A THEORETICAL ANALYSIS OF
EXCITATION AND CHARGE EXCHANGE IN ION-ATOM COLLISIONS

by

TEH-HSIN KUNG WU

A dissertation submitted to the faculty of the University of North Carolina in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Physics and Astronomy.

Chapel Hill

1981

[Signatures]
TEH-HSIN KUNG WU. A theoretical analysis of excitation and charge exchange in ion-atom collisions. (Under the direction of Eugen Merzbacher.)

ABSTRACT

The impact parameter dependence of the excitation and charge transfer process for the simple model of a collision between a bare nucleus and a one-electron nonrelativistic hydrogenic atom has been examined and calculated. Using semiclassical collision theory, Kepler motion on a hyperbolic trajectory is chosen to describe the relative nuclear motion. Applying a variational principle, coupled differential equations are derived for the amplitudes of the electronic states. Because of the Kepler motion, including recoil of the target, a general transformation theory connecting inertial and non-inertial frames of reference has been developed to obtain the appropriate Hamiltonian and wave functions. Fast Fourier transform and Romberg algorithms are applied to evaluate the matrix elements. The GEARS method has been used to solve the coupled differential equations. A preliminary calculation has been performed to evaluate the excitation and charge exchange probabilities.
Acknowledgements

I am indebted to the following people, without whose help, this work could not have been completed. Therefore, I would like to express my appreciation for their support: Dr. Merzbacher, for his enthusiasm and assistance in physics, guidance in the ordinary life, and many patient enjoyable Saturday morning discussions; Hai-ping Wu, for his complete understanding and support of my graduate work, and for sharing both schoolwork and housework with me; Dr. Thompson, for his assistance on computing techniques.

I also like to express my appreciation to the Durham office of the Computer Sciences Corporation for letting me use the word-processor to type this dissertation.
### Table of Contents

**Chapter**

1. Introduction 1

2. Time-Dependent Semiclassical Approach to Inelastic Ion-Atom Collisions 6
   - 2.1 Introduction 6
   - 2.2 Description of the Nuclear Motion 8
   - 2.3 Coupled Equations of Motion of the Electronic States 16

3. Transformation Theory 23
   - 3.1 Introduction 23
   - 3.2 Heisenberg Picture 25
   - 3.3 Schrödinger Picture 28
   - 3.4 Linear Accelerating Frames of Reference 35
   - 3.5 Wave Functions and Translation Factors 44
   - 3.6 Rotating Frames 48

4. Numerical Methods and Computational Results 55
   - 4.1 Determination of Matrix Elements 55
   - 4.2 Numerical Methods 60
   - 4.3 Computational Results and Numerical Errors 64

5. Conclusion 74

References 77

Appendix A: The Direct Coupling Matrix Elements 81

Appendix B: The Exchange Coupling Matrix Elements 88

Appendix C: Conservation of Probability 101

Appendix D: Program Listing 103
CHAPTER 1

INTRODUCTION

The theory of collisions of ions with atoms or other ions has been developed considerably in the last decade. Advances in computational technique have been largely responsible for this progress, but there has also been the influence of the challenge of improved experimental methods, notably in the area of ion sources and detection equipment. Furthermore, because of a great variety of accelerators, a large range of incident energy has been investigated and collisions between fully or partially stripped positive ions and atoms (or ions) have been achieved. The differential cross sections, which reveal more characteristics of the collision process than the total cross sections, have been measured as well as total cross sections.

During a general collision process between an incident ion (called 'projectile') and an atom or another ion (called 'target'), an active electron which is assumed to be attached to the target, and in the ground state, may be excited to a higher energy level of the target, or ionized as a free electron, or transferred to a bound state of the projectile. These three processes are called excitation, ionization, and charge exchange respectively.

There are three fundamental theories to characterize various inner-shell ion-atom collision processes:

(1) the plane wave Born approximation (MIF58). This theory uses plane waves to describe the motion of the electron as well as the target and the projectile. This is essentially first order quantum mechanical approach.

(2) the semiclassical approximation (BAN59). In this
approach the relative motion of the projectile and target is described classically (namely the projectile and target are moving along a trajectory), but the motion of the electron is described quantum mechanically by a wave function.

(3) the binary encounter approximation (GAR70)(GAR73). In this approximation, the interaction of the incident nucleus and the (inner-shell) active electron is a binary encounter between two free particles described classically. The potential of the target nucleus only has an influence on the initial and final electron momentum distributions.

All these three theoretical approaches have their limitations for certain applications. For instance, the plane wave Born approximation is adequate for most ion-atom collisions at high incident energy, but not good for the very close collisions. The semiclassical approximation is valid for close collisions at intermediate incident energy, in which case the orbital velocity of the active electron is compatible with the incident projectile velocity. The binary encounter approximation is remarkably successful in the low-velocity region if the projectile charge is much less than the charge of the target nucleus.

Where appropriate, atomic physicists have modified these theories to examine certain effects. For instance, Dirac wave functions have been used for inner-shell electrons of a heavy target nucleus to investigate the relativistic effect (CLA75)(AMU75)(DEC77). To correct for electron screening, a modified nuclear charge has been used in the electronic wave function (CHE68). Various trajectories have been assumed to examine the effect of Coulomb deflection in the relative motion of the projectile and target nuclei. Since we are interested in a close-collision process at intermediate incident energy region, and since the projectile and target nuclear charges are assumed to be comparable, we apply the semiclassical theory in this dissertation. We
evaluate the excitation and charge exchange probabilities. The latter is at the center of our interest and we shall discuss the relation of our work to parallel approaches which have been used.

In 1966, Wilets and Gallaher (WIL66) performed a coupled-state calculation of H⁺-H scattering in which the protons are described as moving on straight line trajectories and the (spinless) electronic wave function is expanded in terms of a finite subset of traveling hydrogenic wave functions centered about each proton. Five states are included in their calculation. Later, similar work was done with Sturmian states (GAL68), and a pseudo-state expansion using up to 7 states (CHE70).

