinant would be, for example,

$$\Delta_{\substack{k\to p\\ \ell\to q}} = \mathcal{A}\{\Gamma_1(1)\,\Gamma_2(2)\ldots\Gamma_p(k)\ldots\Gamma_q(\ell)\ldots\Gamma_N(N)\} \tag{3.33}$$

and so on.

Many determinants of the kind in Equations (3.32) and (3.33) can be constructed. The total wave function, Ψ, of the N particle system is then described by a linear combination of Q such determinants chosen as described earlier, i.e.,

$$\Psi = \sum_{j=1}^{Q} C_j \Delta_j \tag{3.34}$$

where the expansion coefficients, C's, are still undetermined. The secular equation is then formed by applying the variational principle to the expectation value of the Hamiltonian as discussed earlier in the chapter and substituting for Ψ, in terms of determinants from Equation (3.34), the secular equation obtained is of the form

$$\sum_{L=1}^{Q} [H_{KL} - E_L \, \delta_{KL}] \, C_L = 0 \qquad \text{for } K=1,2,3,\ldots,Q \tag{3.35}$$

where

$$H_{KL} = < \Delta_K | H_{os} | \Delta_L >$$

and

$$\delta_{KL} = < \Delta_K | \Delta_L > \, .$$

Since $H^+ = H$, the $(Q \times Q)$ H is diagonalized by standard matrix techniques, i.e., by a unitary matrix U and the solutions

$$\underset{\sim}{U}^{+}\underset{\sim}{H}\,\underset{\sim}{U} = \begin{bmatrix} E_1 & & & & \\ & E_2 & & \mathcal{O} & \\ & & E_3 & & \\ & \mathcal{O} & & \ddots & \\ & & & & E_Q \end{bmatrix}$$

consist of the first Q approximate eigenvectors (stated as the columns of $\underset{\sim}{U}$ matrix) and the corresponding Q eigenvalues of the total N-electron Hamiltonian. The eigenvalues satisfy the variational theorem that $E_i \geq \mathcal{E}_i$ where $\mathcal{E}_i$ is the exact i^{th} eigenvalue of the full N-electron Hamiltonian.

The goodness of these approximate eigenfunctions, of course, depends upon the space spanned by the Q determinants chosen; this in turn depends upon the orbitals used in constructing them and thus, ultimately, on the Gaussian orbital basis chosen to expand the Hartree-Fock orbitals. Since the orbitals are optimized by the Fock equations specifically for the ground state, one can reasonably conclude that the ground state will be the most accurately described. As a matter of fact, however, since the variational calculation focuses its criteria on energy, we should not expect the wave functions, including the ground state and calculated from such a limited CI as we have performed, to be highly accurate. The crucial question is whether or not the CI procedure has introduced the approximate correlation effects into the wave functions which are necessary to compute accurate oscillator strengths. That, in fact, there are for such calculations, specific correlation effects which must be introduced while other correlation effects may be ignored, has been argued by Oksüz and Sinanoğlu (14) and substantiated by calculations on atoms by Westhaus and Sinanoğlu (15), Nicholaides and Sinanoğlu (16) and Shrivastava and Westhaus (17). We have only begun to explore this idea for molecules and unfortunately have no detailed calculations to present here. We simply used the excited states of the one CI calculation to approximate the excited stationary states of the molecule in our transition probability calculations. Having optimized the orbitals for the ground state, one can not expect the excited states to be accurately represented by the excited states of the CI calculations.

CHAPTER IV

EVALUATION OF ONE ELECTRON MATRIX ELEMENTS

Let $\Psi_i(x_1, x_2, x_3, \ldots, x_N)$ be a wave function of the N-particle system corresponding to an initial state $|i>$ and $\Psi_f(x_1, x_2, \ldots, x_N)$, another wave function, corresponding to another state $|f>$. The total wave function of the system is written as linear combinations of several Slater determinants, e.g.,

$$\Psi_i(x_1, x_2, \ldots, x_N) = \sum_{K=1}^{Q} C^i_K \Delta_K(x_1, x_2, \ldots, x_N) \quad . \tag{4.1}$$

The Slater determinants in turn are built from N orthonormal symmetry-adapted molecular spin orbitals chosen from 2M molecular orbitals.

$$\{\Gamma_1, \Gamma_2, \Gamma_3, \ldots, \Gamma_{2M}\} \ ; \ 2M \geq N \tag{4.2}$$

and

$$\Delta_K(x_1, x_2, \ldots, x_N) = \mathcal{A}(N)\{\Gamma_{k_1}(x_1)\Gamma_{k_2}(x_2) \ldots \Gamma_{k_N}(x_N)\} \tag{4.3}$$

where K stands for a particular ordered set of N molecular spin orbitals. Similarly,

$$\Psi_f(x_1 x_2 \ldots x_N) = \sum_{L=1}^{Q} C^f_L \Delta'_L(x_1 x_2 \ldots x_N) \tag{4.4}$$

where,

$$\Delta_L'(x_1\, x_2\, \ldots\, x_N) = \mathcal{A}(N)\{\Gamma'_{\ell_1}(x_1)\, \Gamma'_{\ell_2}(x_2) \ldots \Gamma'_{\ell_N}(x_N)\} \qquad (4.5)$$

The primes denote that the spin orbitals used to construct these Slater determinants can, in general, be chosen from another set

$$\{\Gamma_1',\ \Gamma_2',\ \Gamma_3',\ \ldots\ \Gamma'_{2M}\}\ ;\ 2M' \geq N\ . \qquad (4.6)$$

The formalism is presented for the general case where the two sets of orthonormal orbitals used to construct the wave functions for the two states have no particular orthonormality relations between them. In the numerical results presented in the next chapter, both states, in fact, were constructed from the same set of orbitals. In general,

$$\int \Gamma_i^*(x)\, \Gamma_j(x)\, dx = \delta_{ij} \quad ;\ i, j = 1,\ 2,\ \ldots,\ 2M\ , \qquad (4.7a)$$

$$\int \Gamma_i'^*(x)\, \Gamma_j'(x)\, dx = \delta_{ij} \quad ;\ i, j = 1,\ 2,\ \ldots,\ 2M'\ , \qquad (4.7b)$$

$$\int \Gamma_i^*(x)\, \Gamma_j'(x)\, dx = S_{ij} \quad ;\ i = 1, 2, \ldots, 2M$$

$$j = 1, 2, \ldots, 2M'\ . \qquad (4.7c)$$

These molecular spin orbitals can themselves be expanded in terms of a basis set. We have chosen, for simplicity in calculations, the Gaussian basis set consisting of atomic orbitals coupled with spin functions. Thus,

$$\Gamma_i(x) = \sum_j b_{ji}\, \theta_j(x) \qquad (4.8a)$$

where $\theta_j(x)$ is the atomic spin orbital made up of a spatial Gaussian function $\chi(\vec{r})$ and a spin function $\lambda_{m_s}(\xi)$, i.e.,

$$\theta_j(x) = \theta_j(\vec{r}, \xi) = \chi_j(\vec{r})\, \lambda_{m_s}(\xi) \tag{4.8b}$$

A Gaussian function at x_A, y_A, z_A can be written as

$$\chi_j(\vec{r}) = \eta \{ (x - x_A)^{n_x} (y - y_A)^{n_y} (z - z_A)^{n_z} \cdot e^{-\alpha[(x - x_A)^2 + (y - y_A)^2 + (z - z_A)^2]} \} \tag{4.8c}$$

where η is a normalization factor and n_x, n_y, n_z characterizes the S, P, D, F, ... orbitals. For spherically symmetric orbitals (i.e., the S-orbitals), n_x, n_y, n_z are all zero. Our calculations are done with floating spherical Gaussian orbitals suggested by Frost (18), but our programs are general for other orbitals (arbitrary n_x, n_y, n_z) also. The Γ's can similarly be expanded in terms of different Gaussian basis set

$$\Gamma_i'(x) = \sum_j b_{ji}' \, \theta_j'(x) \tag{4.9a}$$

where

$$\theta_j'(x) = \chi_j'(\vec{r})\, \lambda_{m_s}(\xi) \tag{4.9b}$$

With the above background we proceed to calculate the matrix elements, namely

$$< \Psi_f | O^j | \Psi_i >$$

where

$$O^j = \sum_{m=1}^{N} O_m^j \tag{4.10}$$

and

$$O^j = O^1, O^2, O^3, O^4, O^5, O^6$$

$$\equiv x, y, z, \frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} .$$

The evaluation of

$$< \Psi_f | O^j | \Psi_i >$$

$$= < \sum_L C_L'^{f} \Delta_L'(x_1 x_2 \ldots x_N) | O^j | \sum_K C_K^i \Delta_K(x_1 x_2 \ldots x_N) >$$

$$= \sum_L \sum_K C_L'^{f*} C_K^i < \Delta_L' | O^j | \Delta_K > \qquad (4.11)$$

requires evaluation of

$$< \Delta_L' | O^j | \Delta_K > = \Big[\int dx_1 \int dx_2 \ldots \int dx_N \Delta_L'^{*}(x_1, x_2, \ldots, x_N) \cdot$$

$$\{O^j\} \; \Delta_K(x_1 x_2 \ldots x_N) \Big]$$

$$= \Big[\int dx_1 \ldots \int dx_N \, \{ \Gamma_{\ell_1}'^{*}(x_1) \, \Gamma_{\ell_2}'^{*}(x_2) \ldots \Gamma_{\ell_N}'^{*}(x_N) \} \cdot$$

$$\mathcal{A}^{+} \{O^j\} \mathcal{A} \{ \Gamma_{k_1}(x_1) \, \Gamma_{k_2}(x_2) \ldots \Gamma_{k_N}(x_N) \} \Big]$$

$$= \Big[\int dx_1 \ldots \int dx_N \, \{ \Gamma_{\ell_1}'^{*}(x_1) \ldots \Gamma_{\ell_N}'^{*}(x_N) \} \cdot$$

$$\{O^j\} \mathcal{A}^2 \{ \Gamma_{k_1}(x_1) \, \Gamma_{k_2}(x_2) \ldots \Gamma_{k_N}(x_N) \} \Big]$$

$$(4.12)$$

since $\mathcal{A}$ is self-adjoint and commutes with O^j.

Now, by the group theory

$$\mathcal{A}^2 = \frac{1}{\sqrt{N!}} \sum_{P} (-1)^{\sigma_P} P \cdot \frac{1}{\sqrt{N!}} \sum_{P'} (-1)^{\sigma_{P'}} P'$$

$$= \frac{1}{\sqrt{N!}} \sum_{P} \sum_{P'} (-1)^{\sigma_P} (-1)^{\sigma_{P'}} P P'$$

$$= \sum_{P''} (-1)^{\sigma_{P''}} P'' \equiv \sum_{P} (-1)^{\sigma_P} P$$

the matrix element in Equation (4.12) is

$$\Big[\int dx_1 \ldots \int dx_N \{ \Gamma'^{*}_{\ell_1}(x_1)\, \Gamma'^{*}_{\ell_2}(x_2) \ldots \Gamma'^{*}_{\ell_N}(x_N) \} \cdot \{ \sum_m O^j_m \} \cdot$$

$$\sum_{P} (-1)^{\sigma_P} P\{ \Gamma_{k_1}(x_1)\, \Gamma_{k_2}(x_2) \ldots \Gamma_{k_N}(x_N) \} \Big]$$

Now

$$\Big[\sum_{P} (-1)^{\sigma_P} P\{ \Gamma_{k_1}(x_1)\, \Gamma_{k_2}(x_2) \ldots \Gamma_{k_N}(x_N) \} \Big]$$

is a sum of N! terms. Thus, $< \Delta'_L | O^j | \Delta_K >$ can be written as sum of N determinants. That is,

$$< \Delta'_L | O^j | \Delta_K > = \sum_{p=1}^{N} D^{LK}_p \qquad (4.13)$$

where the determinant

$$D_p^{LK} = \begin{bmatrix} S_{\ell_1 k_1} & S_{\ell_1 k_2} & \cdots & S_{\ell_1 k_N} \\ S_{\ell_2 k_1} & S_{\ell_2 k_2} & \cdots & S_{\ell_2 k_N} \\ \vdots & \vdots & & \vdots \\ T^j_{\ell_p k_1} & T^j_{\ell_p k_2} & \cdots & T^j_{\ell_p k_N} \\ \vdots & & & \vdots \\ S_{\ell_N k_1} & \cdots & & S_{\ell_N k_N} \end{bmatrix} \leftarrow p^{\text{th}} \text{ row} \qquad (4.14)$$

or

$$< \Delta_L' \,|\, O^j \,|\, \Delta_K >$$

$$= \begin{bmatrix} T^j_{\ell_1 k_1} & T^j_{\ell_1 k_2} & \cdots & T^j_{\ell_1 k_N} \\ S_{\ell_2 k_1} & S_{\ell_2 k_2} & \cdots & S_{\ell_2 k_N} \\ \vdots & & \ddots & \vdots \\ S_{\ell_N k_1} & \cdots & & S_{\ell_N k_N} \end{bmatrix}$$

$$+ \begin{bmatrix} S_{\ell_1 k_1} & S_{\ell_1 k_2} & \cdots & S_{\ell_1 k_N} \\ T^j_{\ell_2 k_1} & T^j_{\ell_2 k_2} & \cdots & T^j_{\ell_2 k_N} \\ \vdots & & \ddots & \vdots \\ S_{\ell_N k_1} & \cdots & & S_{\ell_N k_N} \end{bmatrix} + \cdots\cdot$$

$$+ \begin{bmatrix} S_{\ell_1 k_1} & S_{\ell_1 k_2} & \cdots & S_{\ell_1 k_N} \\ S_{\ell_2 k_1} & S_{\ell_2 k_2} & \cdots & S_{\ell_2 k_N} \\ \vdots & \vdots & \ddots & \vdots \\ T^j_{\ell_N k_1} & T^j_{\ell_N k_2} & \cdots & T^j_{\ell_N k_N} \end{bmatrix} \qquad (4.15)$$

where

$$S_{\ell_i k_p} = \langle \Gamma'_{\ell_i} | \Gamma_{k_p} \rangle$$

$$= \int dx_1\, \Gamma'^{*}_{\ell_i}(x_1)\, \Gamma_{k_p}(x_1) \qquad (4.16)$$

and

$$T^j_{\ell_i k_p} = \langle \Gamma'_{\ell_i} | O^j_1 | \Gamma_{k_p} \rangle$$

$$= \int dx_1\, \Gamma'^{*}_{\ell_i}(x_1)\, O^j(x_1)\, \Gamma_{k_p}(x_1) \quad . \qquad (4.17)$$

Now, expanding the molecular spin orbitals, Γ, in terms of Gaussian basis orbitals according to Equations (4.8) and (4.9), we find

$$S_{\ell_i k_p} = \langle \Gamma'_{\ell_i}(x) | \Gamma_{k_p}(x) \rangle = \langle \sum_t b'_{t\ell_i}\, \theta'_t(x) | \sum_q b_{qk_p}\, \theta_q(x) \rangle$$

$$= \sum_t \sum_q b'^{*}_{t\ell_i}\, b_{qk_p} \langle \chi'_t(\vec{r}) \lambda_{m'_s}(\xi) | \chi_q(\vec{r})\, \lambda_{m_s}(\xi) \rangle$$

$$= [\sum_t \sum_q b'^{*}_{t\ell_i}\, b_{qk_p} \sum_\xi \lambda^{*}_{m'_s}(\xi)\, \lambda_{m_s}(\xi) \int d\vec{r}\, \chi'^{*}_t(\vec{r})\, \chi_q(\vec{r})]$$

$$= \sum_t \sum_q b'^{*}_{t\ell_i}\, b_{qk_p}\, \delta_{m'_s m_s} \langle \chi_t | \chi_q \rangle \qquad (4.18)$$

Similarly,

$$T^j_{\ell_i k_p} = \sum_t \sum_q b'^{*}_{t\ell_i}\, b_{qk_j}\, \delta_{m'_s m_s} \langle \chi_t | O^j_1 | \chi_q \rangle$$

The possibilities of six different O^j operators is already mentioned. Thus, including these six, along with the overlap matrix element Equation (4.18), we have seven types of one electron integrals to calculate. These calculations are shown below.

Let us evaluate, for example, $<\chi'_A|O^j|\chi_B>$ for velocity operator where

$$O^j = \vec{\nabla} j = \frac{\partial}{\partial x}\hat{x} + \frac{\partial}{\partial y}\hat{y} + \frac{\partial}{\partial z}\hat{z}$$

where $\hat{x}$, $\hat{y}$, $\hat{z}$ are the unit vectors along x, y, z axes of coordinate system. Let us look at one of the components in particular, say $\frac{\partial}{\partial x}$

$$< \chi'_A | \frac{\partial}{\partial x} | \chi_B >$$

$$= \eta_A \eta_B \int dx\,dy\,dz \,\{(x-x_A)^{n_{Ax}} (y-y_A)^{n_{Ay}} (z-z_A)^{n_{Az}} \cdot$$

$$e^{-\alpha_A[(x-x_A)^2 + (y-y_A)^2 + (z-z_A)^2]}\}\{\frac{\partial}{\partial x}\} \cdot$$

$$\{(x-x_B)^{n_{Bx}} (y-y_B)^{n_{By}} (z-z_B)^{n_{Bz}} \cdot$$

$$e^{-\alpha_B[(x-x_B)^2 + (y-y_B)^2 + (z-z_B)^2]}\} \qquad (4.20)$$

where χ'_A is centered on (x_A, y_A, z_A). The numbers (n_{Ax}, n_{Ay}, n_{Az}) determine whether it is S, P, D . . . orbital and α_A is a positive quantity. χ_B is similarly centered on (x_B, y_B, z_B). Now we evaluate

$$e^{-\alpha_A(x-x_A)^2}\, e^{-\alpha_B(x-x_B)^2}$$

$$= \left\{ e^{-(\alpha_A+\alpha_B)\left[x^2 - \frac{2(\alpha_A x_A + \alpha_B x_B)\,x}{\alpha_A + \alpha_B}\right]} \cdot e^{-\alpha_A x_A^2 - \alpha_B x_B^2} \right\}$$

$$= \left\{ e^{-(\alpha_A+\alpha_B)\left[x - \frac{\alpha_A x_A + \alpha_B x_B}{\alpha_A + \alpha_B}\right]^2} \cdot \right.$$

$$\left. e^{-\alpha_A x_A^2 - \alpha_B x_B^2 + \frac{(\alpha_A x_A + \alpha_B x_B)^2}{\alpha_A + \alpha_B}} \right\}$$

$$= e^{-\alpha \xi_x^2} \cdot e^{-\left[\frac{\alpha_A \alpha_B}{\alpha}(x_A - x_B)^2\right]}$$

$$= e^{-\alpha \xi_x^2} \cdot e^{-\lambda_x^2} \qquad (4.21)$$

where

$$\left.\begin{aligned} \xi_x &= x - \frac{\alpha_A x_A + \alpha_B x_B}{\alpha_A + \alpha_B}, \quad \alpha = \alpha_A + \alpha_B \\ &\text{and} \\ \lambda_x &= \sqrt{\frac{\alpha_A \alpha_B}{\alpha}}\,(x_A - x_B) \\ d\xi_x &= dx \end{aligned}\right\} \qquad (4.22)$$

and similarly

$$e^{-\alpha_A(y-y_A)^2} \cdot e^{-\alpha_B(y-y_B)^2} = e^{-\alpha \xi_y^2} \cdot e^{-\lambda_y^2} .$$

and

$$e^{-\alpha_A(z-z_A)^2} \cdot e^{-\alpha_B(z-z_B)^2} = e^{-\alpha \xi_z^2} \cdot e^{-\lambda_z^2}$$

Furthermore, we can rewrite $(x-x_A)$ as

$$(x-x_A) = \xi_x + \frac{\alpha_A x_A + \alpha_B x_B}{\alpha_A + \alpha_B} - x_A$$

$$= \xi_x + \frac{x_B \alpha_B - x_A \alpha_B}{\alpha_A + \alpha_B}$$

$$= \xi_x + \sqrt{\frac{\alpha_B}{\alpha_A}} \sqrt{\frac{\alpha_B \alpha_A}{\alpha_A + \alpha_B}} (x_B - x_A) \frac{1}{\sqrt{\alpha}}$$

$$= (\xi_x - \frac{\lambda_x}{\nu} \frac{1}{\sqrt{\alpha}}) \tag{4.23}$$

where

$$\nu = \sqrt{\alpha_A/\alpha_B}$$

Similarly, we get

$$(x-x_B) = (\xi_x + \frac{\nu \lambda_x}{\sqrt{\alpha}}) \tag{4.24}$$

The integral in Equation (4.20) clearly factors into the product of three one-dimensional integrals to become

$$\eta_A \eta_B \int dz (z-z_A)^{n_{Az}} e^{-\alpha_A(z-z_A)^2} (z-z_B)^{n_{Bz}} e^{-\alpha_B(z-z_B)^2} \cdot$$

$$\int dy (y-y_A)^{n_{Ay}} e^{-\alpha_A(y-y_A)^2} (y-y_B)^{n_{By}} e^{-\alpha_B(y-y_B)^2}$$

$$\int dx (x-x_A)^{n_{Ax}} e^{-\alpha_A(x-x_A)^2} \{\frac{\partial}{\partial x}\} (x-x_B)^{n_{Bx}} e^{-\alpha_B(x-x_B)^2}$$

$$= e^{-(\lambda_y^2 + \lambda_z^2)} \int d\xi_z \left(\xi_z - \frac{\lambda_z}{\nu\sqrt{\alpha}}\right)^{n_{Az}} \left(\xi_z + \frac{\nu\lambda_z}{\sqrt{\alpha}}\right)^{n_{Bz}} e^{-\alpha\xi_z^2} .$$

$$\int d\xi_y \left(\xi_y - \frac{\lambda_y}{\nu\sqrt{\alpha}}\right)^{n_{Ay}} \left(\xi_y + \frac{\nu\lambda_y}{\sqrt{\alpha}}\right)^{n_{By}} e^{-\alpha\xi_y^2} \cdot e^{-\lambda_x^2} .$$

$$\int d\xi_x \left(\xi_x - \frac{\lambda_x}{\nu\sqrt{\alpha}}\right)^{n_{Ax}} \Big[n_{Bx} \left(\xi_x + \frac{\nu\lambda_x}{\sqrt{\alpha}}\right)^{n_{Bx}-1}$$

$$- 2\alpha_B \left(\xi_x + \frac{\nu\lambda_x}{\sqrt{\alpha}}\right)^{n_{Bx}+1} \Big] \cdot e^{-\alpha\xi_x^2}$$

$$= \{ e^{-(\lambda_x^2 + \lambda_y^2 + \lambda_z^2)} I_z(n_{Az}, n_{Bz}) \cdot I_y(n_{Ay}, n_{By}) \cdot$$

$$[n_{Bx} I_x (n_{Ax}, n_{Bx}-1) - 2\alpha_B I_x(n_{Ax}, n_{Bx}+1)] \} \qquad (4.25)$$

Let us evaluate I_z from Equation (4.25)

$$I_z(n_{Az}, n_{Bz}) = \int d\xi_z \left(\xi_z - \frac{\lambda_z}{\nu\sqrt{\alpha}}\right)\left(\xi_z + \frac{\nu\lambda_z}{\sqrt{\alpha}}\right)^{n_{Bz}} \cdot e^{-\alpha\xi_z^2}$$

$$= \left[\int d\xi z \sum_{\ell} \binom{n_{Az}}{\ell} \left(\frac{-\lambda_z}{\nu\sqrt{\alpha}}\right)^{\ell} \xi_z^{n_{Az}-\ell} \cdot \right.$$

$$\left. \sum_m \binom{n_{Bz}}{m} \left(\frac{\nu\lambda_z}{\sqrt{\alpha}}\right)^m \xi_z^{n_{Bz}-m} \cdot e^{-\alpha\xi_z^2} \right]$$

$$= \left[\sum_{\ell} \sum_m \binom{n_{Az}}{\ell} \binom{n_{Bz}}{m} (-1)^{\ell} \lambda_z^{m+\ell} \nu^{m-\ell} \cdot \right.$$

$$\left. \frac{1}{\sqrt{\alpha^{\ell+m}}} \int d\xi_z \, \xi_z^{n_{Az} + n_{Bz} - \ell - m} e^{-\alpha\xi_z^2} \right]$$

$$= \sum_{\ell}\sum_{m} C_{\ell m} \frac{\sqrt{\pi/\alpha}}{\sqrt{\alpha^{\ell+m}}} \, \frac{(n_{Az}+n_{Bz}-\ell-m-1)!!}{\sqrt{(2\alpha)^{n_{Az}+n_{Bz}-\ell-m}}}$$

$$= \left[\sum_{\ell}\sum_{m} C_{\ell m} \frac{(n_{Az}+n_{Bz}-\ell-m-1)!!\,\sqrt{\pi}}{\sqrt{2^{n_{Az}+n_{Bz}-\ell-m}}\,\sqrt{\alpha^{n_{Az}+n_{Bz}+1}}}\right] \qquad (4.26)$$

where

$$C_{\ell m} = \binom{n_{Az}}{\ell}\binom{n_{Bz}}{m}(-1)^{\ell}\lambda_z^{m+\ell}\,\nu^{m-\ell}$$

Thus, $I_z(n_{Az}, n_{Bz})$ can be evaluated by Equation (4.26). Similarly, $I_x(n_{Ax}, n_{Bx})$, $I_y(n_{Ay}, n_{By})$ can also be evaluated.

When O = x, the matrix element for length operator is

$$\langle \chi_A' | x | \chi_B \rangle$$

$$= \int dx\,dy\,dz\,\chi_A'(x-x_B)\chi_B + x_B \int dx\,dy\,dz\,\chi_A\chi_B$$

$$= \eta_A\eta_B\,[I_y(n_{Ay}, n_{By})\,I_z(n_{Az}, n_{Bz})\,I_x(n_{Ax}, n_{Bx}+1)$$

$$+ x_B\,I_y(n_{Ay}, n_{By})\,I_z(n_{Az}, n_{Bz})\,I_x(n_{Ax}, n_{Bx})] \qquad (4.27)$$

Similarly the one electron matrix elements for the operator y, z, $\frac{\partial}{\partial y}$ and $\frac{\partial}{\partial z}$ can also be calculated. These matrix elements are then evaluated over the symmetry-adapted molecular orbitals and subsequently over the Slater determinants and finally over the total wave functions.

In our numerical calculations the "WAVE-FUNCTION" and "TRANPROB" together complete the entire calculation required for the determination of the "Transition Probability". The "WAVE-FUNCTION" part of the program provides the expansion coefficients, b's, required in the atomic expansion of the molecular orbitals and the expansion coefficients, C's, required in the determinantal expansion of the wave functions and the energies of ${}^1A_1$, ${}^1B_1$, ${}^3B_1$ ... states. Then "TRANPROB" calculates the electric-dipole oscillator strength using S_R, S_∇ and ΔE, the energy difference between the states ${}^1A_1$ and ${}^1B_1$. The "TRANPROB" is given in Appendix B; since the "WAVE-FUNCTION" program is undergoing improvements, it is not included here.

CHAPTER V

RESULTS AND CONCLUSIONS

These initial transition probability calculations have been done to test the accuracy of the wave functions by comparing the results for calculated oscillator strengths with other calculations of the oscillator strength. There are two equivalent dipole matrix elements S_R, S_∇ which would yield identical results, for vertical transition probability, if the electronic wave functions used are exact. From these results the accuracy in the wave functions, in the region where the respective operators, R and ∇ are sensitive, can be inferred. A large discrepancy in our results with S_R and S_∇ has been observed since the proper correlation effects have not been included in these initial calculations.

Our results are limited to the ${}^1A_1 \rightarrow {}^1B_1$ electronic transition moments at the experimental equilibrium geometry of the ground electronic state. We have calculated the oscillator strength using

a) experimental energy difference obtained from the electronic spectrum;

b) theoretical energy difference obtained by diagonalizing the Hamiltonian in CI calculations.

There have been other explorations of the electronic states of H_2O, notably the one by Taylor and Segal (19). A summary of their results obtained by INDO calculations augmented by experimental data is given in Table I. This gives the equilibrium geometry of each electronic state. They have also computed the oscillator strength for vertical electronic transition in the equilibrium geometry.

We, in our calculations, have used the floating spherical Gaussian orbitals as basis functions. The two sets of calculations were performed with 1) eight basis functions and 2) ten basis functions. These basis functions are given in Table II and Table II, respectively.

The resulting spatial MO are expanded as $\Phi_\mu = \Sigma b_{i\mu} \chi_i$. The spatial orbitals are listed in the Table IV for the eight-basis functions and in Table V for the ten-basis functions. The molecular orbitals are listed according to their HF energy (in increasing order). The orbitals exhibit the ground state symmetry, C_{2V}, for the water molecule and the first five in both the states are energy ordered such that

$$\epsilon_{1a_1} < \epsilon_{2a_1} < \epsilon_{1b_2} < \epsilon_{3a_1} < \epsilon_{1b_1}$$

Since the atomic ground state and the excited states are represented by the same Gaussian functions, the expansion coefficients for the ground (molecular) state and for the excited states are the same, i.e., $b_{i\mu} = b'_{i\mu}$.

The Slater determinants to be used in CI calculations are constructed as follows. First, the spin orbitals are built from the spatial orbitals:

TABLE I

RESULTS OBTAINED BY INDO CALCULATIONS FOR WATER MOLECULE (19)

State	Valence M_O Configuration	Bond Length A^o	Bond Angle (°)	Energy (eV)	Vertical Excitation Energy(eV)
Singlets					
$\tilde{x}\,^1A_1$	$3a_1^2 1b_1^2$	1.03_0	104^o	0.00	
$A\,^1B_1$	$3a_1^2 1b_1 4a_1$	1.11_5	180^o	5.79	7.43
$^1A_2$	$3a_1^2 1b_1 2b_2$	1.11_0	76^o	6.93	8.03
$\tilde{B}\,^1A_1$	$3a_1 1b_1^2 4a_1$	1.11_5	180^o	5.79	10.85
$^1B_2$	$3a_1 1b_1^2 2b_2$	1.10_0	122^o	10.27	11.06
Triplets					
$^3B_1$	$3a_1^2 1b_1 4a_1$	1.11_7	180^o	4.76	6.20
$^3A_2$	$3a_1^2 1b_1 2b_2$	1.11_7	80^o	6.55	7.49
$^3A_1$	$3a_1 1b_1^2 4a_1$	1.11_0	180^o	4.76	8.67
$^3B_2$	$3a_1 1b_1^2 2b_2$	1.11_5	112^o	8.90	9.43

TABLE II

SPHERICAL GAUSSIAN ATOMIC ORBITAL BASIS [GROUND STATE EQUILIBRIUM GEOMETRY (2)] FOR EIGHT ORBITALS*

	α	x	y	z
1	17.361084	0.0	0.0	0.000500
2	0.300000	0.0	0.0	-0.400000
3	0.600000	0.0	0.285200	0.222800
4	0.600000	0.0	-0.285200	0.222800
5	0.500000	0.0	0.570400	0.445600
6	0.500000	0.0	-0.570400	0.445600
7	0.564470	0.100000	0.0	0.022000
8	0.564470	-0.100000	0.0	0.022000

*The atomic orbital bases for the ground state and the excited states are identical.

TABLE III

SPHERICAL GAUSSIAN ATOMIC ORBITAL BASIS [GROUND STATE EQUILIBRIUM GEOMETRY (2)] FOR TEN ORBITALS*

	α	x	y	z
1	17.361099	0.0	0.0	0.000500
2	3.500000	0.0	0.0	-0.100000
3	0.574490	0.0	0.399000	0.376000
4	0.574490	0.0	-0.399000	0.376000
5	0.500000	0.0	0.570400	0.445600
6	0.500000	0.0	-0.570400	0.445600
7	0.564470	0.100000	0.0	0.022000
8	0.564470	-0.100000	0.0	0.022000
9	0.400000	0.0	0.855600	0.668400
10	0.400000	0.0	-0.855600	0.668400

*The atomic orbital bases for the ground state and the excited states are identical.