More recently, Salop and Olsen evaluated the charge transfer and impact-ionization cross section by a (purely classical) three-dimensional Monte Carlo approach (OLS77a) and a coupled-state impact-parameter approximation (SAL77) (OLS77b). Adiabatic molecular wave functions were used for the second method. In addition to these representative papers, the theory of inelastic ion-atom collisions has been the subject of many other investigations. (MOR78)(LIN78) (LIN80)(MEY79)(STØ73)

In this dissertation, we will concentrate on the impact-parameter dependence of the excitation and charge transfer process for the simple model of a collision between a bare charged nucleus and a one-electron nonrelativistic hydrogenic atom. We assume a hyperbolic Kepler trajectory for the relative nuclear motion. The Kepler orbit is more complicated and more realistic than the various trajectories such as a straight line, or two broken straight lines forming by a scattering angle to simulate the Coulomb deflection effect (AND76). We solve the time-dependent Schrödinger equation by expanding the electronic wave functions in a subset of hydrogenic traveling wave functions. Because a hyperbolic trajectory is used for the relative nuclear motion, the translation factors used in these traveling wave functions
do not depend on time linearly. This distinguishes our work from that of Wilets and Gallaher and all others who used straight-line trajectories and nuclear motion at a constant velocity, making the matrix elements easy to evaluate. Because of the hyperbolic trajectory and the Kepler motion, including recoil of the target, we consider the appropriate changes in the electronic Hamiltonian which give the potential, rotational, and angular coupling terms. For the three-body system (the projectile nucleus, the target nucleus, and the electron), we use the full Hamiltonian with no approximation involved. Our goal is to investigate the Coulomb deflection ('curved trajectory' (WU76) and recoiling target) effect on the excitation and charge exchange probabilities. The only approximation in this calculation is the limitation to a small number of basis states and the resulting truncation of the coupled equations.

In Chapter 2, we first introduce the necessary formalism for the nuclear motion. Then using a variational principle, we establish the coupled differential equations for the electronic states. In Chapter 3, we develop a general transformation theory (WU79a) which had previously caused some confusion in the literature. We first start with the Heisenberg picture in an inertial frame of reference and derive the correct formalism in the Schrödinger picture in a non-inertial frame of reference. The Hamiltonian relation and the translation factor between these inertial and non-inertial frames of reference are also derived. In Chapter 4, we evaluate direct and exchange transition matrix elements (WU79b) and solve the coupled differential equations numerically. Chapter 5 contains some conclusions.

In the dissertation, we are using atomic units. By that we mean the following:

(a) Mass is measured in units of the electron mass, so that $m_e = 1$.
(b) Length is measured in units of Bohr radii, so that $a_0 = 1$. 
(c) Charge is measured in units of the electron charge, so that $e = 1$.

(d) $\hbar = 1$

From these conventional definitions, we also observe that the velocity is measured in units of $2.18 \times 10^8$ cm/sec, energy is measured in units of 27.2 ev and time is measured in units of $2.29 \times 10^{-19}$ sec.
CHAPTER 2

TIME-DEPENDENT SEMICLASSICAL APPROACH TO INELASTIC ION-ATOM COLLISIONS

2.1 Introduction

In the last decade, many different approaches have been developed to solve the inelastic ion-atom collision problem. Most of these fall into three categories: (1) fully quantum mechanical treatment of both nuclear and active electron motions; (2) fully classical treatment of both nuclear and active electron motions; (3) classical treatment of the nuclear motion and quantum mechanical treatment of active electron motion. (In an independent-particle description, an active electron is an electron which is ionized, excited or transferred during the collision process). The third case is called impact-parameter or semiclassical atomic collision theory. In this dissertation, we will adopt the semiclassical approach.

Our approach starts with a system in which an ion called "projectile" impinges on an atom called "target", causing an atomic electron to undergo a transition from its initial state. We treat the projectile and target nucleus as classical particles. The interaction between these two nuclei is the Coulomb force. According to classical mechanics, the relative Kepler motion of the projectile and target nucleus with Coulomb interaction follows a hyperbolic trajectory. The atomic electron, which moves in the potential due to the projectile and target nucleus, is described quantum mechanically and its effect on the nuclear
motion is neglected. In this dissertation, we will not consider many-electron atoms in detail, but our methods can be readily generalized to many-electron atoms. Without committing an appreciable error, we also ignore the electron spin.

Atomic collision processes are divided into many different regimes, according to such variables as impact parameter and velocity. Our interest is directed in particular to the following conditions:

(1) Large scattering angle, i.e. small impact parameter. If we would consider small angle scattering, the hyperbolic trajectory is not appreciably different from a straight-line trajectory, which has been widely used in the semiclassical approximation. We want to investigate how a strongly curved hyperbolic trajectory will affect the whole process. Generally, we consider that scattering angles greater than 10 degrees are "large" angles.

(2) Intermediate velocity range, \(1/2 < v_i/v_{el} < 2\). As presented by Briggs (BRI 78), the approximations made in our work are most appropriate for this velocity range. At higher incident velocity \(v_i\), a Born approximation treatment becomes appropriate instead of a close-coupling approach. Too low incident velocity will be a condition for near-adiabatic motion which favors a molecular treatment of the three-body system.

We will direct our effort toward the solution of the time-dependent, nonrelativistic spinless Schrödinger equation. Our main interest is to calculate the probabilities for excitation and charge exchange of the atomic electron. In section 2.2, we will discuss nuclear motion first. In section 2.3, we develop the fundamental formalism for the electronic system and derive the coupled equations of motion of the electron.
2.2 Description of the Nuclear Motion

The goal of this section is to provide the basic formalism and notation for the classical motion of the two nuclei in the context of the semiclassical theory that is being developed here. In particular, we want to derive relations which allow us to determine the time of arrival of the projectile at a point of the trajectory as a function of its location defined by its polar coordinates.

The interaction between the projectile ion A and target nucleus B is the repulsive Coulomb force. In Figure 2-1A, viewed in the laboratory system, the target nucleus B recoils from its initial position as a result of the scattering. In Figure 2-1B, viewed in the C. M. system, two particles A and B are initially moving toward each other; then after the scattering they are moving away from each other (GOL62). Both these frames of reference are inertial. In addition, we will use a non-inertial frame of reference in which the target nucleus B is stationary through the whole scattering process such that the two-body central force problem can be reduced to an effective one-body problem. From now on, we will call this special frame the target frame-of-reference.

Let the masses of projectile ion A and target B be $M_A$ and $M_B$, and

$$M = M_A + M_B$$

$$\mu = \frac{M_AM_B}{M}$$

where $\mu$ is the reduced mass. The charges of projectile A and target B are $Z_1$ and $Z_2$. The scattering between the projectile and the target is equivalent to a single particle of mass $\mu$ scattering from a fixed central potential ($Z_1Z_2/R$) where the internuclear distance $R$ is also the distance of the single particle from the center of force. The incident
velocity is the initial laboratory velocity $v_1$, and the
impact parameter is $b$. The orbit of the particle with mass
$\mu$ is shown in Figure 2-2 (KON69). The orbit is a branch of
a hyperbola, which has its focus at the center of target B
and which has the equation

$$R(\phi) = \frac{b^2}{-a+c \cos \phi}$$  \hspace{1cm} (2.1)

The parameters $a$ and $c$ are defined in the following
paragraphs.

Mathematically, "$a$" is the semimajor axis of the
hyperbola. ($b$, the impact parameter, is also the semiminor
axis). In collision theory, "$a$" is defined

$$a = \frac{Z_1 Z_2}{\mu v_1^2}$$  \hspace{1cm} (2.2.a)

which is half of the 'collision diameter'. The collision
diameter equals the distance of closest approach for a
'head-on' ($b=0$) collision.

"$c$" is the length

$$c = (a^2 + b^2)^{1/2}$$  \hspace{1cm} (2.2.b)

In terms of these lengths, the distance of closest approach
may be written

$$\gamma = a + c$$  \hspace{1cm} (2.2.c)

The angle $\phi$ is the polar angle referring to the line
joining the center of force to the turning point
("perihelion"). The orbit has asymptotes ($R \to \infty$) at
angle $\phi = \phi_L$ for which

$$\cos \phi_L = a/c$$  \hspace{1cm} (2.2.d)

and the angle $\phi$ has a range

$$-\phi_L < \phi < \phi_L$$
At $\phi=0$, $R(0)=b^2/(-a+c)=a+c=r$, the distance of closest approach.

As shown in Figure 2-2, the scattering angle $\theta$ is the angle between the incident and final directions of the scattered particle, and is

$$\theta = |\pi - 2\phi_L| < \pi \quad (2.2.e)$$

Now that we have discussed the geometric properties of the orbit, we proceed to compile some of the dynamic aspects of the problem.

Since the Coulomb interaction involves only the radial distance, the potential has spherical symmetry, and the total angular momentum in the C. M. frame

$$\mathbf{L} = \mu b v_1 \hat{y} = \ell \hat{y} \quad (2.3.a)$$

is conserved. $\hat{y}$ is the unit vector perpendicular to the plane of motion. $\ell$ is the constant magnitude of the angular momentum. In terms of radial distance and angular velocity, $\ell$ is

$$\ell = \mu R^2 \dot{\phi} \quad (2.3.b)$$
or

$$\ell = \mu R^2 \frac{d\phi}{dt}$$

Integrating the above equation and applying (2.1), we get

$$t = \frac{\mu}{\ell} \int \frac{b^4}{(-a+c\cos\phi)^2} \, d\phi + \text{const.}$$

To describe the collision, we choose the following initial conditions:
\[
\begin{align*}
  t = -\infty, & \quad R \to \infty, \quad \phi \to -\phi_L \\
  t = 0, & \quad R = \gamma, \quad \phi = 0 \\
  t = \infty, & \quad R \to \infty, \quad \phi \to \phi_L
\end{align*}
\] (2.4)

so that time \( t(\phi) \) as a function of the angle \( \phi \) is

\[
  t(\phi) = \int_0^\phi \frac{\nu b^4}{\lambda} d\phi = \int_0^\phi \frac{d\phi}{(c \cdot \cos \phi - a)^2}
\]

or

\[
  t(\phi) = \frac{1}{v_1} \left( \frac{R}{c \cdot \sin \phi + a \cdot \ln \left( \frac{b \cdot \tan \phi / 2 + c - a}{b \cdot \tan \phi / 2 + a - c} \right) } \right)
\] (2.5)

where we have used equation \( (2.3.a) \). This relation will be used frequently in the following.

Finally, we require an expression for the projectile velocity in terms of the position coordinates. Since Coulomb forces are conservative, the total energy of the system, \( E \), in the center-of-mass frame is a constant of the motion:

\[
  E = \frac{1}{2} \mu(R^2 + R^2 \dot{\phi}^2) + V_N(R)
\] (2.6)

where \( V_N(R) \) is the potential energy

\[
  V_N(R) = \frac{z_1 z_2}{R}
\] (2.7)

Applying \( (2.3.b) \) and solving \( (2.6) \) for \( \dot{R} \), we obtain

\[
  \dot{R} = \pm \left\{ \frac{2}{\mu} \left( E - V_N(R) - \frac{\lambda^2}{2 \mu R^2} \right) \right\}^{\frac{1}{2}}
\] (2.8)

where "+" is for \( t>0 \) and "-" is for \( t<0 \).
Let us choose Cartesian coordinates for the plane of motion: \((x,y,z)\) coordinates denote the center-of-mass frame, and \((x_B,y_B,z_B)\) (or \((x_A,y_A,z_A)\)) coordinates refer to the target frame (or projectile frame). The internuclear axis is along the \(z_A(z_B)\) axis, and the \(y(y_A\) and \(y_B)\) axis is out of the plane of motion as shown in Figure 2-3. In the target frame of reference, the internuclear distance vector is

\[
\dot{\hat{R}} = - R \hat{z}_B \quad (2.9.a)
\]

The velocity \(\dot{\hat{R}}\) is

\[
\dot{\hat{R}} = (-\dot{R})\hat{z}_B + (-\dot{R}\phi)\hat{x}_B = (-\dot{R})\hat{z}_B + (-bv_1/R)\hat{x}_B \quad (2.9.b)
\]

and the acceleration \(\ddot{\hat{R}}\), according to Newton's law,

\[
\ddot{\hat{R}} = - \frac{Z_1Z_2}{\mu R^2} \hat{z}_B \quad (2.9.c)
\]

The position vectors of the projectile and the target from the center of mass are

\[
\hat{R}^A = \frac{M_B}{M} \hat{R} = - \frac{M_B}{M} R \hat{z}_B \quad (2.10.a)
\]

\[
\hat{R}^B = - \frac{M_A}{M} \hat{R} = \frac{M_A}{M} R \hat{z}_B
\]

The accelerations are

\[
\ddot{\hat{R}}^A = - \frac{Z_1Z_2}{M_AR^2} \hat{z}_B \quad (2.10.b)
\]

\[
\ddot{\hat{R}}^B = \frac{Z_1Z_2}{M_BR^2} \hat{z}_B
\]
\[ \mathbf{\hat{R}}(t) = \mathbf{\hat{R}}^A(t) - \mathbf{\hat{R}}^B(t) \]

**Figure 2-3**
Note that all time derivatives of vectors are taken with respect to the center-of-mass frame, but their components are here expressed in terms of the target frame (i.e. vectors are projected in the target frame).

2.3 Coupled Equations of Motion of the Electronic States

Now we will discuss the quantum mechanical treatment of the electronic system when a projectile ion is scattered from a target atom. Our ultimate purpose is to calculate the time-dependent state vector $|\psi(t)\rangle$ of the active atomic electron. The electronic motion is considered in the time-dependent field determined by the Kepler motion of the projectile ion relative to the target nucleus. In the center-of-mass frame of reference, the time-dependent electronic Schrödinger equation of motion is

$$H_0 |\psi(t)\rangle = i \frac{\partial}{\partial t} |\psi(t)\rangle \quad (2.11)$$

where $H_0$ is the time-dependent Schrödinger Hamiltonian of the electronic system.