TABLE IV

THE EXPANSION COEFFICIENTS, b'S FOR EIGHT-ORBITAL BASIS

	Φ_1	Φ_2	Φ_3	Φ_4	Φ_5	Φ_6	Φ_7	Φ_8
χ_1	0.935957	-0.289889	0.0	-0.090823	0.0	0.254306	0.000010	0.229495
χ_2	-0.008782	0.161394	0.0	0.423454	0.0	3.019263	0.000139	-2.996997
χ_3	0.484432	0.097485	-0.635784	-1.601466	0.0	2.962797	-9.559875	-10.228598
χ_4	0.484423	0.097485	0.635784	-1.601480	0.0	2.962036	9.560209	-10.228499
χ_5	-0.280033	0.310663	-0.981838	-0.753640	0.0	-0.610147	5.456485	4.446961
χ_6	-0.280027	0.310663	0.981838	-0.753632	0.0	-0.609713	-5.456569	4.446905
χ_7	-0.147785	0.053492	0.0	2.108002	-6.673815	-3.767438	-0.000191	7.312892
χ_8	-0.147786	0.053492	0.0	2.108002	6.673815	-3.767438	-0.000191	7.312892

TABLE V

THE EXPANSION COEFFICIENTS, b'S FOR TEN-ORBITAL BASIS

	Φ_1	Φ_2	Φ_3	Φ_4	Φ_5	Φ_6	Φ_7	Φ_8	Φ_9	Φ_{10}
χ_1	0.790090	-0.246714	0.0	-0.110393	0.0	0.104755	0.0	0.192821	0.0	-1.043598
χ_2	0.291364	-0.111406	0.0	0.050349	0.0	0.073865	0.0	0.272419	0.0	1.868972
χ_3	0.103207	0.640105	-6.090241	-4.229549	0.0	-0.665725	-5.394416	-11.559936	-32.734161	-4.582755
χ_4	0.103207	0.640105	6.090241	-4.229549	0.0	-0.665725	5.394416	-11.559936	32.734161	-4.582755
χ_5	-0.076750	-0.994611	6.179340	4.053670	0.0	-3.323939	11.225254	18.645447	36.920151	7.328547
χ_6	-0.076750	-0.994611	-6.179340	4.053670	0.0	-3.323939	-11.225254	18.645447	-36.920151	7.328547
χ_7	-0.037591	0.390076	0.0	1.627406	-6.673815	1.216110	0.0	-0.835043	0.0	-0.906694
χ_8	-0.037591	0.390076	0.0	1.627406	6.673815	1.216110	0.0	-0.835043	0.0	-0.906694
χ_9	0.011712	0.547574	-2.308533	-1.582271	0.0	2.978290	-5.665077	-6.517885	-9.484904	-2.257087
χ_{10}	0.011712	0.547574	2.308533	-1.582271	0.0	2.978290	5.665077	-6.517885	9.484904	-2.257087

$$\Gamma_{\underset{\sim}{\mu}} = \Phi_{\mu}(\vec{r})\,\alpha(\xi) \qquad \text{odd } \underset{\sim}{\mu}: \quad \mu = \frac{\underset{\sim}{\mu} + 1}{2}$$

$$\Gamma_{\underset{\sim}{\mu}} = \Phi_{\mu}(\vec{r})\,\beta(\xi) \qquad \text{even } \underset{\sim}{\mu}: \quad \mu = \frac{\underset{\sim}{\mu}}{2} \quad .$$

In the first set with eight spatial orbitals the restricted determinantal basis is constructed from the 16 spin orbitals.

$$\Delta_1 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_9\,\Gamma_{10}]$$

$$\Delta_2 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_{13}\,\Gamma_{14}]$$

$$\Delta_3 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_{15}\,\Gamma_{16}]$$

$$\Delta_4 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_{13}\,\Gamma_{16}]$$

$$\Delta_5 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_{14}\,\Gamma_{15}]$$

$$\Delta_6 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_9\,\Gamma_{12}]$$

$$\Delta_7 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_{10}\,\Gamma_{11}]$$

The wave functions for the various electronic states of the system are

$$\Psi_{\lambda} = \sum_{I=1}^{7} C_{I\lambda}\,\Delta_I$$

The expansion coefficients, C's, for the ${}^1A_1$, ${}^1B_1$, and ${}^3B_1$ are given in Table VI.

In the second set with ten spatial orbitals, the determinantal basis from 20 spin orbitals is,

$$\Delta_1 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_9\,\Gamma_{10}]$$

$$\Delta_2 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_7\,\Gamma_8\,\Gamma_{11}\,\Gamma_{12}]$$

$$\Delta_3 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_5\,\Gamma_6\,\Gamma_9\,\Gamma_{10}\,\Gamma_{11}\,\Gamma_{12}]$$

$$\Delta_4 = \mathcal{A}[\Gamma_1\,\Gamma_2\,\Gamma_3\,\Gamma_4\,\Gamma_7\,\Gamma_8\,\Gamma_9\,\Gamma_{10}\,\Gamma_{11}\,\Gamma_{12}]$$

TABLE VI

THE CI EXPANSION COEFFICIENTS, C'S, FOR THE $^1A_1$, $^1B_1$ AND $^3B_1$ STATES IN THE SEVEN-DETERMINANTAL BASIS SET

$\Psi \rightarrow$	$^1A_1$	$^1B_1$	$^3B_1$
Δ_1	0.999924	0.0	0.0
Δ_2	-0.009678	0.0	0.0
Δ_3	-0.007596	0.0	0.0
Δ_4	-0.000000	0.0	0.0
Δ_5	0.000000	0.0	0.0
Δ_6	0.0	0.707107	-0.707107
Δ_7	0.0	0.707107	0.707107

$$\Delta_5 = \mathcal{A}[\Gamma_1 \Gamma_2 \Gamma_5 \Gamma_6 \Gamma_7 \Gamma_8 \Gamma_9 \Gamma_{10} \Gamma_{11} \Gamma_{12}]$$

$$\Delta_6 = \mathcal{A}[\Gamma_1 \Gamma_2 \Gamma_3 \Gamma_4 \Gamma_5 \Gamma_6 \Gamma_7 \Gamma_8 \Gamma_9 \Gamma_{16}]$$

$$\Delta_7 = \mathcal{A}[\Gamma_1 \Gamma_2 \Gamma_3 \Gamma_4 \Gamma_5 \Gamma_6 \Gamma_7 \Gamma_8 \Gamma_{10} \Gamma_{15}]$$

$$\Delta_8 = \mathcal{A}[\Gamma_1 \Gamma_2 \Gamma_3 \Gamma_4 \Gamma_5 \Gamma_6 \Gamma_7 \Gamma_8 \Gamma_9 \Gamma_{12}]$$

$$\Delta_9 = \mathcal{A}[\Gamma_1 \Gamma_2 \Gamma_3 \Gamma_4 \Gamma_5 \Gamma_6 \Gamma_7 \Gamma_8 \Gamma_{10} \Gamma_{11}]$$

The wave functions for the various electronic states of the system are

$$\Psi_{I\lambda} = \sum_{I=1}^{9} C_{I\lambda} \Delta_I$$

The expansion coefficients, C's, for the $^1A_1$, $^1B_1$, and $^3B_1$ are given in Table VII.

The results of our transition probability, given by the formulae

$$f_R = \frac{2}{3}(E_f - E_i) |< \Psi_f| \sum_{j=1}^{N} \vec{r}_j |\Psi_i> |^2$$

$$f_\nabla = \frac{2}{3}(E_f - E_i)^{-1} |< \Psi_f| \sum_{j=1}^{N} \vec{\nabla}_j |\Psi_i> |^2$$

(where f ≡ oscillator strength), are reported in Table VIII.

Our oscillator strength calculations with the length operator give fairly good agreement while those calculated with the velocity operator are in poor agreement with the results of Taylor and Segal (oscillator strength equal to 0.02) and with the results of Miller et al. (21) given in Table IX. These later calculations were done with Hartree-Fock orbitals initially beginning with Slater type basis. The discrepancies in our results with the velocity operator may be because we have used

TABLE VII

THE CI EXPANSION COEFFICIENTS, C'S, FOR THE $^1A_1$, $^1B_1$, $^3B_1$ STATES IN THE NINE-DETERMINANTAL BASIS SET

$\Psi \rightarrow$	$^1A_1$	$^1B_1$	$^3B_1$
Δ_1	0.998627	0.0	0.0
Δ_2	-0.018025	0.0	0.0
Δ_3	-0.034514	0.0	0.0
Δ_4	-0.028088	0.0	0.0
Δ_5	-0.020939	0.0	0.0
Δ_6	0.0	-0.053838	0.039556
Δ_7	0.0	0.042462	0.051044
Δ_8	0.0	0.705972	-0.705127
Δ_9	0.0	0.704912	0.706134

TABLE VIII

RESULTS OF OSCILLATOR STRENGTH CALCULATIONS FOR THE $^1A_1 \rightarrow {}^1B_1$ TRANSITION IN WATER

No. of Orbitals	$(E_f - E_i)$ A. U.	f_R	f_∇
8	Experimental 0.271614	0.0255	0.9957
	Theoretical 0.667504	0.0626	0.4052
10	Experimental 0.271614	0.0133	0.3639
	Theoretical 0.459651	0.0224	0.6158

TABLE IX

OSCILLATOR STRENGTH CALCULATED FOR THE $^1A_1 \rightarrow {}^1B_1$ TRANSITION IN H_2O (21)*

Basis Set	f_R	f_∇
O:6, 4 | H:4	0.022	0.050
O:12, 6, 1 | H:5, 2	0.032	0.0623

*The notation O:12, 6, 1, for example, means there are twelve S-type, six P-type and one D-type Slater determinants associated with O-atom.

the Gaussian orbitals behaving as (e^{-r^2}) while the molecular orbitals behave as (e^{-r}). Even though the Gaussian functions may behave pretty accurately at larger distances it is not true for their derivatives because the derivatives of Gaussian orbitals depend upon (re^{-r^2}) while those of molecular orbitals depend only upon (e^{-r}). The improvement in the results can be made by increasing

1) the number of atomic orbitals,
2) the number of Slater determinants in the CI calculations, and
3) by doing another CI calculation to determine the excited state.

REFERENCES

1. Blokhinstev, D. I. Principles of Quantum-Mechanics. Boston: Allyn and Bacon, 1964.
2. Hertzberg, Gerhard. Spectra of Diatomic Molecules. New York: D. Van Nostrand Co., Inc., 1951.
3. Rao, C. M. R. Ultraviolet and Visible Spectroscopy. Butterworth and Co. Publishers Ltd., 1961.
4. Bethe, H. A. and E. E. Salpeter. Intermediate Quantum-Mechanics. New York: W. A. Benjamin, Inc., 1964.
5. Messiah, A. Quantum Mechanics, Vol. II. Amsterdam: North Holland Publishing Co., 1962.
6. Shrivastava, S. K. and P. Westhaus. Electric Dipole Transitions in Carbon, Oklahoma State University, 1971.
7. Hirshfelder, J. O., C. F. Curtiss and R. B. Bird. Molecular Theory of Gases and Liquids. John Wiley and Sons, Inc., 1967.
8. Chandrasekhar. Astrophysics J., Vol. 102, 1945, p. 243.
9. Westhaus, Paul and Oktay Sinanoglu. Phys. Rev., Vol. 183, No. 1, July, 1961.
10. Slater, J. C. Quantum Theory of Atomic Structure, Vol. 1. New York: McGraw-Hill Publ., 1960.
11. Offenhartz, Peter O. Atomic and Molecular Theory. New York: McGraw-Hill Publ., 1970.
12. Kauzmann, W. Quantum Chemistry. New York: Academy Press, Inc., 1957.
13. Freeman, A. J. "Configuration Interaction Study of the Electronic Structure by Atomic and Molecular Method." J. of Chem. Phys., Vol. 28, No. 2, Feb., 1957, p. 230.

14. Oksüz and Sinanoglu. Phys. Rev., Vol. 181, No. 42, 1969, p. 54.

15. Westhaus, P. and Sinanoglu, O. Phys. Rev., Vol. 183, No. 1, 1969, p. 56.

16. Nicholaides and Sinanoglu. Phys. Rev. A., Vol. 4, No. 4, 1971, p. 4.

17. Shrivastava and Westhaus. To be published.

18. Frost, Arther. J. Phys. Chem., Vol. 72, April, 1968, p. 1289.

19. Claydon, C. R., G. A. Segal and H. S. Taylor. "Theoretical Interpretation of the Optical and Electron Scattering Spectra of H_2O." J. Chem. Phys., Vol. 54, May, 1971.

20. Handy, M. C. "The Transcorrelated Method for Accurate Correlation Energies Using Gaussian-Type Functions." Molecular Physics, Vol. 23, No. 1, 1972, p. 1.

21. Miller, Miellzorek and Krauss. J. Chem. Phys., Vol. 51, No. 1, 1969, p. 26.

APPENDIX A

WAVE-MECHANICAL FORMULATION OF THE FRANK-CONDON PRINCIPLE

The "electronic oscillator strength" is a somewhat artificial concept in that the initial and final <u>state</u> of the molecule should be specified by nuclear vibration-rotation - as well as electronic - quantum numbers. The general time evolution theory follows the same outline as presented in the thesis. We are faced with evaluating the transition matrix element,

$$M_{i\rightarrow f} = \left[\int \Psi_f^*(X,x) \left\{ \sum_i \frac{-e\vec{r}_i}{m} + \sum_{\mu=1}^{A} \frac{Z_\mu e\vec{R}_\mu}{M_\mu} \right\} \Psi_i(X,x)\, dX\, dx \right] \tag{A.1}$$

Here, M_μ and $Z_\mu e$ are the mass and charge of Nucleus μ. In principle Ψ_i and Ψ_f are the exact eigenstates of the total molecular Hamiltonian which includes the electronic and nuclear, kinetic and potential energy operators as well as the coupling between them. Notationwise

$$X = \{X_1\, X_2 \ldots X_A\}$$

and

$$x = \{x_1\, x_2 \ldots x_N\}$$

are the configuration space coordinates of the nuclei and electrons. In practice we invoke the Born-Öppenheimer approximation and write

$$\Psi_i \approx \Phi_\ell(x;X)\, V_{\nu;\ell}(X) \qquad \text{(A.2a)}$$

and

$$\Psi_f \approx \Phi_{\ell'}(x;X)\, V_{\nu';\ell'}(X) \qquad \text{(A.2b)}$$

Here, for example, $\Phi_\ell(x,X)$ is the electronic wave function - parameterized by the nuclear configuration - and $V_{\nu;\ell}(X)$ is the ν^{th} nuclear vibration-rotation state in electronic state ℓ. Since we want to consider transitions involving electronic excitation $\ell \neq \ell'$, the orthogonality of the electronic wave functions

$$\int dx\, \Phi^*_{\ell'}(x,X)\, \Phi_\ell(x,X) = 0 \quad ; \quad \ell \neq \ell' \qquad \text{(A.3)}$$

reduces $M_{i\to f}$ in the Born-Öppenheimer Approximation to

$$M_{i\to f} \left[\int dX\, V^*_{\nu';\ell'}(X) \left\{ \int dx\, \Phi^*_{\ell'}(x,X) \sum_{i=1}^{N} \frac{-e\vec{r}_i}{m}\, \Phi_\ell(x,X) \right\} \cdot V_{\nu;\ell}(X) \right] \qquad \text{(A.4)}$$

The expression in braces

$$\vec{O}_{\ell'\ell}(X) = \int dx\, \Phi^*_{\ell'}(x,X) \sum \frac{-e\vec{r}_i}{m}\, \Phi_\ell(x,X) \qquad \text{(A.5)}$$

is what we call the "electronic transition moment". Clearly this vector quantity depends upon the configuration space coordinates of the nuclei. To carry through the calculation in the spirit of the Born-Öppenheimer Approximation requires knowing $\vec{O}_{\ell'\ell}(X)$ over all nuclear configuration space and, then computing the integral Equation (A. 4) as in transition between vibrational states $V_{\nu;\ell}$ and $V_{\nu';\ell'}$.

We shall be concerned with finding $\vec{O}_{\ell'\ell}(X)$ for a single point in nuclear configuration space, X_e, the equilibrium configuration of the ground electronic state of the molecule. Using further approximations one can arrive at an idea of the band-averaged electronic oscillator strength by evaluating

$$f_R = \frac{2}{3}\{U_{\ell'}(X_e) - U_{\ell}(X_e)\}\,|\vec{O}_{\ell'\ell}(X_e)|^2 \tag{A. 6}$$

Here, $U_\ell(X_e)$ and $U_{\ell'}(X_e)$ are the electronic Born-Öppenheimer energies at X_e. The band-average is essentially an average over the final vibrational states in which we replace $(E_{\ell\nu} - E_{\ell'\nu'})$ by $[U_\ell(X_e) - U_{\ell'}(X_e)]$ and ignore any dependence of the electronic transition moment upon the nuclear configuration coordinates. That is, defining the band average

$$f = \sum_{\nu'} f_{\ell\nu \to \ell'\nu'}$$

$$= \Big[\frac{2}{3}\sum_{\nu'}(E_{\ell\nu} - E_{\ell'\nu'}) \cdot \int dX\, V^*_{\ell';\nu'}(X)\vec{O}_{\ell\ell'}(X)\, V_{\nu;\ell} \cdot$$

$$\Big\{\int dX'\, V^*_{\nu';\ell'}(X')\, \vec{O}_{\ell\ell'}(X')\, V_{\nu;\ell}(X')\Big\}^*\Big] \tag{A. 7}$$

We approximate this by

$$\Big[\frac{2}{3}\sum_{\nu'}[U_{\ell}(X_e) - U_{\ell'}(X_e)]\int dX\, V^*_{\nu;\ell}(X)\, V_{\nu';\ell'}(X)\,\cdot$$

$$\int dX'\, V^*_{\nu;\ell}(X')\, V_{\nu';\ell'}(X')\,|\vec{O}_{\ell'\ell}(X_e)|^2\Big]$$

$$= \frac{2}{3}[U_{\ell}(X_e) - U_{\ell'}(X_e)]\,|\vec{O}_{\ell\ell'}(X_e)|^2\int dX\, V^*_{\nu\ell}(X)\, V_{\nu\ell}(X)$$

$$= \frac{2}{3}[U_{\ell}(X_e) - U_{\ell'}(X_e)]\,|\vec{O}_{\ell\ell'}(X_e)|^2 \qquad \text{(A. 8)}$$

Here, we have used the completeness of the vibrational states $\{V_{\nu';\ell'}\}$, whence,

$$\int dX \int dX'\, V^*_{\nu;\ell}(X)[\sum_{\nu'} V_{\nu';\ell'}(X)\, V^*_{\nu';\ell'}(X')]\, V_{\nu;\ell}(X')$$

$$= \int dX\, V^*_{\nu;\ell}(X) \int dX'\, \delta(X - X')\, V_{\nu,\ell}(X')$$

$$= \int dX\, V^*_{\nu;\ell}(X)\, V_{\nu;\ell}(X) = 1 \qquad \text{(A. 9)}$$

Less drastic approximations along these same lines lead to the Frank-Condon principle and the oscillator strength for a given vertical transition within the Born-Öppenheimer scheme as

$$f_{\ell\nu\rightarrow\ell'\nu'} = \frac{2}{3}[E_{\ell\nu} - E_{\ell'\nu'}]\,\Big|\int dX\, V^*_{\nu';\ell'}(x)\, \vec{O}(x)\, V_{\nu;\ell}(x)\Big|^2$$

$$\simeq \frac{2}{3}[E_{\ell\nu} - E_{\ell'\nu'}]|\vec{O}_{\ell\ell'}(X_e)|^2\, \Big|\int dX\, V^*_{\nu';\ell'}(X) V_{\nu;\ell}(X)\Big|^2$$

$$\text{(A. 10)}$$

Here, the "vibration overlap integrals"

$$\int V^*_{\nu';\ell'}(X)\;\; V_{\nu;\ell}(X)\, dX \qquad \text{(A. 11)}$$

are the so-called Frank-Condon factors, tables of which have been compiled by several authors for various forms of the vibrational wave functions. The point to be made here is simply that the electronic transition moments at X_e, $\vec{O}_{\ell\ell'}(X_e)$, which we have evaluated, are also a necessary ingredient in this approximation. There are, of course, other refinements of the theory intermediate between Equation (A. 4) and Equation (A. 10). The natural direction for continuing the work begun in this thesis lies in including the proper vibrational treatment.