The Hamiltonian is

$$H_0 = \frac{1}{2} \mathbf{p}^2 - \frac{Z_1}{|\mathbf{r} - \mathbf{r}_A(t)|} - \frac{Z_2}{|\mathbf{r} - \mathbf{r}_B(t)|} \quad (2.12)$$

where the $\mathbf{r}_A(t)$ and $\mathbf{r}_B(t)$ are the position vectors of the projectile A and target B. The Hamiltonian $H_0$ is time dependent through these two vectors. Here we ignore the atomic electron-electron or other electron-nucleus interactions. We only consider the interactions between the projectile, the target and the active atomic electron.

Equation (2.11) is subject to well-defined boundary
and initial conditions. Since all calculations of transition probabilities, cross sections, and scattering amplitudes must necessarily be approximate, it is desirable to formulate the quantum mechanical problem in a manner that allows maximum flexibility. We will apply a variational approach to this impact-parameter problem. (McD70) (SIL60) (STO73) (GEL69) (SHA74).

We define the action integral:

$$ S = \int_{t_1}^{t_2} L \, dt, $$

where $L$ is a functional of the independent bra and ket trial states $\langle \psi_1^t(t) |$ and $| \psi_2^t(t) \rangle$, by

$$ L = \langle \psi_1^t(t) | H_\Omega - i \frac{a}{\hbar t} | \psi_2^t(t) \rangle $$

Therefore, the action integral may be written in the form

$$ S(\psi_1^t, \psi_2^t) = \int_{t_1}^{t_2} \langle \psi_1^t(t) | H_\Omega - i \frac{a}{\hbar t} | \psi_2^t(t) \rangle dt $$

If $| \psi_1 \rangle$ and $| \psi_2 \rangle$ are solutions of equation (2.11), then

$$ S(\psi_1, \psi_2) = 0 $$

To first order in $| \delta \psi_1 \rangle = | \psi_1^t(t) \rangle - | \psi_1 \rangle$ and $| \delta \psi_2 \rangle = | \psi_2^t(t) \rangle - | \psi_2 \rangle$, the variations of the action is

$$ \delta S = \int_{t_1}^{t_2} \langle \psi_1 | H_\Omega - i \frac{a}{\hbar t} | \psi_2 \rangle dt + \int_{t_1}^{t_2} \langle \psi_1 | H_\Omega - i \frac{a}{\hbar t} | \delta \psi_2 \rangle dt $$

$$ + \int_{t_1}^{t_2} \langle \delta \psi_1 | H_\Omega - i \frac{a}{\hbar t} | \psi_2 \rangle dt $$
\[ \delta S = \int_{t_1}^{t_2} \langle \delta \psi_1 | H_0 - i \frac{\partial}{\partial t} | \psi_2 \rangle dt + \int_{t_1}^{t_2} \langle \psi_1 | H_0 + i \frac{\partial}{\partial t} | \delta \psi_2 \rangle dt \]

\[- i \langle \psi_1 | \delta \psi_2 \rangle \bigg|_{t_1}^{t_2} + \int_{t_1}^{t_2} \langle \delta \psi_1 | H_0 - i \frac{\partial}{\partial t} | \delta \psi_2 \rangle dt \]

Also, the variation of the action integral is equal to the value of the action \( S \) corresponds to the varied states:

\[ \delta S = S(\psi_1^T, \psi_2^T) \]

\[ = -i \langle \psi_1 | \delta \psi_2 \rangle \bigg|_{t_1}^{t_2} + \int_{t_1}^{t_2} dt \langle \delta \psi_1 | H_0 - i \frac{\partial}{\partial t} | \delta \psi_2 \rangle \]

Showing that, to the first order, the value of \( \delta S \) depends only on the endpoints and not on the behavior of the states between \( t_1 \) and \( t_2 \).

Since the equation of motion is of first order in time, the variational principle cannot be formulated in terms of stationary values for the action. To do so would imply that the variation \( |\delta \psi_2\rangle \) vanishes at both endpoints, but this is generally inconsistent with the equation of motion. (ST073)

In applying the variational principle to the determination of an optimum trial function, which approximates the actual solution closely, we note that to first order

\[ \delta S = \int_{t_1}^{t_2} \langle \delta \psi_1 | H_0 - i \frac{\partial}{\partial t} | \psi_2^T(t) \rangle dt + \int_{t_1}^{t_2} \langle \psi_1^T(t) | H_0 + i \frac{\partial}{\partial t} | \delta \psi_2 \rangle dt \]

\[- i \langle \psi_1 | \delta \psi_2 \rangle \bigg|_{t_1}^{t_2} + \int_{t_1}^{t_2} \langle \delta \psi_1 | H_0 - i \frac{\partial}{\partial t} | \delta \psi_2 \rangle dt \]

The requirement that
\[
\frac{\delta S + i \langle \psi_1 | \delta \psi_2 \rangle}{t_1} \bigg|_{t_2} = 0
\]

to first order is therefore equivalent to the condition

\[
\int_{t_1}^{t_2} \langle \delta \psi_1 | H_0 - i \frac{\partial}{\partial t} | \psi_2^t(t) \rangle \ dt = 0 \quad (2.13.a)
\]

\[
\int_{t_1}^{t_2} \langle \psi_1^t(t) | H_0 + i \frac{\partial}{\partial t} | \delta \psi_2 \rangle \ dt = 0 \quad (2.13.b)
\]

The action principle is applied by substituting trial state vectors which are superpositions of a finite number of some simple physical states with variable coefficients into the variational conditions (2.13) to derive the coupled equations. We will expand \( |\psi_1^t(t)\rangle \) and \( |\psi_2^t(t)\rangle \) in terms of the same truncated set of basis vectors.