APPENDIX B

PROGRAM FOR THE CALCULATION OF TRANSITION PROBABILITY

```
          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
 ISN 0002             SUBROUTINE NIXON
 ISN 0003             IMPLICIT REAL*8(A-H,O-Z)
 ISN 0004             REAL*8 KQ
 ISN 0005             DIMENSION KQ(10)
 ISN 0006             DIMENSION CHAR(80)
 ISN 0007             COMMON /ONB/ OVLP(30,30),OPW(3,3,30,30)
 ISN 0008             COMMON /ORB/ NAORB1,NAORB2,NMORB1,NMORB2
 ISN 0009             COMMON /BT/ B(2,10,10),TAOMAT(7,10,10)
 ISN 0010             COMMON /OPRA/ MLZ(2),KOP1,KOP2,MDELTA
 ISN 0011             COMMON /AUX/ SFAC(20),DFAC(20),COEF(20,20)
 ISN 0012             COMMON  /TP1/   IDEBUG,IPRINT
 ISN 0013             COMMON /BUGMAN/ IBUGAA,IBUGCC,IBUGWW,IBUG1,IBUG2,IBUG3,IBUG4,IBUG5
 ISN 0014             DIMENSION MX(2,10),MY(2,10),MZ(2,10),BETA(2,10),X(2,10),Y(2,10),
                     1   Z(2,10),SUN(10) ,ANSO(10),ANSM(10),ANSP(10)
 ISN 0015             DATA KQ/8H  B(1)   ,8H  B(2)   ,8H  B(3)   ,8H  B(4)   ,8H  B(5)   ,8H
                     1 B(6)   ,8H  B(7)   ,8H  B(8)   ,8H  B(9)   ,8H  B(10)  /
 ISN 0016             PI=3.14159270O
 ISN 0017             SQPI=DSQRT(PI)
 ISN 0018             READ(5,10)IDBUGA,IPRINT
 ISN 0019             READ(5,10)IDBUGB,IDBUGC
 ISN 0020             READ(5,10)    IDEBUG,IPRINT
 ISN 0021             READ(5,10)    IBUGAA,IBUGCC
 ISN 0022             READ(5,10)    IBUGWW,IBUG1
 ISN 0023             READ(5,10)    IBUG2,IBUG3
 ISN 0024             READ(5,10)    IBUG4,IBUG5
 ISN 0025             READ(5,10) KOP1,KOP2
 ISN 0026          10 FORMAT(2I3)
 ISN 0027             DO 14 ISTATE = 1,2
 ISN 0028             READ(5,105) (CHAR(I),I= 1,80)
 ISN 0029             WRITE(6,999)
 ISN 0030             WRITE(6,110) (CHAR(I),I = 1,80)
 ISN 0031             READ(5,10) NAORB,NMORB
 ISN 0032             GO TO (1,2),ISTATE
 ISN 0033           1 NAORB1=NAORB
 ISN 0034             NMORB1=NMORB
 ISN 0035             GO TO 3
 ISN 0036           2 NAORB2=NAORB
 ISN 0037             NMORB2=NMORB
 ISN 0038           3 CONTINUE
 ISN 0039             DO   5  I=1,NAORB
 ISN 0040           5 READ(5,20)MX(ISTATE,I),MY(ISTATE,I),MZ(ISTATE,I),
                     1BETA(ISTATE,I),X(ISTATE,I),Y(ISTATE,I),Z(ISTATE,I)
 ISN 0041          20 FORMAT(3I10,4D10.6)
 ISN 0042             DO 8 MU = 1,NMORB
 ISN 0043             DO 8 I = 1,NAORB
 ISN 0044           8 READ(5,90) B(ISTATE,I,MU)
 ISN 0045             MUDWN = 1
 ISN 0046             MUUP = 4
 ISN 0047          11 IF(MUUP.GT.NMORB) MUUP = NMORB
 ISN 0049             WRITE(6,115) (KQ(MU),MU=MUDWN,MUUP)
 ISN 0050             DO 13 I = 1,NAORB
 ISN 0051          13 WRITE(6,120)I,BETA(ISTATE,I),X(ISTATE,I),Y(ISTATE,I),Z(ISTATE,I),
                     1MX(ISTATE,I),MY(ISTATE,I),MZ(ISTATE,I),(B(ISTATE,I,MU),MU=MUDWN,MU
                     1UP)
 ISN 0052             IF(MUUP.EQ.NMORB) GO TO 14
 ISN 0054             MUDWN = MUUP+1
```

```
ISN 0055          MUUP = MUDWN+3
ISN 0056          GO TO 11
ISN 0057       14 CONTINUE
ISN 0058          DO 15  I=1,NAORB1
ISN 0059          NAX=MX(1,I)
ISN 0060          NAY=MY(1,I)
ISN 0061          NAZ=MZ(1,I)
ISN 0062          XA=X(1,I)
ISN 0063          YA=Y(1,I)
ISN 0064          ZA=Z(1,I)
ISN 0065          AALPHA=BETA(1,I)
ISN 0066          DO 15  J=1,NAORB2
ISN 0067          NBX=MX(2,J)
ISN 0068          NBY=MY(2,J)
ISN 0069          NBZ=MZ(2,J)
ISN 0070          XB=X(2,J)
ISN 0071          YB=Y(2,J)
ISN 0072          ZB=Z(2,J)
ISN 0073          BALPHA=BETA(2,J)
ISN 0074          ALPHA=AALPHA+BALPHA
ISN 0075          NX=NAX+NBX
ISN 0076          NY=NAY+NBY
ISN 0077          NZ=NAZ+NBZ
ISN 0078          IF(IBUG1.NE.IDEBUG) GO TO 29
ISN 0080          WRITE(6,41)ALPHA,NX,NY,NZ
ISN 0081       41 FORMAT(1H0,10X,5HALPHA,10X,2HNX,10X,2HNY,10X,2HNZ /5X,F10.6,4I12/)
ISN 0082       29 CONTINUE
ISN 0083          AMULT=DSQRT(AALPHA*BALPHA/ALPHA)
ISN 0084          XLAMDA=AMULT*(XA-XB)
ISN 0085          YLAMDA=AMULT*(YA-YB)
ISN 0086          ZLAMDA=AMULT*(ZA-ZB)
ISN 0087          TWO=2.0D0
ISN 0088          GAMMA=DSQRT(AALPHA/BALPHA)
ISN 0089          IF(IBUG2.NE.IDEBUG) GO TO 31
ISN 0091          WRITE(6,30)XLAMDA,YLAMDA,ZLAMDA
ISN 0092       30 FORMAT(//20X,6HXLAMDA,10X,6HYLAMDA,10X,6HZLAMDA /10X,3F16.6)
ISN 0093       31 CONTINUE
ISN 0094          EX=DEXP(-(XLAMDA**2+YLAMDA**2+ZLAMDA**2))
ISN 0095          IF(IBUG3.NE.IDEBUG) GO TO 32
ISN 0097          WRITE(6,40)AMULT,GAMMA,EX
ISN 0098       40 FORMAT(//20X,5HAMULT,10X,5HGAMMA,13X,2HEX /10X,3F16.6)
ISN 0099       32 CONTINUE
ISN 0100          ADNOM = ((4.0D0*AALPHA)**(NAX+NAY+NAZ+1))*DSQRT(2.0D0*AALPHA)
ISN 0101          AN = 2.0D0*PI*SQPI*DFAC(2*NAX+1)*DFAC(2*NAY+1)*DFAC(2*NAZ+1)/ADNOM
ISN 0102          AN = DSQRT(AN)
ISN 0103          BDNOM = ((4.0D0*BALPHA)**(NBX+NBY+NBZ+1))*DSQRT(2.0D0*BALPHA)
ISN 0104          BN = 2.0D0*PI*SQPI*DFAC(2*NBX+1)*DFAC(2*NBY+1)*DFAC(2*NBZ+1)/BDNOM
ISN 0105          BN = DSQRT(BN)
ISN 0106          ABNORM=AN*BN
ISN 0107          IF(IBUG4.NE.IDEBUG) GO TO 51
ISN 0109          WRITE(6,999)
ISN 0110          WRITE(6,50) AN,BN,ABNORM
ISN 0111       50 FORMAT(//10X,2HAN,10X,2HBN,10X,6HABNORM/2X,F10.6,2X,F10.6,F16.6/)
ISN 0112       51 CONTINUE
ISN 0113          DO 25 K=1,3
ISN 0114          SUN(K)=0.0D0
ISN 0115          GO TO (21,22,23),K
ISN 0116       21 NA=NAX
```

```
ISN 0117              NB=NBX
ISN 0118              ALAMDA=XLAMDA
ISN 0119              GO TO 24
ISN 0120           22 NA=NAY
ISN 0121              NB=NBY
ISN 0122              ALAMDA=YLAMDA
ISN 0123              GO TO 24
ISN 0124           23 NA=NAZ
ISN 0125              NB=NBZ
ISN 0126              ALAMDA=ZLAMDA
ISN 0127           24 NAP1=NA+1
ISN 0128              NBP1=NB+1
ISN 0129              NBM1=NB-1
ISN 0130              CALL SUMIT(NA,NBM1,GAMMA,ALAMDA,ANS)
ISN 0131              ANSM(K)=ANS
ISN 0132              CALL SUMIT(NA,NB   ,GAMMA,ALAMDA,ANS)
ISN 0133              ANSO(K)=ANS
ISN 0134              CALL SUMIT(NA,NBP1,GAMMA,ALAMDA,ANS)
ISN 0135              ANSP(K)=ANS
ISN 0136              IF(IBUG5.NE.IDEBUG) GO TO 81
ISN 0138              WRITE(6,80) ANSM(K),ANSO(K),ANSP(K)
ISN 0139           80 FORMAT(//20X,7HANSM(K),10X,7HANSO(K),10X,7HANSP(K)/11X,3F16.6)
ISN 0140           81 CONTINUE
ISN 0141           25 CONTINUE
ISN 0142              BNX=NBX
ISN 0143              BNY=NBY
ISN 0144              BNZ=NBZ
ISN 0145              B1=DSQRT(ALPHA**NX)
ISN 0146              B2=DSQRT(ALPHA**NY)
ISN 0147              B3=DSQRT(ALPHA**NZ)
ISN 0148              B4 = B1*ALPHA
ISN 0149              B5 = B2*ALPHA
ISN 0150              B6 = B3*ALPHA
ISN 0151              C1 = EX*ANSO(2)*ANSO(3)/(B2*B3*ALPHA)
ISN 0152              C2 = EX*ANSO(1)*ANSO(3)/(B1*B3*ALPHA)
ISN 0153              C3 = EX*ANSO(1)*ANSO(2)/(B1*B2*ALPHA)
ISN 0154              D1=(ANSM(1)*BNX/B1)-(TWO*BALPHA*ANSP(1)/B4)
ISN 0155              D2=(ANSM(2)*BNY/B2)-(TWO*BALPHA*ANSP(2)/B5)
ISN 0156              D3=(ANSM(3)*BNZ/B3)-(TWO*BALPHA*ANSP(3)/B6)
ISN 0157              E0=DSQRT(ALPHA**(NX+NY+NZ+3))
ISN 0158              TAOMAT(1,I,J)=EX*ANSO(1)*ANSO(2)*ANSO(3)/(ABNORM*E0)
ISN 0159              TAOMAT(2,I,J)=XB*TAOMAT(1,I,J)+C1*ANSP(1)/(ABNORM*B4)
ISN 0160              TAOMAT(3,I,J)=YB*TAOMAT(1,I,J)+C2*ANSP(2)/(ABNORM*B5)
ISN 0161              TAOMAT(4,I,J)=ZB*TAOMAT(1,I,J)+C3*ANSP(3)/(ABNORM*B6)
ISN 0162              TAOMAT(5,I,J)=C1*D1/ABNORM
ISN 0163              TAOMAT(6,I,J)=C2*D2/ABNORM
ISN 0164              TAOMAT(7,I,J)=C3*D3/ABNORM
ISN 0165              IF(IPRINT.EQ.0) GO TO 15
ISN 0167              DO  35 K=1,7
ISN 0168              WRITE(6,70)K,I,J,TAOMAT(K,I,J)
ISN 0169           70 FORMAT(/,2X,' TAOMAT(',I2,',',I2,',',I2,') =  ',F16.6,/)
ISN 0170           35 CONTINUE
ISN 0171           15 CONTINUE
ISN 0172              CALL EXMAT
ISN 0173           90 FORMAT(F16.6)
ISN 0174          105 FORMAT(80A1)
ISN 0175          110 FORMAT(//,80A1,///)
ISN 0176          115 FORMAT(//'  I      BETA         X          Y          Z       NX    NY
```

```
                 INZ ',4(3X,A8,4X),//)
   ISN 0177     120 FORMAT(1X,I3,1X,4(1X,F8.4,1X),3(1X,I3,1X),4(2X,F11.6,2X))
   ISN 0178     999 FORMAT(1H1)
   ISN 0179           RETURN
   ISN 0180           END

*OPTIONS IN EFFECT*       NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*       SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      179 ,PROGRAM SIZE =      7070

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                        97K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                             OS/360  FORTRAN H                                          DATE  73.085/13.53.25

           COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                                   SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002              SUBROUTINE AUXCOE
  ISN 0003              IMPLICIT REAL*8(A-H,O-Z)
  ISN 0004              COMMON /AUX/ SFAC(20),DFAC( 20),COEF(20,20)
  ISN 0005              SFAC(1) = 1.0D0
  ISN 0006              DFAC(1) = 1.0D0
  ISN 0007              DO 210  I=1,11
  ISN 0008          210 SFAC(I+1)=I*SFAC(I)
  ISN 0009              DO 220  I=2,11,2
  ISN 0010              A=I-1
  ISN 0011              DFAC(I+1)=A*DFAC(I-1)
  ISN 0012              DFAC(I)=0.0
  ISN 0013          220 CONTINUE
  ISN 0014              COEF(1,1) = 1.0D0
  ISN 0015              DO  230   I=2,11
  ISN 0016              COEF(I,1) = 1.0D0
  ISN 0017              COEF(1,I) = 0.0D0
  ISN 0018              DO  230   J=2,11
  ISN 0019              IF(I .EQ. J) GO TO 231
  ISN 0021              IF(I .LT. J) GO TO 232
  ISN 0023              IMJ=I-J+1
  ISN 0024              COEF(I,J)=SFAC(I)/(SFAC(J)*SFAC(IMJ))
  ISN 0025              GO TO 233
  ISN 0026          231 COEF(I,J)=1.0D0
  ISN 0027              GO TO 233
  ISN 0028          232 COEF(I,J)=0.0D0
  ISN 0029          233 CONTINUE
  ISN 0030          230 CONTINUE
  ISN 0031              RETURN
  ISN 0032              END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      31 ,PROGRAM SIZE =      766

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                          133K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                    OS/360  FORTRAN H                              DATE  73.085/13.53.43

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002              SUBROUTINE   SUMIT(NA,NB,GAMMA,ALAMDA,ANS)
  ISN 0003              IMPLICIT REAL*8(A-H,O-Z)
  ISN 0004              COMMON /AUX/ SFAC(20),DFAC(20),COEF(20,20)
  ISN 0005              ONEM=-1.0D0
  ISN 0006              SUM=0.0D0
  ISN 0007              PI=3.1415927D0
  ISN 0008              EPS=0.000001D0
  ISN 0009              SQPI=DSQRT(PI)
  ISN 0010              TWO=2.0D0
  ISN 0011              NA1=NA+1
  ISN 0012              NB1=NB+1
  ISN 0013              ZERO=0.0D0
  ISN 0014              IF(DABS(ALAMDA) .LE. EPS) GO TO 11
  ISN 0016              DO 15  IJ=1,NA1
  ISN 0017              I=IJ-1
  ISN 0018              IF(NB .LT. 0) GO TO 12
  ISN 0020              DO 15  JK=1,NB1
  ISN 0021              J=JK-1
  ISN 0022              NMIJ=NA+NB-I-J
  ISN 0023              AUXCO=COEF(NA1,IJ)*COEF(NB1,JK)*DFAC(NMIJ+1)*SQPI/DSQRT(TWO**NMIJ)
  ISN 0024              SUM=SUM+ONEM**I*GAMMA**(J-I)*ALAMDA**(I+J)*AUXCO
  ISN 0025           15 CONTINUE
  ISN 0026              ANS=SUM
  ISN 0027              RETURN
  ISN 0028           11 NAB=NA+NB
  ISN 0029              IF(NAB .LT. 0) GO TO 12
  ISN 0031              ANS=SQPI*DFAC(NAB+1)/DSQRT(TWO**NAB)
  ISN 0032              RETURN
  ISN 0033           12 ANS=0.0D0
  ISN 0034              RETURN
  ISN 0035              END

*OPTIONS IN EFFECT*       NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*       SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      34 ,PROGRAM SIZE =     1108

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                  133K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                             OS/360  FORTRAN H                                   DATE  73.085/13.54.05

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002             SUBROUTINE EXMAT
  ISN 0003             IMPLICIT REAL*8(A-H,O-Z)
  ISN 0004             COMMON /ORB/ NAORB1,NAORB2,NMORB1,NMORB2
  ISN 0005             COMMON /BT/ B(2,10,10),TAOMAT(7,10,10)
  ISN 0006             COMMON /ONB/ OVLP(30,30),OPW(3,3,30,30)
  ISN 0007             COMMON  /TP1/   IDEBUG,IPRINT
  ISN 0008             COMMON /BUGMAN/ IBUGAA,IBUGCC,IBUGWW,IBUG1,IBUG2,IBUG3,IBUG4,IBUG5
  ISN 0009             DO 20  MU=1,NMORB1
  ISN 0010             DO 20  NU=1,NMORB2
  ISN 0011             SUM1=0.0
  ISN 0012             SUM2=0.0
  ISN 0013             SUM3=0.0
  ISN 0014             SUM4=0.0
  ISN 0015             SUM5=0.0
  ISN 0016             SUM6=0.0
  ISN 0017             SUM7=0.0
  ISN 0018             DO 10  I=1,NAORB1
  ISN 0019             DO 10 J=1,NAORB2
  ISN 0020             SUM1=SUM1+B(1,I,MU)*B(2,J,NU)*TAOMAT(1,I,J)
  ISN 0021             SUM2=SUM2+B(1,I,MU)*B(2,J,NU)*TAOMAT(2,I,J)
  ISN 0022             SUM3=SUM3+B(1,I,MU)*B(2,J,NU)*TAOMAT(3,I,J)
  ISN 0023             SUM4=SUM4+B(1,I,MU)*B(2,J,NU)*TAOMAT(4,I,J)
  ISN 0024             SUM5=SUM5+B(1,I,MU)*B(2,J,NU)*TAOMAT(5,I,J)
  ISN 0025             SUM6=SUM6+B(1,I,MU)*B(2,J,NU)*TAOMAT(6,I,J)
  ISN 0026             SUM7=SUM7+B(1,I,MU)*B(2,J,NU)*TAOMAT(7,I,J)
  ISN 0027          10 CONTINUE
  ISN 0028             OVLP(MU,NU)=SUM1
  ISN 0029             OPW(1,1,MU,NU)=SUM2
  ISN 0030             OPW(1,2,MU,NU)=SUM3
  ISN 0031             OPW(1,3,MU,NU)=SUM4
  ISN 0032             OPW(2,1,MU,NU)=SUM5
  ISN 0033             OPW(2,2,MU,NU)=SUM6
  ISN 0034             OPW(2,3,MU,NU)=SUM7
  ISN 0035          20 CONTINUE
  ISN 0036             IF(IPRINT.EQ.0) GO TO 998
  ISN 0038             WRITE(6,999)
  ISN 0039             DO 65  J=1,NMORB2
  ISN 0040             DO 65  I=1,NMORB1
  ISN 0041             WRITE(6,100) I,J,OVLP(I,J)
  ISN 0042             WRITE(6,110)I,J,OPW(1,1,I,J),I,J,OPW(1,2,I,J),I,J,OPW(1,3,I,J)
  ISN 0043             WRITE(6,120)I,J,OPW(2,1,I,J),I,J,OPW(2,2,I,J),I,J,OPW(2,3,I,J)
  ISN 0044          65 CONTINUE
  ISN 0045         100 FORMAT(///10X,'OVLP(',I2,',',I2,')=',F10.6)
  ISN 0046         110 FORMAT(/2X,'OPW(1,1,',I2,',',I2,')=',F10.6,8X,
                      1 'OPW(1,2,',I2,',',I2,')=',F10.6,8X,'OPW(1,3,',I2,',',I2,')=',
                      2 F10.6 ,8X '...COR. TO LENGTH    OPERATOR.')
  ISN 0047         120 FORMAT(/2X,'OPW(2,1,',I2,',',I2,')=',F10.6,8X,
                      1 'OPW(2,2,',I2,',',I2,')=',F10.6,8X,'OPW(2,3,',I2,',',I2,')=',
                      2 F10.6 ,8X '...COR. TO VELOCITY OPERATOR.')
  ISN 0048         998 CONTINUE
  ISN 0049         999 FORMAT(1H1)
  ISN 0050             RETURN
  ISN 0051             END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
```

```
          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002                SUBROUTINE CALCUL
  ISN 0003                IMPLICIT REAL *8 (A-H,O-Z)
  ISN 0004                REAL*8 KQ
  ISN 0005                DIMENSION KQ(4)
  ISN 0006                DIMENSION SUM(3,3,4)
  ISN 0007                COMMON  /TP1/   IDEBUG,IPRINT
  ISN 0008                COMMON /DET1/ IALPHA(2,30),IBETA(2,30),MMANG(2,30),NDET(2),NEL
  ISN 0009                COMMON /DET2/ ISLDET(2,30,15)
  ISN 0010                COMMON /DET3/ CICOEF(2,30,4)
  ISN 0011                COMMON /STA1/ DNORM(2,4),ENG(2,4),EXDE
  ISN 0012                COMMON /TRY1/ NPAIR(2,4),NTYPE(2),NTRY
  ISN 0013                COMMON /UPRA/ MLZ(2),KOP1,KOP2,MDELTA
  ISN 0014                COMMON /CAUSE/ MCGOV
  ISN 0015                COMMON /DEGEN/ XMAB,XMEM
  ISN 0016                COMMON /SWER/ ANS(3,3)
  ISN 0017                DATA KQ/8H  CI(1) ,8H  CI(2) ,8H  CI(3) ,8H  CI(4) /
  ISN 0018                CONST = 2.0D0/3.0D0
                 C        THE ENERGY DIFFERENCE IS IN HARTREES AND LENGTH IN BOHRS
  ISN 0019                EPS = 0.0000010D0
  ISN 0020                DO 5 JCOMP = 1,3
  ISN 0021                DO 5 KOP = KOP1,KOP2
  ISN 0022                DO 5 NN = 1,NTRY
  ISN 0023              5 SUM(JCOMP,KOP,NN) = 0.0D0
  ISN 0024                K1 = NDET(1)
  ISN 0025                L1 = NDET(2)
  ISN 0026                DO 20 K = 1,K1
  ISN 0027                DO 20 L = 1,L1
  ISN 0028                IIAL = IABS(IALPHA(1,K)-IALPHA(2,L))
  ISN 0029                IIBE = IABS(IBETA(1,K)-IBETA(2,L))
  ISN 0030                IF(MCGOV.EQ.0) GO TO 7
  ISN 0032                MM = MMANG(2,L) - MMANG(1,K)
  ISN 0033              7 IF(MDELTA.GT.100) MM=MDELTA
  ISN 0035                IF(IIAL.EQ.0.AND.IIBE.EQ.0.AND.MM.EQ.MDELTA) GO TO 9
  ISN 0037                PRINT 1999
  ISN 0038                PRINT 8,K,L,IIAL,IIBE,MM,MDELTA
  ISN 0039              8 FORMAT( ' INCONSISTENT QUANTUM NUMBERS: DET1 = ',I2,' AND DET2 = '
                         1,I2,' ALPHDIF = ',I2,', BETADIF = ',I2,' MM2-MM1 = ',I2,' DELMZ ='
                         2,I2)
  ISN 0040              9 CONTINUE
  ISN 0041                XTESTT=CICOEF(1,K,1)*CICOEF(2,L,1)
  ISN 0042                IF(DABS(XTESTT).LT.EPS) GO TO 20
  ISN 0044                CALL XPARR(K,L)
  ISN 0045                DO 19 KOP = KOP1,KOP2
  ISN 0046             13 DO 18 NN = 1,NTRY
  ISN 0047                N1 = NPAIR(1,NN)
  ISN 0048                N2 = NPAIR(2,NN)
  ISN 0049                DIV = DNORM(1,N1)*DNORM(2,N2)
  ISN 0050                COF = CICOEF(1,K,N1)*CICOEF(2,L,N2)/DIV
  ISN 0051                SUM(1,KOP,NN) = SUM(1,KOP,NN) + ANS(KOP,1)*COF
  ISN 0052                SUM(2,KOP,NN) = SUM(2,KOP,NN) + ANS(KOP,2)*COF
  ISN 0053                SUM(3,KOP,NN) = SUM(3,KOP,NN) + ANS(KOP,3)*COF
  ISN 0054             18 CONTINUE
  ISN 0055             19 CONTINUE
  ISN 0056             20 CONTINUE
  ISN 0057                DO 104 ISTAT = 1,2
  ISN 0058                PRINT 1999
```

```
ISN 0059            IF(MCOOV.EQ.0) GO TO 103
ISN 0061            GO TO (101,102), ISTAT
ISN 0062        101 PRINT 1001,MLZ(1)
ISN 0063            GO TO 103
ISN 0064        102 PRINT 1002,MLZ(2)
ISN 0065        103 NTTYP = NTYPE(ISTAT)
ISN 0066            PRINT 1005, (KQ(III),III=1,NTTYP)
ISN 0067            NDTUP = NDET(ISTAT)
ISN 0068            DO 104 NDT = 1,NDTUP
ISN 0069            PRINT 1008, NDT,(ISLDET(ISTAT,NDT,NNN),NNN=1,NEL)
ISN 0070            PRINT 1009, (CICOEF(ISTAT,NDT,NTP),NTP=1,NTTYP)
ISN 0071        104 CONTINUE
ISN 0072            DO 30 NN = 1,NTRY
ISN 0073            N1 = NPAIR(1,NN)
ISN 0074            N2 = NPAIR(2,NN)
ISN 0075            PRINT 2000,N1,N2
ISN 0076            DELTAE = ENG(2,N2) - ENG(1,N1)
ISN 0077            DO 29 KOP =KOP1,KOP2
ISN 0078            PRINT 1999
ISN 0079            DSQ = 0.0D0
ISN 0080            DO 22 JCOMP = 1,3
ISN 0081         22 DSQ = DSQ + (SUM(JCOMP,KOP,NN))**2
ISN 0082            GO TO (201,202,203), KOP
ISN 0083        201 PRINT 2001
ISN 0084            FACTH = DELTAE
ISN 0085            FACEX = EXDE
ISN 0086            GO TO 24
ISN 0087        202 PRINT 2002
ISN 0088            FACTH = 1.0D0/DELTAE
ISN 0089            FACEX = 1.0D0/EXDE
ISN 0090            GO TO 24
ISN 0091        203 PRINT 2003
ISN 0092            FACTH = (1.0D0/DELTAE)**3
ISN 0093            FACEX = (1.0D0/EXDE)**3
ISN 0094         24 FABTH = CONST*XMAB*FACTH*DSQ
ISN 0095            FABEX = CONST*XMAB*FACEX*DSQ
ISN 0096            FEMTH = CONST*XMEM*FACTH*DSQ
ISN 0097            FEMEX = CONST*XMEM*FACEX*DSQ
ISN 0098            PRINT 2016, DELTAE
ISN 0099            PRINT 2015, EXDE
ISN 0100            PRINT 2018, DSQ
ISN 0101            PRINT 2020
ISN 0102            PRINT 2021, FABTH
ISN 0103            PRINT 2022, FABEX
ISN 0104            PRINT 2023
ISN 0105            PRINT 2024, FEMTH
ISN 0106            PRINT 2025,FEMEX
ISN 0107         29 CONTINUE
ISN 0108         30 CONTINUE
ISN 0109       1001 FORMAT(//' FOR THE LOWER STATE MLAMBDA = ',I2)
ISN 0110       1002 FORMAT(//' FOR THE UPPER STATE MLAMBDA = ',I2)
ISN 0111       1005 FORMAT(////' DET',12X,'  DETERMINANT   ', 28X,4(3X,A8,4X))
ISN 0112       1008 FORMAT( / I2,5X,15I3)
ISN 0113       1009 FORMAT('+',5X,'A(',45X,')',7X,4(2X,F11.6,2X))
ISN 0114       1999 FORMAT(1H1)
ISN 0115       2000 FORMAT(//' THE LOWER STATE WAVEFUNCTION IS TYPE ',I2,' THE UPPER S
                   1TATE WAVE FUNCTION IS TYPE ',I2)
ISN 0116       2001 FORMAT(//////////20X,'USING THE DIPOLE LENGTH OPERATOR')
```

```
  ISN 0117       2002 FORMAT(//////////20X,'USING THE DIPOLE VELOCITY OPERATOR')
  ISN 0118       2003 FORMAT(//////////20X,'USING THE DIPOLE ACCELERATION OPERATOR')
  ISN 0119       2015 FORMAT(//' THE EXPERIMENTAL ENERGY DIFFERENCE = ',D17.9,' A.U.')
  ISN 0120       2016 FORMAT(/' THE THEORETICAL ENERGY DIFFERENCE = ',D17.9,' A.U.')
  ISN 0121       2018 FORMAT(///' THE SQUARE OF THE DIPOLE MATRIX ELEMENT DSC = ',D17.9,
                     1' (A.U.)**2')
  ISN 0122       2020 FORMAT(////20X,' MULTIPLET ABSORPTION OSCILATOR STRENGTHS')
  ISN 0123       2021 FORMAT(//' USING THEORETICAL ENERGY DIFF, FABTH = ',D17.9)
  ISN 0124       2023 FORMAT(////20X,' MULTIPLET EMMISION OSCILATOR STRENTHS')
  ISN 0125       2022 FORMAT( //' USING EXPERIMENTAL ENERGY DIFF., FABEX = ',D17.9)
  ISN 0126       2024 FORMAT(//' USING  THEORETICAL ENERGY DIFF., FEMTH = ',D17.9)
  ISN 0127       2025 FORMAT(//' USING EXPERIMENTAL ENERGY DIFFERENCE., FEMEX = ',D17.9)
  ISN 0128            RETURN
  ISN 0129            END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =     128 ,PROGRAM SIZE =      4076

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                         109K BYTES OF CORE NOT USED
```

```
LEVEL 21.8 ( MAY 72 )                          OS/360  FORTRAN H                                          DATE  73.085/13.55.11

        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                            SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002            SUBROUTINE WAVEIN
  ISN 0003            IMPLICIT REAL *8 (A-H,O-Z)
  ISN 0004            INTEGER*4 LIN
  ISN 0005            LOGICAL*1 LET(80)
  ISN 0006            COMMON  /TP1/   IDEBUG,IPRINT
  ISN 0007            COMMON /DET1/ IALPHA(2,30),IBETA(2,30),MMANG(2,30),NDET(2),NEL
  ISN 0008            COMMON /DET2/ ISLDET(2,30,15)
  ISN 0009            COMMON /DET3/ CICOEF(2,30,4)
  ISN 0010            COMMON /MOL1/ NPRIN(2,27),NLORB(2,27),NMORB(2,27),NIREP(2,27)
  ISN 0011            COMMON /CRB/ NAORB1,NAORB2,NMORB1,NMORB2
  ISN 0012            COMMON /STA1/ DNORM(2,4),ENG(2,4),EXDE
  ISN 0013            COMMON /TRY1/ NPAIR(2,4),NTYPE(2),NTRY
  ISN 0014            COMMON /OPRA/ MLZ(2),KCP1,KCP2,MDELTA
  ISN 0015            COMMON /CAUSE/ MCGOV
  ISN 0016            COMMON /DEGEN/ XMAB,XMEM
  ISN 0017            JIN = 5
  ISN 0018            READ(JIN,1) LET
  ISN 0019            LIN =  1
  ISN 0020            CALL FINDI4(MCGOV       ,LET,LIN)
  ISN 0021            READ(JIN,1) LET
  ISN 0022            LIN = 1
  ISN 0023            CALL FINDI4(NEL       ,LET,LIN)
  ISN 0024            READ(JIN,1) LET
  ISN 0025            LIN = 1
  ISN 0026            CALL FINDR8(EXDE       ,LET,LIN)
  ISN 0027            DO 100 ISTAT = 1,2
  ISN 0028            IF(MCGOV.EQ.0) GO TO 6
  ISN 0030            GO TO (2,3), ISTAT
  ISN 0031          2 NUP = NMORB1
  ISN 0032            GO TO 4
  ISN 0033          3 NUP = NMORB2
  ISN 0034          4 DO 5 JARB = 1,NUP
  ISN 0035            READ(JIN,1) LET
  ISN 0036            LIN = 1
  ISN 0037            CALL FINDI4(NPRIN(ISTAT,JARB)      ,LET,LIN)
  ISN 0038            CALL FINDI4(NLORB(ISTAT,JARB)      ,LET,LIN)
  ISN 0039            CALL FINDI4(NMORB(ISTAT,JARB)      ,LET,LIN)
  ISN 0040            CALL FINDI4(NIREP(ISTAT,JARB)      ,LET,LIN)
  ISN 0041          5 CONTINUE
  ISN 0042            READ(JIN,1) LET
  ISN 0043            LIN = 1
  ISN 0044            CALL FINDI4(MLZ(ISTAT)       ,LET,LIN)
  ISN 0045          6 CONTINUE
  ISN 0046            READ(JIN,1) LET
  ISN 0047            LIN = 1
  ISN 0048            CALL FINDI4(NDET(ISTAT)     ,LET,LIN)
  ISN 0049            CALL FINDI4(NTYPE(ISTAT)      ,LET,LIN)
  ISN 0050            NTDET = NDET(ISTAT)
  ISN 0051            NTTYP = NTYPE(ISTAT)
  ISN 0052            READ(JIN,1) LET
  ISN 0053            LIN = 1
  ISN 0054            DO 45 NTP = 1,NTTYP
  ISN 0055            CALL FINDR8(ENG(ISTAT,NTP)     ,LET,LIN)
  ISN 0056         45 DNORM(ISTAT,NTP) =0.0D0
  ISN 0057            DO 90 NDT = 1,NTDET
  ISN 0058            MMANG(ISTAT,NDT) = 0
```

```
ISN 0059            READ(JIN,1) LET
ISN 0060            LIN = 1
ISN 0061            DO 75 NNN = 1,NEL
ISN 0062            CALL FINDI4 (ISLDET(ISTAT,NDT,NNN)       ,LET,LIN)
ISN 0063            IF(MCGOV.EQ.0) GO TO 75
ISN 0065            J = ISLDET(ISTAT,NDT,NNN)
ISN 0066            JP = (J+1)/2
ISN 0067            MMANG(ISTAT,NDT) = MMANG(ISTAT,NDT) + NMORB(ISTAT,JP)
ISN 0068         75 CONTINUE
ISN 0069            IF(MCGOV.EQ.0) GO TO 78
ISN 0071            IF(MLZ(ISTAT).EQ.MMANG(ISTAT,NDT)) GO TO 78