Trial state vectors can be constructed in many different ways (BRI78). Since we consider charge transfer processes as well as excitation explicitly, we shall include basis set functions that can represent adequately the bound states of the electron around the projectile. The trial state vectors are assumed to be superpositions of the basis sets on two centers, the target nucleus \( Z_2 \) and the projectile ion \( Z_1 \):

\[
|\psi_1^t(t)\rangle = \sum_n a_n^t(t) |\phi_n(A, t)\rangle + \sum_n b_n^t(t) |\phi_n(B, t)\rangle
\]

\[
|\psi_2^t(t)\rangle = \sum_n a_n^t(t) |\phi_n(A, t)\rangle + \sum_n b_n(t) |\phi_n(B, t)\rangle
\]

where
\[ |\Phi_n(A,t)\rangle = \text{the normalized bound state of an electron around the projectile} \]
\[ |\Phi_n(B,t)\rangle = \text{the normalized bound state of an electron around the target nucleus} \]

The variation of the trial function \( |\psi_1^\pm(t)\rangle \) is

\[
|\delta \psi_+\rangle = \sum_n \delta a_n(t)|\Phi_n(A,t)\rangle + \sum_n \delta b_n(t)|\Phi_n(B,t)\rangle
\]

Substituting the above equations into (2.13.a), we get

\[
\int dt \sum_m \sum_n (\delta a_m^{\dagger}(t)a_n(t)<\phi_m(A,t)|H_0-i\frac{\partial}{\partial t}|\Phi_n(A,t)\rangle + b_n(t)<\phi_m(A,t)|H_0-i\frac{\partial}{\partial t}|\Phi_n(B,t)\rangle)
\]

\[
+ \delta b_m^{\dagger}(t)(a_n(t)<\phi_m(B,t)|H_0-i\frac{\partial}{\partial t}|\Phi_n(A,t)\rangle + b_n(t)<\phi_m(B,t)|H_0-i\frac{\partial}{\partial t}|\Phi_n(B,t)\rangle)
\]

\[-i\delta a_m^{\dagger}(t)(\dot{a}_n(t)<\phi_m(A,t)|\Phi_n(A,t)\rangle + \delta n(t)<\phi_m(A,t)|\Phi_n(B,t)\rangle)
\]

\[-i\delta b_m^{\dagger}(t)(\dot{a}_n(t)<\phi_m(B,t)|\Phi_n(A,t)\rangle + \delta n(t)<\phi_m(B,t)|\Phi_n(B,t)\rangle)\]

\[= 0\]

Since the atomic basis on the same center are orthonormal sets, we have

\[<\phi_m(A,t)|\phi_n(A,t)\rangle = \delta_{mn} \quad (2.16)\]

\[<\phi_m(B,t)|\phi_n(B,t)\rangle = \delta_{mn}\]

If we treat all \( a_m(t), a_m^{\dagger}(t), b_m(t), b_m^{\dagger}(t) \) as independent variables, then (2.15) requires that all the coefficients of \( \delta a_m^{\dagger}(t) \) and \( \delta b_m^{\dagger}(t) \) be zero. Applying (2.16), we obtain
the coupled equations

\[ \sum_{n} a_n(t) H_{mn}(A) + \sum_{n} b_n(t) K_{mn}(A,B) = i \frac{\partial \hat{a}_n(t)}{\partial t} + i \sum_{n} \hat{b}_n(t) S_{mn} \]

\[ \sum_{n} a_n(t) \bar{H}_{mn}(B,A) + \sum_{n} b_n(t) \bar{H}_{mn}(B) = i \frac{\partial \hat{a}_n(t)}{\partial t} S_{mn}^\dagger + i \hat{b}_n(t) \]

where the matrix elements are defined

\[ H_{mn}(A) = \langle \phi_m(A,t) | H_0 - i \frac{\partial}{\partial t} | \phi_n(A,t) \rangle \]

\[ \bar{H}_{mn}(B) = \langle \phi_m(B,t) | H_0 - i \frac{\partial}{\partial t} | \phi_n(B,t) \rangle \]

\[ K_{mn}(A,B) = \langle \phi_m(A,t) | H_0 - i \frac{\partial}{\partial t} | \phi_n(B,t) \rangle \]

\[ \bar{K}_{mn}(B,A) = \langle \phi_m(B,t) | H_0 - i \frac{\partial}{\partial t} | \phi_n(A,t) \rangle \]

\[ S_{mn} = \langle \phi_m(A,t) | \phi_n(B,t) \rangle \]

\[ S_{mn}^\dagger = \langle \phi_m(B,t) | \phi_n(A,t) \rangle \]

If we vary the trial state \(|\psi_2(t)\rangle\) and substitute it into (2.13.b), we will get coupled equations which are just the complex conjugates of equation (2.17). So we consider that equations (2.17) and the matrix elements (2.18) are the coupled equations of motion, which approximate equation (2.11) for the active electron in the center-of-mass frame of reference.

In order to solve the coupled equations, the matrix elements shall be evaluated first. The matrices \(H_{mn}(A)\) (and \(\bar{H}_{mn}(B)\)), \(K_{mn}(A,B)\) (and \(\bar{K}_{mn}(B,A)\)), and \(S_{mn}\) in (2.18) are so-called 'direct coupling', 'exchange coupling', and
'overlap' matrices. In the next chapter, we will discuss a general transformation theory which will make these matrix elements easy to evaluate. In Chapter 4, the coupled equations will then be solved subject to the physically appropriate initial conditions:

\[
\begin{align*}
    b_n(-\infty) &= \begin{cases} 
        1 & \text{for } n=1 \\
        0 & \text{for } n=2,3,4,\ldots
    \end{cases} \\
    a_n(-\infty) &= 0 & \text{for } n=1,2,3,4,\ldots
\end{align*}
\]

(2.19)

so that the target atom is in the ground state before the collision starts.
CHAPTER 3

TRANSFORMATION THEORY

3.1 Introduction

In a semiclassical theory of the collision between an incident ion A and a target atom B, the motion of the projectile A relative to the target B is described by a hyperbolic Kepler orbit. During this process, an electron originally attached to target atom B may be excited, ionized or transferred to projectile A. From the previous chapter, the active electron's equations of motion in the center of mass system of nuclei A and B are derived as coupled differential equations. In order to solve the coupled differential equations, the coefficients need to be evaluated first. In the center-of-mass frame of reference, the Hamiltonian of the electron is straightforward. However, the wave functions of the electron are complicated and not easy to obtain. We can avoid this difficulty by using an accelerated frame of reference centered at either A or B, where the much less complicated atomic wave functions can be used to evaluate the matrix elements and to solve the equations of motion. Furthermore, the time-dependent interactions in the center of mass system

\[ V(r,t) = - \frac{Z_1}{|\hat{r} - \hat{r}_A(t)|} - \frac{Z_2}{|\hat{r} - \hat{r}_B(t)|} \]

can be written as
\[ V(r, t) = -\frac{Z_1}{|r_B - \vec{R}(t)|} - \frac{Z_2}{r_B} \]

in the accelerated frame of reference centered at B (see Figure 2-3). Then the only remaining time dependence of the interaction is simply the internuclear distance \( R(t) \) between A and B. Therefore, the potential energy is easy to handle, but on the other hand in these non-inertial frames of reference, which are centered at either A or B, the electronic Hamiltonian is not simply the total energy and cannot be derived by the usual simple correspondence from the Hamiltonian function of classical mechanics. A transformation theory developed particularly for this problem will provide the correct relations among operators, wave functions and Hamiltonians, transformed from an inertial frame of reference to a non-inertial one.