ISN 0073            PRINT 1999
ISN 0074            PRINT 76,NDT,MLZ(ISTAT),MMANG(ISTAT,NDT)
ISN 0075         76 FORMAT( ' IN DETERMINANT ',I2,' MZ = ',I2,'; HOWEVER, MLAMBDAZ = '
                   1,I2)
ISN 0076         78 CONTINUE
ISN 0077            CALL REORD(ISTAT,NDT,NEL,PHASE,NALPHA,NBETA)
ISN 0078            IALPHA(ISTAT,NDT) = NALPHA
ISN 0079            IBETA(ISTAT,NDT) = NBETA
ISN 0080            READ(JIN,1) LET
ISN 0081            LIN = 1
ISN 0082            DO 80 NTP = 1,NTTYP
ISN 0083            CALL FINDR8(CICOEF(ISTAT,NDT,NTP)    ,LET,LIN)
ISN 0084            CICOEF(ISTAT,NDT,NTP) = PHASE*CICOEF(ISTAT,NDT,NTP)
ISN 0085         80 DNORM(ISTAT,NTP) = DNORM(ISTAT,NTP) +(CICOEF(ISTAT,NDT,NTP))**2
ISN 0086            IF(IDEBUG.EQ.0) GO TO 5985
ISN 0088            PRINT 1999
ISN 0089            PRINT 5979, (CICOEF(ISTAT,NDT,NTP),NTP=1,NTTYP)
ISN 0090       5979 FORMAT( 4F10.6)
ISN 0091            PRINT 5980,(ISLDET(ISTAT,NDT,NNN),NNN=1,NEL)
ISN 0092       5980 FORMAT( 10I3)
ISN 0093            PRINT 5981, PHASE,IALPHA(ISTAT,NDT),IBETA(ISTAT,NDT)
ISN 0094       5981 FORMAT( ' PHASE,IALPHA,IBETA = ', F10.5,4X,2I4)
ISN 0095       5985 CONTINUE
ISN 0096         90 CONTINUE
ISN 0097            DO 95 NTP = 1,NTTYP
ISN 0098         95 DNORM(ISTAT,NTP) = DSQRT(DNORM(ISTAT,NTP))
ISN 0099        100 CONTINUE
ISN 0100            MDELTA=150
ISN 0101            IF(MCGOV .NE. 0) GO TO 102
ISN 0103            READ(JIN,1)LET
ISN 0104            LIN=1
ISN 0105            CALL FINDR8(XMAB      ,LET,LIN)
ISN 0106            CALL FINDR8(XMEM      ,LET,LIN)
ISN 0107            GO TO 103
ISN 0108        102 CONTINUE
ISN 0109            DLG = 0.0D0
ISN 0110            DLX = 0.0D0
ISN 0111            IF(MLZ(1).EQ.0) DLG = 1.0D0
ISN 0113            IF(MLZ(2).EQ.0) DLX = 1.0D0
ISN 0115            XMAB = (2.0D0-DLG*DLX)/(2.0D0-DLG)
ISN 0116            XMEM = (2.0D0-DLG*DLX)/(2.0D0-DLX)
ISN 0117            MDELTA = MLZ(2) - MLZ(1)
ISN 0118        103 CONTINUE
ISN 0119            READ(JIN,1)  LET
ISN 0120            LIN = 1
ISN 0121            CALL FINDI4(NTRY        ,LET,LIN)
ISN 0122            DO 105 ISTAT = 1,2
```

```
   ISN 0123            READ(JIN,1) LET
   ISN 0124            LIN = 1
   ISN 0125            DO 104 NN = 1,NTRY
   ISN 0126            CALL FINDI4(NPAIR(ISTAT,NN)       ,LET,LIN)
   ISN 0127        104 CONTINUE
   ISN 0128        105 CONTINUE
   ISN 0129          1 FORMAT(80A1)
   ISN 0130       1999 FORMAT(1H1)
   ISN 0131            RETURN
   ISN 0132            END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =     131 ,PROGRAM SIZE =     2986

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                          113K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                  OS/360  FORTRAN H                                     DATE  73.085/13.55.40

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                                    SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002               SUBROUTINE REORD(NST,NDT,NEL,PHASE,NALPHA,NBETA)
  ISN 0003               IMPLICIT REAL *8 (A-H,O-Z)
  ISN 0004               DIMENSION IODD(10),IEVEN(10)
  ISN 0005               COMMON /DET2/ ISLDET(2,30,15)
  ISN 0006               PHASE = 1.0D0
  ISN 0007               NALPHA = 0
  ISN 0008               NBETA = 0
  ISN 0009               DO 50 I = 1,NEL
  ISN 0010               IORB = ISLDET(NST,NDT,I)
  ISN 0011               ITEST = IORB - 2*(IORB/2)
  ISN 0012               IF(ITEST) 711,30,40
  ISN 0013           711 PRINT 150
  ISN 0014               GO TO 50
  ISN 0015            30 NBETA = NBETA + 1
  ISN 0016               IEVEN(NBETA) = IORB
  ISN 0017               GO TO 50
  ISN 0018            40 NALPHA = NALPHA + 1
  ISN 0019               IODD(NALPHA) = IORB
  ISN 0020               PHASE = PHASE*(-1.0D0)**(I-NALPHA)
  ISN 0021            50 CONTINUE
  ISN 0022               N = 0
  ISN 0023               IF(NALPHA.EQ.0) GO TO 126
  ISN 0025               DO 125 I = 1,NALPHA
  ISN 0026               N = N+1
  ISN 0027           125 ISLDET(NST,NDT,N) = IODD(I)
  ISN 0028               IF(NBETA.EQ.0) GO TO 131
  ISN 0030           126 DO 130 J = 1,NBETA
  ISN 0031               N = N+1
  ISN 0032           130 ISLDET(NST,NDT,N) = IEVEN(J)
  ISN 0033           131 CONTINUE
  ISN 0034               RETURN
  ISN 0035           150 FORMAT( 'THIS IS IMPOSSIBLE. CHECK REORDER')
  ISN 0036               RETURN
  ISN 0037               END

*OPTIONS IN EFFECT*       NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*       SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       36 ,PROGRAM SIZE =      1032

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                          137K BYTES OF CORE NOT USED
```

```
          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002              SUBROUTINE XPARR(K,L)
  ISN 0003              IMPLICIT REAL *8 (A-H,O-Z)
  ISN 0004              DIMENSION D(10,10),DT(10,10),DTD(10,10),DDT(10,10)
  ISN 0005              DIMENSION WM(3,3,10,10),RES(3,3)
  ISN 0006              DIMENSION EIV1(10),EIV2(10),U(10,10),V(10,10)
  ISN 0007              DIMENSION TEMP(10,10)
  ISN 0008              DIMENSION UPR(10,10),VPR(10,10)
  ISN 0009              COMMON /DET1/ IALPHA(2,30),IBETA(2,30),MMANG(2,30),NDET(2),NEL
  ISN 0010              COMMON /DET2/ ISLDET(2,30,15)
  ISN 0011              COMMON /BUGMAN/ IBUGAA,IBUGCC,IBUGWW,IBUG1,IBUG2,IBUG3,IBUG4,IBUG5
  ISN 0012              COMMON  /TP1/   IDEBUG,IPRINT
  ISN 0013              COMMON /ONB/ CVLP(30,30),CPW(3,3,30,30)
  ISN 0014              COMMON /OPRA/ MEZ(2),KOP1,KOP2,MDELTA
  ISN 0015              COMMON /SWER/ ANS(3,3)
  ISN 0016              EPSI = 0.000001D0
  ISN 0017              DO 7007 JCOMP = 1,3
  ISN 0018              DO 7007 KOP = 1,3
  ISN 0019              ANS(KOP,JCOMP) = 0.0D0
  ISN 0020         7007 RES(KOP,JCOMP) = 0.0D0
  ISN 0021              NALPHA = IALPHA(1,K)
  ISN 0022              NBETA = IBETA(2,L)
  ISN 0023              IF(MDELTA.GT.100) GO TO 4
  ISN 0025              IF(MDELTA) 3,2,1
  ISN 0026            1 JCOMPD = 1
  ISN 0027              JCOMPU = 1
  ISN 0028              GO TO 5
  ISN 0029            2 JCOMPD = 3
  ISN 0030              JCOMPU = 3
  ISN 0031              GO TO 5
  ISN 0032            3 JCOMPD = 2
  ISN 0033              JCOMPU = 2
  ISN 0034              GO TO 5
  ISN 0035            4 JCOMPD = 1
  ISN 0036              JCOMPU = 3
  ISN 0037            5 CONTINUE
  ISN 0038              IF(IBUG4.EQ.0) GO TO 4301
  ISN 0040              PRINT 1999
  ISN 0041         1999 FORMAT(1H1)
  ISN 0042              PRINT 4300,K,L,MDELTA,JCOMPD,JCOMPU
  ISN 0043         4300 FORMAT(/'  IN XPARR WITH DETG = ',I2,' AND DETX = ',I2,', WE HAVE
                       1MDELTA = ',I3,' AND JCOMPD = ',I3,' AND JCOMPU = ',I3,//)
  ISN 0044         4301 CONTINUE
  ISN 0045              DO 100 NBLCK = 1,2
  ISN 0046              GO TO (11,12), NBLCK
  ISN 0047           11 NDOWN = 1
  ISN 0048              NUP = NALPHA
  ISN 0049              NDM = NALPHA
  ISN 0050              IF(NDM.EQ.0) GO TO 63
  ISN 0052              GO TO 20
  ISN 0053           12 NDOWN = NALPHA + 1
  ISN 0054              NUP = NALPHA + NBETA
  ISN 0055              NDM = NBETA
  ISN 0056              IF(NDM.EQ.0) GO TO 64
  ISN 0058           20 I = 0
  ISN 0059              DO 50 N = NDOWN,NUP
  ISN 0060              I = I + 1
```

```
ISN 0061            J = 0
ISN 0062            DO 50 M = NDOWN,NUP
ISN 0063            J = J+1
ISN 0064            IP = (ISLDET(1,K,N) + 1)/2
ISN 0065            JP = (ISLDET(2,L,M) + 1)/2
ISN 0066         50 D(I,J) = OVLP(IP,JP)
ISN 0067            GO TO (61,62), NBLCK
ISN 0068         61 DALPHA = DET(NDM,D)
ISN 0069            GO TO 100
ISN 0070         62 DBETA = DET(NDM,D)
ISN 0071            GO TO 100
ISN 0072         63 DALPHA = 1.0D0
ISN 0073            GO TO 100
ISN 0074         64 DBETA = 1.0D0
ISN 0075        100 CONTINUE
ISN 0076            IF(IBUG4.EQ.0) GO TO 4112
ISN 0078            PRINT 4110, DALPHA,DBETA
ISN 0079       4110 FORMAT( ' DALPHA = ', D13.6, ' DBETA = ', D13.6,/)
ISN 0080       4112 CONTINUE
ISN 0081            NB1 = 1
ISN 0082            NB2 = 2
ISN 0083            IF(DABS(DALPHA).LT.EPSI) NB2 = 1
ISN 0085            IF(DABS(DBETA).LT.EPSI) NB1 = 2
ISN 0087            IF(NB1.EQ.2.AND.NB2.EQ.1) GO TO 1500
ISN 0089            DO 1000 NBLCK = NB1,NB2
ISN 0090            DO 931  JCOMP=1,3
ISN 0091            DO 931  KOP=1,3
ISN 0092        931 RES(KOP,JCOMP)=0.0
ISN 0093            GO TO (101,102), NBLCK
ISN 0094        101 NDOWN = 1
ISN 0095            NUP = NALPHA
ISN 0096            DDD = DALPHA
ISN 0097            DDOTH = DBETA
ISN 0098            NDM = NALPHA
ISN 0099            IF(NDM.EQ.0) GO TO 1000
ISN 0101            GO TO 103
ISN 0102        102 NDOWN = NALPHA + 1
ISN 0103            NUP = NALPHA + NBETA
ISN 0104            DDD = DBETA
ISN 0105            DDOTH = DALPHA
ISN 0106            NDM = NBETA
ISN 0107            IF(NDM.EQ.0) GO TO 1000
ISN 0109        103 I = 0
ISN 0110            DO 150 N = NDOWN,NUP
ISN 0111            I = I + 1
ISN 0112            J = 0
ISN 0113            DO 150 M = NDOWN,NUP
ISN 0114            J = J + 1
ISN 0115            IP = (ISLDET(1,K,N) + 1)/2
ISN 0116            JP = (ISLDET(2,L,M) + 1)/2
ISN 0117            D(I,J) = OVLP(IP,JP)
ISN 0118            DO 145 JCOMP = JCOMPD,JCOMPU
ISN 0119            DO 145 KOP = KOP1,KOP2
ISN 0120        145 WM(KOP,JCOMP,I,J) = OPW(KOP,JCOMP,IP,JP)
ISN 0121        150 DT(J,I) = D(I,J)
ISN 0122            CALL MATMUL(NDM,D,DT,DDT)
ISN 0123            CALL MATMUL(NDM,DT,D,DTD)
ISN 0124            IF(IBUG4.EQ.0) GO TO 4445
```

```
ISN 0126              PRINT 4002
ISN 0127         4002 FORMAT(/' THE MATRIX D ',/)
ISN 0128              DO 4010 IWW = 1,NDM
ISN 0129              PRINT 4444, (D(IWW,JWW),JWW=1,NDM)
ISN 0130         4010 CONTINUE
ISN 0131              PRINT 4012
ISN 0132         4012 FORMAT(/' THE MATRIX DT ',/)
ISN 0133              DO 4020 IWW = 1,NDM
ISN 0134              PRINT 4444, (DT(IWW,JWW),JWW=1,NDM)
ISN 0135         4020 CONTINUE
ISN 0136              DO 4030 JCOMP = JCOMPD,JCOMPU
ISN 0137              DO 4030   KOP=KOP1,KOP2
ISN 0138              PRINT 4022, KOP,JCOMP
ISN 0139         4022 FORMAT(/' THE MATRIX WM FOR KOP = ',I3,' AND JCOMP = ',I3,/)
ISN 0140              DO 4030 IWW = 1,NDM
ISN 0141              PRINT 4444, (WM(KOP,JCOMP,IWW,JWW),JWW = 1,NDM)
ISN 0142         4030 CONTINUE
ISN 0143              PRINT 4202
ISN 0144         4202 FORMAT(/' THE MATRIX DTD ',/)
ISN 0145              DO 4210 IWW = 1,NDM
ISN 0146              PRINT 4444, (DTD(IWW,JWW),JWW= 1,NDM)
ISN 0147         4210 CONTINUE
ISN 0148              PRINT 4212
ISN 0149         4212 FORMAT(/' THE MATRIX DDT ',/)
ISN 0150              DO 4220 IWW = 1,NDM
ISN 0151              PRINT 4444, ( DDT(IWW,JWW),JWW=1,NDM)
ISN 0152         4220 CONTINUE
ISN 0153         4444 FORMAT( 8(2X,D13.6))
ISN 0154         4445 CONTINUE
ISN 0155              DO 505 I = 1,NDM
ISN 0156              DO 505 J = 1,NDM
ISN 0157              V(I,J) = -DTD(I,J)
ISN 0158          505 U(I,J) = -DDT(I,J)
ISN 0159              IF(DABS(DDD).GT.EPSI) GO TO 892
ISN 0161              IF(NDM.EQ.1) GO TO 515
ISN 0163              NCALL = 2
ISN 0164              NSHOUT = 1
ISN 0165              CALL EIGEN(V,TEMP,NDM,EIV1,NCALL,VNORM,10)
ISN 0166              DO 506 IGA1 = 1,NCALL
ISN 0167          506 EIV1(IGA1) = -EIV1(IGA1)
ISN 0168              IF(DABS(EIV1(2)).LT.EPSI) GO TO 899
ISN 0170          510 CALL EIGEN(U,TEMP,NDM,EIV2,NSHOUT,UNORM,10)
ISN 0171              DO 512 IGA2 = 1,NSHOUT
ISN 0172          512 EIV2(IGA2) = -EIV2(IGA2)
ISN 0173              GO TO 520
ISN 0174          515 EIV1(1) = 0.0D0
ISN 0175              EIV2(1) = 0.0D0
ISN 0176              U(1,1) = 1.0D0
ISN 0177              V(1,1) = 1.0D0
ISN 0178              NCALL = 1
ISN 0179              NSHOUT = 1
ISN 0180          520 CONTINUE
ISN 0181              DO 550 IP = 1,NDM
ISN 0182              DO 550 JP = 1,NDM
ISN 0183              COF = U(IP,1)*V(JP,1)
ISN 0184              D(IP,JP) = D(IP,JP) + COF
ISN 0185              DO 550 KOP = KOP1,KOP2
ISN 0186              DO 550 JCOMP = JCOMPD,JCOMPU
```

```
ISN 0187          550 RES(KOP,JCOMP) = RES(KOP,JCOMP) + COF*WM(KOP,JCOMP,IP,JP)
ISN 0188              PP = DET(NDM,D)
ISN 0189              GO TO 900
ISN 0190          892 CONTINUE
ISN 0191              NCALL = NDM
ISN 0192              NSHOUT = NDM