Because of the close correspondence between the classical theory and quantum dynamics, the Heisenberg picture approach, with operators changing in time and state vectors staying fixed, is better suited than the Schrödinger picture to formulate a quantum theory of the classical analogue of the desired coordinate transformations. After the correct formalism has been derived in the Heisenberg picture for transformation from an inertial frame of reference to a non-inertial one, the theory will be rewritten in the Schrödinger picture, which has a more intuitive form for scattering processes.

It is important to note that the transformation to a non-inertial frame is not the usual Galilean transformation. In contrast to a uniform velocity as in the Galilean case, the new frame of reference centered at either the projectile A or the target B is both linearly accelerating and rotating. However, the Galilean transformation is a special case of our general transformation theory. Section 3.2 and 3.3 will contain the general time-dependent unitary transformation theory, which is a generalization of the usual time-
independent quantum mechanical transformation theory. The application to the linear acceleration case and wave functions are examined in detail in section 3.4 and 3.5. The rotational transformation is discussed briefly in Section 3.6.

3.2 Heisenberg Picture

Consider a physical system which is viewed by observers in an inertial frame of reference \( F \) and in a non-inertial frame of reference \( F^\sigma \). These two frames of reference relate to each other through some time-dependent parameters in a general way. Since the nature of the physical system is characterized by state vectors and the state vectors in the Heisenberg picture are time-independent, we will discuss state vectors first.

Let \( |\psi\rangle \) and \( |\psi^\sigma\rangle \) be the Heisenberg state vectors of the physical system in \( F \) and \( F^\sigma \) respectively. We choose

\[
|\psi\rangle = |\psi^\sigma\rangle
\]

(3.1)

as our assumption.

We also choose a unitary operator for the transformation of observables from \( F \) to \( F^\sigma \) in the Heisenberg picture. This unitary operator, provides a canonical transformation, preserves hermiticity, and keeps eigenvalues invariant in the \( F^\sigma \) frame. Let \( U^\dagger_t(t) \) be this unitary operator which in general may be explicitly time-dependent. The subscript "\( t \)" denotes the time-dependent feature of the Heisenberg operators. The operator \( U^\dagger_t(t) \) satisfies

\[
U^\dagger_t(t)U_t(t) = U_t(t)U^\dagger_t(t) = 1
\]

(3.2)

and hence
\[
\frac{dU_{\mathcal{L}}(t)}{dt} U_{\mathcal{L}}^\dagger(t) + U_{\mathcal{L}}(t) \frac{dU_{\mathcal{L}}^\dagger(t)}{dt} = 0 \tag{3.3}
\]

\(A_t\) and \(A_t^\sigma\) are defined as Heisenberg operators of the physical system in \(\mathcal{F}\) and \(\mathcal{F}^\sigma\). With every Heisenberg operator \(A_t\) in \(\mathcal{F}\), we associate a Heisenberg operator \(A_t^\sigma\) in \(\mathcal{F}^\sigma\) according to the rule:

\[
A_t^\sigma = U_{\mathcal{L}}(t) A_t U_{\mathcal{L}}^\dagger(t) \tag{3.4}
\]

The equation of motion of operator \(A_t\) is

\[
i \frac{dA_t}{dt} = [A_t, H_t] + i \left( \frac{\partial A_t}{\partial t} \right) \tag{3.5}
\]

where \(H_t\) is the Hamiltonian of the physical system in \(\mathcal{F}\). Equation (3.4) tells us how \(A_t^\sigma\) develops in time. It will be useful to determine a new Hamiltonian \(\hat{H}_t^\sigma\), so that the equation of motion for \(A_t^\sigma\) becomes:

\[
i \frac{dA_t^\sigma}{dt} = [A_t^\sigma, \hat{H}_t^\sigma] + i \left( \frac{\partial A_t^\sigma}{\partial t} \right) \tag{3.6}
\]

The last equation is a conjecture, and we shall have to prove its validity and derive the form of \(\hat{H}_t^\sigma\).

According to quantum dynamics, the Hamiltonian \(H_t\) in the inertial frame of reference \(\mathcal{F}\) is an observable whose expectation value \(\langle H_t \rangle\) measures the total energy of the physical system. However, the Hamiltonian \(\hat{H}_t^\sigma\) in the non-inertial frame of reference \(\mathcal{F}^\sigma\) does generally not represent the total energy of the physical system. Generally, the Hamiltonians \(H_t\) and \(\hat{H}_t^\sigma\) do not transform merely as operators according to (3.4). In other words, generally, \(H_t^\sigma\) differs
from $\hat{H}_t$.

In order to find out how the Hamiltonian in one reference frame is related to the Hamiltonian in another reference frame, we use (3.4) to calculate the time derivative of $A_t^\sigma$:

$$\frac{dA_t^\sigma}{dt} = i \frac{d}{dt} (U_t(t)A_tU_t^\dagger(t))$$

$$= i \frac{dU_t(t)}{dt} A_t U_t^\dagger(t) + iU_t(t) \frac{dA_t}{dt} U_t^\dagger(t) + iU_t(t)A_t \frac{dU_t(t)}{dt}$$

From (3.3), (3.4), and (3.5), we obtain

$$i \frac{dA_t^\sigma}{dt}$$

$$= i \frac{dU_t(t)}{dt} U_t^\dagger(t)A_t^\sigma + iA_t^\sigma U_t(t) \frac{dU_t(t)}{dt} + U_t(t)[A_t^\sigma, H_t] U_t^\dagger(t) + iU_t(t) \frac{\partial A_t}{\partial t} U_t^\dagger(t)$$

$$= i[A_t^\sigma, U_t(t) \frac{dU_t(t)}{dt}] + iU_t(t) \frac{\partial A_t}{\partial t} U_t^\dagger(t)$$

$$= [A_t^\sigma, U_t(t) H_t U_t^\dagger(t) + iU_t(t) \frac{\partial A_t}{\partial t}] + i(\frac{\partial A_t^\sigma}{\partial t})$$

where we have used (3.4) to define

$$\frac{\partial A_t}{\partial t} = U_t(t) \frac{\partial A_t}{\partial t} U_t^\dagger(t)$$

We thus see that an equation of the form (3.6) is satisfied no matter which non-inertial frame of reference we are in,
if we define

$$\hat{H}^\sigma_t = U^\dagger_t(t)H_tU_t(t) + iU_t(t)\frac{d U^\dagger_t(t)}{dt} \quad (3.7.a)$$