ISN 0193              CALL EIGEN(V,TEMP,NDM,EIV1,NDM,VNORM,10)
ISN 0194              CALL EIGEN(U,TEMP,NDM,EIV2,NDM,UNORM,10)
ISN 0195              DO 894 J = 1,NDM
ISN 0196              IF(DABS(EIV1(J)-EIV2(J)).GT.EPSI) GO TO 8711
ISN 0198              GO TO 893
ISN 0199         8711 PRINT 8712, J,EIV1(J),EIV2(J)
ISN 0200         8712 FORMAT( ' TROUBLE AT J = ',I2,'EIV1 = ',F12.9,'AND EIV2 = ',F12.9)
ISN 0201          893 EIV1(J) = -EIV1(J)
ISN 0202              EIV2(J)=-EIV2(J)
ISN 0203          894 CONTINUE
ISN 0204              DO 895 I = 1,NDM
ISN 0205              DO 895 J = 1,NDM
ISN 0206          895 UPR(I,J) = U(J,I)
ISN 0207              CALL MATMUL(NDM,UPR,D,VPR)
ISN 0208              CALL MATMUL(NDM,VPR,V,UPR)
ISN 0209              DO 898 JCOMP = JCOMPD,JCOMPU
ISN 0210              DO 898 KOP = KOP1,KOP2
ISN 0211              DO 897 I = 1,NDM
ISN 0212              STRM = 0.0D0
ISN 0213              DO 896 J1 = 1,NDM
ISN 0214              DO 896 J2 = 1,NDM
ISN 0215          896 STRM=STRM+U(J1,I)*V(J2,I)*WM(KOP,JCOMP,J1,J2)
ISN 0216          897 RES(KOP,JCOMP)=RES(KOP,JCOMP)+STRM/UPR(I,I)
ISN 0217          898 CONTINUE
ISN 0218              PP = DET(NDM,D)
ISN 0219              GO TO 900
ISN 0220          899 PP = 0.0D0
ISN 0221          900 DO 905 JCOMP = JCOMPD,JCOMPU
ISN 0222              DO 905 KOP = KOP1,KOP2
ISN 0223          905 ANS(KOP,JCOMP)=ANS(KOP,JCOMP)+DDOTH*PP*RES(KOP,JCOMP)
ISN 0224              IF(IBUG4.EQ.0) GO TO 5140
ISN 0226              PRINT 5121,K,L
ISN 0227         5121 FORMAT(/'  IN XPAR WE HAVE DETG = ',I3,' AND DETX = ',I3,/)
ISN 0228              PRINT 5122,  NBLCK,NDM,DDD,DDDTH,DDU,DDV,PP
ISN 0229         5122 FORMAT(/'  THIS IS BLOCK ',I1,' WITH NDM =',I2,' AND DDD =',D13.6,
                    1 '  AND DDTH = ',D13.6  /'  DDU =',D13.6, 'AND DDV = ',D13.6,' AND
                    2  PP=',D13.6,/)
ISN 0230              PRINT 5124
ISN 0231         5124 FORMAT(/'  THE EIGENVALUES OF DTD ARE: ',/)
ISN 0232              PRINT 5144, ( EIV1(IXX),IXX=1,NCALL)
ISN 0233              PRINT 5125
ISN 0234         5125 FORMAT(/'  THE EIGENVALUES OF DDT ARE:  ',/)
ISN 0235              PRINT 5144, ( EIV2(IXX),IXX=1,NSHOUT)
ISN 0236               PRINT 5127, NCALL
ISN 0237         5127 FORMAT(/'  THE FIRST ',I2,' EIGENVECTORS OF DTD ARE: ',/)
ISN 0238              DO 5130 IXX = 1,NDM
ISN 0239              PRINT 5144,( V(IXX,JXX),JXX=1,NCALL)
ISN 0240         5130 CONTINUE
ISN 0241              PRINT 5132, NSHOUT
ISN 0242         5132 FORMAT(/'  THE FIRST ',I2,' EIGENVECTORS OF DDT ARE: ',/)
ISN 0243              DO 5135 IXX = 1,NDM
ISN 0244              PRINT 5144,(U(IXX,JXX),JXX=1,NSHOUT)
```

```
  ISN 0245       5135 CONTINUE
  ISN 0246            DO 5138 KOP = KOP1,KOP2
  ISN 0247            PRINT 5137, (KOP,JCOMP,ANS(KOP,JCOMP),JCOMP=JCOMPD,JCOMPU)
  ISN 0248       5137 FORMAT(/,3( '  ANS(',I2,',',I2,') = ',D13.6))
  ISN 0249       5138 CONTINUE
  ISN 0250       5144 FORMAT( 8(2X,D13.6))
  ISN 0251       5140 CONTINUE
  ISN 0252       1000 CONTINUE
  ISN 0253            RETURN
  ISN 0254       1500 DO 1505 JCOMP = 1,3
  ISN 0255            DO 1505 KOP = 1,3
  ISN 0256            ANS(KOP,JCOMP) = 0.000
  ISN 0257            PRINT 3994, KOP,JCOMP,ANS(KOP,JCOMP)
  ISN 0258       3994 FORMAT(  ' ANS(',I1,',',I1,') = ',F10.6)
  ISN 0259       1505 CONTINUE
  ISN 0260            RETURN
  ISN 0261            END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =      260 ,PROGRAM SIZE =    21540

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                   65K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                                   OS/360  FORTRAN H                                          DATE  73.085/13.56.33

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002            SUBROUTINE MATMUL(LIM,A,B,C)
  ISN 0003            IMPLICIT REAL *8 (A-H,O-Z)
  ISN 0004            DIMENSION A(10,10),B(10,10),C(10,10)
  ISN 0005            DO 10 I = 1,LIM
  ISN 0006            DO 10 J = 1, LIM
  ISN 0007            C(I,J) = 0.0D0
  ISN 0008            DO 10 K = 1,LIM
  ISN 0009         10 C(I,J) = C(I,J) + A(I,K)*B(K,J)
  ISN 0010            RETURN
  ISN 0011            END

*OPTIONS IN EFFECT*       NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*       SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*      SOURCE STATEMENTS =       10 ,PROGRAM SIZE =       546

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                            137K BYTES OF CORE NOT USED
```

```
LEVEL 21.6 ( MAY 72 )                          OS/360  FORTRAN H                                   DATE  73.085/13.55.56

          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002              FUNCTION DET(JJUP,A)
  ISN 0003              IMPLICIT REAL *8 (A-H,O-Z)
                  C     THIS SUBROUTINE COMPUTES THE DETERMINANT OF THE MATRIX  A  BY THE
                  C     'SWEEP OUT' METHOD.  IN THE PROCESS THE MATRIX  A  IS DESTROYED.
  ISN 0004              DIMENSION A(10,10)
  ISN 0005              QXZ = 1.0D0
  ISN 0006              DO 1190 I = 1,JJUP
  ISN 0007              BIGA = A(I,I)
  ISN 0008              L = I
  ISN 0009              DO 1120 J = I,JJUP
  ISN 0010              IF(DABS(A(J,I))-DABS(BIGA)) 1120,1120,1110
  ISN 0011         1110 BIGA = A(J,I)
  ISN 0012              L = J
  ISN 0013         1120 CONTINUE
  ISN 0014              IF(BIGA) 1130,1125,1130
  ISN 0015         1125 DET = 0.0D0
  ISN 0016              RETURN
  ISN 0017         1130 PHASE = 1.0D0
  ISN 0018              IF(L-I) 1150,1150,1140
  ISN 0019         1140 DO 1145 K = I,JJUP
  ISN 0020              HOLD = A(I,K)
  ISN 0021              A(I,K) = A(L,K)
  ISN 0022         1145 A(L,K) = HOLD
  ISN 0023              PHASE = -1.0D0
  ISN 0024         1150 DO 1155 K = I,JJUP
  ISN 0025         1155 A(K,I) = A(K,I)/BIGA
  ISN 0026              MIN = I + 1
  ISN 0027              IF(JJUP-MIN) 1190, 1160, 1160
  ISN 0028         1160 DO 1170 M = MIN,JJUP
  ISN 0029              X = A(M,I)
  ISN 0030              DO 1170 N = I,JJUP
  ISN 0031         1170 A(M,N) = A(M,N) - A(I,N)*X
  ISN 0032         1190 QXZ = QXZ*PHASE*BIGA
  ISN 0033              DET = QXZ
  ISN 0034              RETURN
  ISN 0035              END

*OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*     SOURCE STATEMENTS =       34 ,PROGRAM SIZE =       960

*STATISTICS*  NO  DIAGNOSTICS GENERATED

****** END OF COMPILATION ******                                         133K BYTES OF CORE NOT USED
```

```
          COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                             SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002               SUBROUTINE FINDI4(I4,LET,I)
                   C        FREE FORMAT INPUT SUBROUTINE FOR FORTRAN INPUT
  ISN 0003               DIMENSION LET(80),NUM(10),D(5)
  ISN 0004               INTEGER*4 FLAG,EXP,ESIGN,FIRST,SIGN,ENTER,EXPFR,END
  ISN 0005               REAL*8 R8,FRACT,FAC,TEMP,X,INTGER
  ISN 0006               LOGICAL*1 LET,NUM/'0','1','2','3','4','5','6','7','8','9'/,
                        1          BLANK/' '/,LETE/'E'/,PLUS/'+'/,MINUS/'-'/,DECPT/'.'/,
                        2          LETD/'D'/,L,D
  ISN 0007               EQUIVALENCE (D,TEMP,ITEMP)
                   C                                   ENTRY STATEMENTS
  ISN 0008               ENTER=1
  ISN 0009               GO TO 100
  ISN 0010               ENTRY FINDR4(R4,LET,I)
  ISN 0011               ENTER=2
  ISN 0012               GO TO 100
  ISN 0013               ENTRY FINDR8(R8,LET,I)
  ISN 0014               ENTER=3
                   C                                 INITIALIZE VARIABLES
  ISN 0015           100 SIGN=1
  ISN 0016               FLAG=1
  ISN 0017               ISIGN=1
  ISN 0018               ESIGN=1
  ISN 0019               TEMP=0.D0
  ISN 0020               INTGER=0.D0
  ISN 0021               FRACT=0.D0
  ISN 0022               EXPFR=0
  ISN 0023               EXP=0
  ISN 0024               END=0
                   C                           BEGIN LOOP TO DECODE CARD CHARACTERS
  ISN 0025           128 FIRST=0
  ISN 0026           102 IF (I.GT.80) GO TO 110
  ISN 0028               L=LET(I)
  ISN 0029               IF (L.EQ.PLUS) GO TO 107
  ISN 0031               IF (L.EQ.MINUS) GO TO 106
  ISN 0033               IF (L.EQ.LETE.OR.L.EQ.LETD) GO TO 104
  ISN 0035               IF (L.EQ.DECPT) GO TO 109
  ISN 0037               IF (L.EQ.BLANK) GO TO 111
  ISN 0039               IF (FIRST.EQ.0) FIRST=I
                   C     END=1 IF A NON-BLANK CHARACTER HAS BEEN FOUND
                   C     FIRST = LOCATION IN LET OF FIRST CHARACTER OF DIGIT STRING
                   C     LAST = LOCATION IN LET OF LAST CHARACTER OF DIGIT STRING
                   C     FLAG =1 FOR INTEGER DIGIT STRING
                   C     FLAG =2 FOR DECIMAL FRACTION DIGIT STRING
                   C     FLAG =3 FOR EXPONENT DIGIT STRING
  ISN 0041           107 END=1
  ISN 0042           101 I=I+1
  ISN 0043               GO TO 102
  ISN 0044           105 IF (END) 101,101,117
  ISN 0045           106 SIGN=-1
  ISN 0046               GO TO 107
  ISN 0047           104 IF (FIRST.NE.0) GO TO 119
  ISN 0049               IF (END.EQ.0) INTGER=1.D0
  ISN 0051               IF (LET(I+1).EQ.BLANK) I=I+1
  ISN 0053               SIGN=1
  ISN 0054               FLAG=3
  ISN 0055               GO TO 107
```

```
ISN 0056         109 IF (FIRST.NE.0) GO TO 119
ISN 0058             ISIGN=SIGN
ISN 0059             FLAG=2
ISN 0060             GO TO 107
ISN 0061         110 IF (END.EQ.0) I=82
ISN 0063             END=1
ISN 0064         111 IF (FIRST.EQ.0) GO TO 105
               C                       DECODE DIGIT STRING AND STORE IN X
ISN 0066         119 LAST=I-1
ISN 0067             X=0.D0
ISN 0068             FAC=1.D0
ISN 0069         115 DO 116 J=1,10
ISN 0070             IF (LET(LAST).EQ.NUM(J)) GO TO 118
ISN 0072         116 CONTINUE
ISN 0073             PRINT 200,LET,LAST
ISN 0074         200 FORMAT(1X,'FIND ERROR-ILLEGAL CHARACTER IN INPUT FIELD'/1X,80A1,
                   1         5X'COL.='I2)
ISN 0075             STOP
ISN 0076         118 X=X+FAC*(J-1)
ISN 0077             IF (LAST.EQ.FIRST) GO TO 114
ISN 0079             LAST=LAST-1
ISN 0080             FAC=10.D0*FAC
ISN 0081             GO TO 115
               C                     STORE X,SIGN AND GO ON TO THE NEXT DIGIT STRING
ISN 0082         114 GO TO (125,126,127),FLAG
ISN 0083         125 INTGER=X
ISN 0084             ISIGN=SIGN
ISN 0085             GO TO 128
ISN 0086         126 FRACT=X
ISN 0087             EXPFR=FIRST-I
ISN 0088             GO TO 128
ISN 0089         127 EXP=X
ISN 0090             ESIGN=SIGN
               C                   COMBINE DECODED DIGIT STRINGS TO FORM CONSTANT
ISN 0091         117 TEMP=ISIGN*(INTGER+FRACT*(10.D0**EXPFR))*(10.D0**(ESIGN*EXP))
ISN 0092         103 GO TO (131,132,133),ENTER
               C                                 MAKE UP INTEGER*4 CONSTANT
ISN 0093         131 IFRACT=FRACT
ISN 0094             INT=INTGER
ISN 0095             I1=10**EXP
ISN 0096             I2=EXPFR+ESIGN*EXP
ISN 0097             I3=10**IABS(I2)
ISN 0098             IF (ESIGN.LT.0) GO TO 135
ISN 0100             IF (I2.LT.0) GO TO 136
ISN 0102             I5=IFRACT*I3
ISN 0103             GO TO 137
ISN 0104         136 I5=IFRACT/I3
ISN 0105         137 I4=ISIGN*(INT*I1+I5)
ISN 0106             GO TO 134
ISN 0107         135 I4=ISIGN*INT/I1
ISN 0108             GO TO 134
               C                   MAKE UP REAL*4 CONSTANT BY ROUNDING AND TRUNCATING TEMP
ISN 0109         132 L=D(1)
ISN 0110             IF (D(5).GT.128) ITEMP=ITEMP+1
ISN 0112             IF (L.NE.D(1)) ITEMP=ITEMP+1048576
ISN 0114             R4=TEMP
ISN 0115             GO TO 134
               C                             MAKE UP REAL*8 CONSTANT = TEMP
```

```
ISN 0116        133 R8=TEMP
ISN 0117        134 RETURN
ISN 0118            END
```

```
                                          FORTRAN H ERROR MESSAGES                                          PAGE 004

                 ERROR NO   LEVEL        ERROR MESSAGE

     ISN     0007   IEK061I       4        THE EQUIVALENCE STATEMENT CONTAINS A NON-SUBSCRIPTED ARRAY ITEM. INCORRE
                                           CT ADCONS MAY BE GENERATED.

*OPTIONS IN EFFECT*       NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

*OPTIONS IN EFFECT*       SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

*STATISTICS*      SOURCE STATEMENTS =      117 ,PROGRAM SIZE =      2236

*STATISTICS*     1 DIAGNOSTICS GENERATED, HIGHEST SEVERITY CODE IS  4

****** END OF COMPILATION ******                                        121K BYTES OF CORE NOT USED
```

```
        COMPILER OPTIONS - NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,
                           SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF
  ISN 0002            SUBROUTINE EIGEN (A,B,NSUB,VALU,MSUB,ANORM,NMAX)
                C
                C     EIGENVALUES AND EIGENVECTORS OF A REAL SYMMETRIC MATRIX
                C
                C**** REMOVE NEXT STATEMENT IN SINGLE PRECISION VERSION
  ISN 0003            IMPLICIT REAL*8 (A-H,O-Z)
  ISN 0004            INTEGER*2 INDEX
  ISN 0005            DIMENSION A(NMAX,NSUB), B(NMAX,NSUB), VALU(MSUB)
  ISN 0006            DIMENSION    DIAG(100), SUPERD(100), WVEC(100), PVEC(100),
                     1             QVEC(100), VALL(100), Q(100), U(100),
                     2             INDEX(100), FACTOR(100), V(100), T(100,3)
  ISN 0007            COMMON /F202/ DIAG, SUPERD, WVEC, PVEC, T
  ISN 0008            EQUIVALENCE (WVEC(1),VALL(1),FACTOR(1),U(1)),
                     1            (PVEC(1),QVEC(1),Q(1),V(1))
                C**** REPLACE DATA VALUE WITH Z3C200000 IN SINGLE PRECISION VERSION
  ISN 0009            DATA E1/Z3420000000000000/
                C
                C     HOUSEHOLDER SIMILARITY TRANSFORMATION TO CO-DIAGONAL FORM
                C     ----------- ---------- -------------- -- ----------- ----
  ISN 0010            N=NSUB
  ISN 0011            M=MSUB
  ISN 0012            IF (N.LT.0) GO TO 220
  ISN 0014            IF (M.LE.0.OR.N.LE.2) GO TO 50
                C
                C     GENERATE IDENTITY MATRIX
                C
  ISN 0016            DO 40 I=2,N
  ISN 0017            DO 40 J=2,N
  ISN 0018            B(J,I)=0.0
  ISN 0019            IF (I.EQ.J) B(J,I)=1.0
  ISN 0021         40 CONTINUE
                C
  ISN 0022         50 DO 200 I=1,N
                C
                C     REDUCE COLUMN OF MATRIX
                C
  ISN 0023            I1=I+1
  ISN 0024            I2=I1+1
  ISN 0025            IF (I2.GT.N) GO TO 160
  ISN 0027            SUM=0.0
  ISN 0028            DO 70 J=I2,N
  ISN 0029         70 SUM=SUM+A(J,I)**2
  ISN 0030            IF (SUM.EQ.0.0) GO TO 160
  ISN 0032            J=I1
  ISN 0033            TEMP=A(J,I)
                C**** REPLACE FUNCTION NAMES IN NEXT FOUR STATEMENTS IN SINGLE PRECISION
  ISN 0034            SUM=DSQRT(SUM+ TEMP  **2)
  ISN 0035            A(J,I )=-DSIGN(SUM, TEMP  )
  ISN 0036            WVEC(J )=DSQRT(      1.0+DABS( TEMP   )/SUM)
  ISN 0037            DIV=DSIGN(     WVEC(J )*SUM, TEMP   )
  ISN 0038            DO 85 J=I2,N
  ISN 0039         85 WVEC(J)=A(J,I)/DIV
  ISN 0040            SCALAR=0.0
  ISN 0041            DO 95 J=I1,N
  ISN 0042            PVEC(J)=0.0
  ISN 0043            DO 90 K=I1,N
```

```
ISN 0044          90 PVEC(J)=PVEC(J)+A(K,J)*WVEC(K)
ISN 0045             SCALAR=SCALAR+PVEC(J)*WVEC(J)
ISN 0046          95 CONTINUE
ISN 0047             SCALAR=SCALAR/2.0
ISN 0048             DO 120 J=I1,N
ISN 0049             QVEC(J)=PVEC(J)-SCALAR*WVEC(J)
ISN 0050             DO 120 K=I1,J
ISN 0051             A(K,J)=A(K,J)-(WVEC(K)*QVEC(J)+WVEC(J)*QVEC(K))
ISN 0052             A(J,K)=A(K,J)
ISN 0053         120 CONTINUE
ISN 0054             IF (M.LE.0) GO TO 160
               C
               C     SAVE ROTATION FOR LATER APPLICATION TO CO-DIAGONAL VECTORS
               C
ISN 0056             DO 150 K=2,N
ISN 0057             TEMP=0.0
ISN 0058             DO 140 J=I1,N
ISN 0059         140 TEMP=TEMP+WVEC(J)*B(J,K)
ISN 0060             DO 150 J=I1,N
ISN 0061             B(J,K)=B(J,K)-WVEC(J)*TEMP
ISN 0062         150 CONTINUE
               C
               C     MOVE CO-DIAGONAL FORM ELEMENTS FOR ITERATIVE PROCEDURE
               C
ISN 0063         160 J=I
ISN 0064             DIAG(I)=A(J,I)
ISN 0065             IF (I.NE.N) SUPERD(I)=A(J+1,I)
ISN 0067         200 CONTINUE
               C
               C     GIVENS EIGENVALUE ITERATION FROM STURM CHAIN OF CO-DIAGONAL MINORS
               C     ------ ---------- --------- ---- ----- ----- -- ----------- ------
ISN 0068         220 N=IABS (N)
ISN 0069             M=IABS (M)
               C
               C     CALCULATE NORM OF MATRIX AND INITIALIZE EIGENVALUE BOUNDS
               C
ISN 0070             ANORM2=DIAG(1)**2
ISN 0071             IF (N.EQ.1) GO TO 235
ISN 0073             DO 230 L=2,N
ISN 0074             Q(L-1)=SUPERD(L-1)**2
ISN 0075             ANORM2=DIAG(L)**2+Q(L-1)+Q(L-1)+ANORM2
ISN 0076         230 CONTINUE
               C**** REPLACE FUNCTION NAME  IN NEXT      STATEMENT  IN SINGLE PRECISION
ISN 0077         235 ANORM=DSQRT(ANORM2)
ISN 0078             IF (M.EQ.0) RETURN
ISN 0080             DO 240 L=1,M
ISN 0081             VALU(L)=ANORM
ISN 0082             VALL(L)=-ANORM
ISN 0083         240 CONTINUE
ISN 0084             EPS1=ANORM*E1
ISN 0085             IF (EPS1.EQ.0.0) RETURN
               C
ISN 0087         250 DO 570 L=1,M
               C
               C     CHOOSE NEW TRIAL VALUE WHILE TESTING BOUNDS FOR CONVERGENCE
               C
ISN 0088         260 TAU=(VALU(L)+VALL(L))/2.0
ISN 0089             IF (     (TAU-VALL(L)).LE.E1*(DABS(TAU)+E1)) GO TO 570
```

```
               C
               C      DETERMINE SIGNS OF PRINCIPAL MINORS
               C
ISN 0091              MATCH=0
ISN 0092              T2=0.0
ISN 0093              T1=1.0
ISN 0094              DO 450 L1=1,N
ISN 0095              P=DIAG(L1)-TAU
ISN 0096              IF (T2.NE.0.0) GO TO 330
               C****  REPLACE FUNCTION NAME  IN NEXT      STATEMENT  IN SINGLE PRECISION
ISN 0098          300 T1=DSIGN(1.0D0,T1)
ISN 0099          330 IF (T1.NE.0.0) GO TO 400
               C****  REPLACE FUNCTION NAME  IN NEXT      STATEMENT  IN SINGLE PRECISION
ISN 0101              T0=-DSIGN(1.0D0,T2)
ISN 0102              T2=0.0
ISN 0103              IF (Q(L1-1).NE.0.0) GO TO 410
ISN 0105              GO TO 300
ISN 0106          400 T0=P-Q(L1-1)*T2/T1
ISN 0107              T2=1.0
               C
               C      COUNT AGREEMENTS IN SIGN (ZERO CONSIDERED POSITIVE)
               C
ISN 0108          410 IF (T0) 440, 420, 430
ISN 0109          420 T2=T1
ISN 0110              IF (T2) 440, 430, 430
ISN 0111          430 MATCH=MATCH+1
ISN 0112          440 T1=T0
ISN 0113          450 CONTINUE
               C
               C      ESTABLISH TIGHTER BOUNDS ON EIGENVALUES
               C
ISN 0114              DO 530 L1=L,M
ISN 0115              IF (L1.LE.MATCH) GO TO 500
ISN 0117              IF (VALU(L1).LE.TAU) GO TO 260
ISN 0119              VALU(L1)=TAU
ISN 0120              GO TO 530
ISN 0121          500 VALL(L1)=TAU
ISN 0122          530 CONTINUE
ISN 0123              GO TO 260
               C
ISN 0124          570 CONTINUE
               C
               C      EIGENVECTORS OF CO-DIAGONAL SYMMETRIC MATRIX--INVERSE ITERATION
               C      ------------ -- ----------- --------- ------  ------- ---------
ISN 0125              M=MSUB
ISN 0126              IF (M.LT.0) RETURN
               C
ISN 0128              DO 970 I=1,M
               C
               C      CHECK FOR REPEATED VALUE
               C
ISN 0129              IF (I.EQ.1) GO TO 725
ISN 0131              IF (VALU(I-1)-VALU(I).LT.1.0E+4*EPS1) GO TO 730
ISN 0133          725 I1=-1
ISN 0134          730 I1=I1+1
               C
               C      TRIANGULARIZE CO-DIAGONAL FORM AFTER EIGENVALUE SUBTRACTION
               C
```

```
ISN 0135          DO 760 L=1,N
ISN 0136          V(L)=EPS1*(I1*L+1)
ISN 0137          T(L,2)=DIAG(L)-VALU(I)
ISN 0138          IF (L.LT.N) GO TO 740
ISN 0140          T(L,3)=0.0
ISN 0141          GO TO 760
ISN 0142      740 T(L,3)=SUPERD(L)
ISN 0143          IF (T(L,3).EQ.0.0) T(L,3)=EPS1
ISN 0145          T(L+1,1)=T(L,3)
ISN 0146      760 CONTINUE
ISN 0147          DO 820 J=1,N
ISN 0148          T(J,1)=T(J,2)
ISN 0149          T(J,2)=T(J,3)
ISN 0150          T(J,3)=0.0
ISN 0151          GO TO 780
ISN 0152      770 T(J,1)=EPS1
                C**** REPLACE FUNCTION NAME  IN NEXT      STATEMENT  IN SINGLE PRECISION
ISN 0153      780 VTEMP=DABS(T(J,1))
ISN 0154          IF (VTEMP.LT.EPS1) GO TO 770
ISN 0156          IF (J.EQ.N) GO TO 820
ISN 0158          INDEX(J)=0
                C**** REPLACE FUNCTION NAME  IN NEXT      STATEMENT  IN SINGLE PRECISION
ISN 0159          IF (DABS(T(J+1,1)).LE.VTEMP) GO TO 810
ISN 0161          INDEX(J)=1
ISN 0162          DO 800 K=1,3
ISN 0163          VTEMP=T(J,K)
ISN 0164          T(J,K)=T(J+1,K)
ISN 0165          T(J+1,K)=VTEMP
ISN 0166      800 CONTINUE
ISN 0167      810 VTEMP   =T(J+1,1)/T(J,1)
ISN 0168          FACTOR(J)=VTEMP
ISN 0169          T(J+1,2)=T(J+1,2)-  VTEMP  *T(J,2)
ISN 0170          T(J+1,3)=T(J+1,3)-  VTEMP  *T(J,3)
ISN 0171      820 CONTINUE
ISN 0172          ITER=1
ISN 0173          IF (I1.GT.0) GO TO 920
                C
                C         BACK SUBSTITUTE TO OBTAIN EIGENVECTOR
                C
ISN 0175      860 DO 870 L1=1,N
ISN 0176          L=N+1-L1
ISN 0177          V( L )=(V( L )-T(L,2)*V( L+1 )-T(L,3)*V( L+2 ))/T(L,1)
ISN 0178      870 CONTINUE
ISN 0179          GO TO (875,920), ITER
                C
                C         PERFORM SECOND ITERATION
                C
ISN 0180      875 ITER=2
ISN 0181          IF (N.EQ.1) GO TO 860
ISN 0183      880 DO 910 L=2,N
ISN 0184          IF (INDEX(L-1).EQ.0) GO TO 900
ISN 0186          VTEMP=V(L-1)
ISN 0187          V(L-1)=V(L)
ISN 0188          V(L)=VTEMP
ISN 0189      900 V(L)=V(L)-FACTOR(L-1)*V(L-1)
ISN 0190      910 CONTINUE
ISN 0191          GO TO 860
                C
```

```
                  C      ORTHOGONALIZE VECTOR TO OTHERS ASSOCIATED WITH REPEATED VALUE
                  C
    ISN 0192        920 IF (I1.EQ.0) GO TO 945
    ISN 0194            DO 940 L1=1,I1
    ISN 0195            K=I-L1
    ISN 0196            VTEMP=0.0
    ISN 0197            DO 930 J=1,N
    ISN 0198        930 VTEMP=VTEMP+A(J,K)*V(J)
    ISN 0199            DO 940 J=1,N
    ISN 0200        940 V(J)=V(J)-A(J,K)*VTEMP
    ISN 0201            GO TO (880,945), ITER
                  C
                  C      NORMALIZE VECTOR
                  C
    ISN 0202        945 VNORM2=0.0
    ISN 0203            DO 950 L=1,N
    ISN 0204        950 VNORM2=VNORM2+V( L )**2
                  C**** REPLACE FUNCTION NAME  IN NEXT      STATEMENT  IN SINGLE PRECISION
    ISN 0205            VNORM=DSQRT(VNORM2)
    ISN 0206            DO 960 J=1,N
    ISN 0207        960 A(J,I)=V( J )/VNORM
                  C
    ISN 0208        970 CONTINUE
                  C
                  C      ROTATION OF CO-DIAGONAL VECTORS INTO MATRIX EIGENVECTORS
                  C      -------- -- ----------- ------- ---- ------ ------------
    ISN 0209            N=NSUB
    ISN 0210            IF (N.LE.2) RETURN
    ISN 0212            DO 990 I=1,M
    ISN 0213            DO 980 K=2,N
    ISN 0214            U(K)=0.0
    ISN 0215            DO 980 J=2,N
    ISN 0216        980 U(K)=U(K)+B(J,K)*A(J,I)
    ISN 0217            DO 990 J=2,N
    ISN 0218        990 A(J,I)=U(J)
                  C
    ISN 0219            RETURN
    ISN 0220            END

 *OPTIONS IN EFFECT*      NAME=  MAIN,OPT=02,LINECNT=60,SIZE=0000K,

 *OPTIONS IN EFFECT*      SOURCE,EBCDIC,NOLIST,NODECK,LOAD,NOMAP,NOEDIT,ID,NOXREF

 *STATISTICS*     SOURCE STATEMENTS =     219 ,PROGRAM SIZE =     4416

 *STATISTICS*  NO  DIAGNOSTICS GENERATED

 ****** END OF COMPILATION ******                                       85K BYTES OF CORE NOT USED

 *STATISTICS*    1 DIAGNOSTICS THIS STEP, HIGHEST SEVERITY CODE IS  4
```

VITA

Chanda Ashok Sapre

Candidate for the Degree of

Master of Science

Thesis: THE CALCULATION OF ELECTRIC DIPOLE OSCILLATOR STRENGTH FOR POLYATOMIC MOLECULES USING FLOATING SPHERICAL GAUSSIAN ORBITALS

Major Field: Physics

Biographical:

Personal Data: Born in Khandwa, India, April 27, 1945, the daughter of Mr. and Mrs. J. P. Maholay.

Education: Graduated from the Institute of Science of the University of Bombay in 1967 with a Bachelor of Science degree with physics major and mathematics minor and in 1969 with a Master of Science degree in physics. Completed the requirements for the Master of Science degree at Oklahoma State University, May, 1973.

Professional Experience: Graduate Teaching Assistant, Department of Physics, Oklahoma State University, 1971-1973.