Equation (3.7.a) which transforms Hamiltonian $H_t$ into $\hat{H}^\sigma_t$ is different from the operator transformation (3.4) by an extra term $iU_t(t)\frac{d U^\dagger_t(t)}{dt}$. The unitary operator $U_t(t)$ is expressed in terms of the observables in $F$ and thus (3.5) is also true for $U_t(t)$. We may now write (3.7.a) in the form

$$\hat{H}^\sigma_t = U^\dagger_t(t)H_tU_t(t) + U_t(t)([U^\dagger_t(t),H_t] + i\frac{\partial U^\dagger_t(t)}{\partial t})$$

$$= U^\dagger_t(t)H_tU_t(t) + H_t - U^\dagger_t(t)H_tU_t(t) + iU_t(t)\frac{\partial U^\dagger_t(t)}{\partial t}$$

Therefore,

$$\hat{H}^\sigma_t = H_t + i\frac{\partial U^\dagger_t(t)}{\partial t} \quad (3.7.b)$$

One simple consequence of equation (3.7.b) is that if $(iU_t(t)\frac{\partial U^\dagger_t(t)}{\partial t})$ does not explicitly depend on time, $\langle \hat{H}^\sigma_t \rangle$ and $\langle H_t \rangle$ differ only by a time-independent constant. The Galilean transformation is an example.

3.3 Schrödinger Picture

We have derived the relationship between the Heisenberg state vectors, operators and Hamiltonians for the frames of reference $F$ and $F^\sigma$ in the previous section. Now we are
going to establish the equivalent formulations in the Schrödinger picture. In going from the Heisenberg to the Schrödinger picture, one requires that probabilities and eigenvalues and thus expectation values remain unchanged.

In the Schrödinger picture, the state of the physical system in frame $F$ at time $t$ is determined by the state vector $|\Psi(t)\rangle$. From quantum mechanics, this state vector can be obtained by a unitary time development operator $T(t,t_0)$

$$|\Psi(t)\rangle = T(t,t_0)|\Psi(t_0)\rangle \quad (3.8)$$

as long as the state vector $|\Psi(t_0)\rangle$ is known at an earlier time $t_0$. Here we choose the Schrödinger state vector at $t_0$ to be the same as Heisenberg state vector $|\psi\rangle$

$$|\Psi(t_0)\rangle = |\psi\rangle \quad (3.9)$$

It follows from (3.8) and (3.9) that $T(t,t_0)$ connects the Heisenberg and the Schrödinger state vectors in the inertial frame of reference $F$

$$|\Psi(t)\rangle = T(t,t_0)|\psi\rangle \quad (3.10)$$

Similar equations can be written in the non-inertial frame of reference $\mathcal{F}^\sigma$

$$|\Psi^\sigma(t)\rangle = T^\sigma(t,t_0)|\psi^\sigma\rangle = T^\sigma(t,t_0)|\psi\rangle \quad (3.11)$$

From now on $T(t)$ and $T^\sigma(t)$ are used as terms of convenience to represent $T(t,t_0)$ and $T^\sigma(t,t_0)$.

Operators $A_0$ and $A_0^\sigma$ referred to are the Schrödinger forms of the Heisenberg picture operators $A_t$ and $A_t^\sigma$ in $F$ and $\mathcal{F}^\sigma$. They are also transformed by the time development
operators according to

\[ A_o = T(t)A_t T^\dagger(t) \]  \hspace{1cm} (3.12)

and

\[ A_o^\sigma = T^\sigma(t)A_t^\sigma T^\sigma\dagger(t) \]  \hspace{1cm} (3.13)

where the subscript 'o' denotes the time-independent feature of the Schrödinger picture.

Differentiating (3.12) and applying (3.5), we obtain

\[
\frac{dT(t)}{dt} - A_t T^\dagger(t) + T(t)\frac{dA_t}{dt} - T^\dagger(t) + T(t)A_t \frac{dT^\dagger(t)}{dt} = 0
\]

Hence, using (3.12) once more,

\[
[A_o, \frac{dT(t)}{dt} - T^\dagger(t) + iT(t)H_T T^\dagger(t)] = 0
\]

Since this is true for any operator \( A_o \), we get the equation of motion for the time development operator as follows:

\[
i \frac{dT(t)}{dt} = T(t)H_T = H_0 T(t) \]  \hspace{1cm} (3.14)

where \( H_0 = TH_T T^{-1} \). The same operations can be applied to
the operator $T^\sigma(t)$. The equation of motion for $T^\sigma(t)$ is thus derived as

$$i \frac{dT^\sigma(t)}{dt} = T^\sigma(t)\hat{H}^\sigma_t$$  \hspace{1cm} (3.15)

Now differentiating (3.10) with respect to $t$, we obtain

$$i \frac{d|\psi(t)>}{dt} = i \frac{dT(t)}{dt} |\psi>$$

Then using (3.14) and (3.10), we get

$$i \frac{d|\psi(t)>}{dt} = T(t)\hat{H}^\sigma_t|\psi> = T(t)\hat{H}^\sigma_tT^\dagger(t)|\psi(t)>$$

such that the Schrödinger equation of motion of state vector $|\psi(t)>$ is written as (MER70)

$$i \frac{d|\psi(t)>}{dt} = \hat{H}_0|\psi(t)>$$  \hspace{1cm} (3.16)

The same argument can be made in the non-inertial frame of reference $F^\sigma$. The Schrödinger equation of motion of state vector $|\psi^\sigma(t)>$ is

$$i \frac{d|\psi^\sigma(t)>}{dt} = \hat{H}^\sigma_0|\psi^\sigma(t)>$$  \hspace{1cm} (3.17)

where $\hat{H}^\sigma_0$, the Hamiltonian, is defined
\[ \hat{H}_O^\sigma = T^\sigma(t) \hat{H}_T^\sigma T^{\sigma\dagger}(t) \] (3.18)

Let us summarize the transformations we have established so far. Table 3-1 gives a schematic relation.

<table>
<thead>
<tr>
<th>frame of reference</th>
<th>Heisenberg</th>
<th>Schrödinger</th>
</tr>
</thead>
<tbody>
<tr>
<td>inertial F</td>
<td>(A_t, H_T \xrightarrow{T(t)} A_O, H_O)</td>
<td>(\uparrow)</td>
</tr>
<tr>
<td></td>
<td>(U_T(t))</td>
<td>(\uparrow)</td>
</tr>
<tr>
<td>non-inertial (F^\sigma)</td>
<td>(A_t^\sigma, \hat{H}_T^\sigma \xrightarrow{T^\sigma(t)} A_O^\sigma, \hat{H}_O^\sigma)</td>
<td>(\uparrow)</td>
</tr>
</tbody>
</table>

Table 3-1

The transformation between the two frames of reference in the Schrödinger picture is our main interest. If we know the relationship between \(H_O\) and \(\hat{H}_O^\sigma\), \(|\psi(t)\rangle\) and \(|\psi^\sigma(t)\rangle\), we are ready for the collision problem discussed in Section 3.1.

From (3.10) and (3.11), we may write

\[ |\psi^\sigma(t)\rangle = T^\sigma(t)|\psi\rangle = T^\sigma(t)T^{\sigma\dagger}(t)|\psi(t)\rangle \] (3.19.a)

A new operator \(S(t)\) can be defined as

\[ S(t) = T^\sigma(t)T^{\sigma\dagger}(t) \] (3.20)

such that it transforms the Schrödinger state vector \(|\psi(t)\rangle\) into \(|\psi^\sigma(t)\rangle\) according to

\[ |\psi^\sigma(t)\rangle = S(t)|\psi(t)\rangle \] (3.19.b)
Here $S(t)$ is also a unitary operator. This $S(t)$ operator connects the Schrödinger state vectors in $F$ and $F^\sigma$ frames of reference together. Differentiating both sides of (3.20) with respect to time, we obtain

$$i \frac{dS(t)}{dt} = i \frac{dT^\sigma(t)}{dt} T^\dagger(t) + i T^\sigma(t) \frac{dT^\dagger(t)}{dt}$$

Substituting the equations of motion of $T^\dagger(t)$ and $T^\sigma(t)$, (3.14) and (3.15), into the above equation, we get

$$i \frac{dS(t)}{dt} = T^\sigma(t) \hat{H}^\sigma_t T^\dagger(t) - T^\sigma(t) H_t T^\dagger(t)$$

$$= T^\sigma(t) (\hat{H}^\sigma_t - H_t) T^\dagger(t)$$

In Section 3.2, we have derived the relation of $H_t$ and $\hat{H}^\sigma_t$. So from (3.7.b), we write

$$i \frac{dS(t)}{dt} = i T^\sigma(t) \left( U_t(t) \frac{\partial U^\dagger_t(t)}{\partial t} \right) T^\dagger(t)$$

$$= i S(t) T(t) \left( U_t(t) \frac{\partial U^\dagger_t(t)}{\partial t} \right) T^\dagger(t)$$

which yields the equation of motion for the operator $S(t)$:

$$\frac{dS(t)}{dt} = S(t) \left( U_o(t) \frac{\partial U^\dagger_o(t)}{\partial t} \right)$$  \hspace{1cm} (3.21)

where we use (3.12) to define

$$U_o(t) = T(t) U_t(t) T^\dagger(t)$$
and
\[
\frac{\partial U_\sigma^\dagger(t)}{\partial t} = T(t) \frac{\partial U_\sigma^\dagger(t)}{\partial t} \tilde{T}^\dagger(t)
\]

It is now apparent that if \( U_\sigma(t) \), the unitary operator in the Schrödinger picture which corresponds to the unitary operator \( U_\tau(t) \) in the Heisenberg picture, is known, \( S(t) \) can be solved and the relation between \( |\psi(t)\rangle \) and \( |\psi_\sigma(t)\rangle \) can be derived. The operator \( S(t) \) also can be used to relate the Hamiltonians, \( H_\sigma \) and \( H_\sigma^\tau \) in the Schrödinger picture. Applying (3.7.b) and (3.12) to equation (3.18), we obtain

\[
\frac{\hat{H}_\sigma^\dagger}{\partial t} = T_\sigma(t) (H_\tau + i U_\tau(t) \frac{\partial U_\sigma^\dagger(t)}{\partial t}) T_\sigma^\dagger(t)
\]

\[
= T_\sigma(t)T_\sigma^\dagger(t)H_\sigma T(t)T_\sigma^\dagger(t) + i T_\sigma(t)U_\tau(t) \frac{\partial U_\sigma^\dagger(t)}{\partial t} T_\sigma^\dagger(t)
\]

\[
= S(t)H_\sigma S_\sigma^\dagger(t) + iS(t)T(t)(U_\tau(t) \frac{\partial U_\sigma^\dagger(t)}{\partial t})T(t)S_\sigma^\dagger(t)
\]

which may be written

\[
\frac{\hat{H}_\sigma}{\partial t} = S(t) (H_\sigma + i U_\sigma(t) \frac{\partial U_\sigma^\dagger(t)}{\partial t}) S_\sigma^\dagger(t) \tag{3.22}
\]

Equations (3.19.b), (3.21) and (3.22) complete our general time-dependent transformation theory. Using these equations, we can relate operators, Hamiltonians, and state vectors in \( F \) and \( F'^\sigma \) frames of reference, as long as \( U_\sigma(t) \) and \( U_\tau(t) \) are constructed. In the following sections, we will discuss the application of this transformation theory to accelerated frames of reference and show its suitability for this purpose.
3.4 Linear Accelerating Frames of Reference

In the previous sections, we have derived a general transformation theory which can transform a state in an inertial frame of reference $F$ to the corresponding state in a non-inertial frame of reference $F^\sigma$. Now we will apply this transformation theory to a specific non-inertial frame of reference $F^a$. We assume that this non-inertial frame $F^a$ moves without rotation with a time-dependent linear, but not necessarily constant, acceleration $\dot{\bar{a}}(t)$. The translational motion of $F^a$ is characterized by the displacement $\bar{d}(t)$ of the coordinate origin, as shown in Figure 3-1. (Subscript or superscript 'a' attached to the notations indicate that $F^a$ is a linearly accelerating frame of reference relative to an inertial frame of reference $F$.) We shall look into the dynamics of a single particle first. Our discussion will be quite explicit and detailed on purpose, because there is much confusion in the literature in these matters.

Classically, the coordinates, $\bar{r}=(x,y,z)$ and $\bar{r}^a=(x^a,y^a,z^a)$, and kinetic momenta, $\bar{p}=m\bar{r}=(p_x,p_y,p_z)$ and $\bar{p}^a=(p_x^a,p_y^a,p_z^a)$, of the single particle in $F$ and $F^a$ have the following relations:

$$\bar{r}^a = \bar{r} - \bar{d}(t) \quad (3.23)$$

$$\bar{p}^a = \bar{p} - m \bar{d}(t) \quad (3.24)$$

where $m$ is the mass of the particle and $\bar{d}(t)$ is the translation velocity of $F^a$ relative to $F$. Owing to the close correspondence between classical dynamics and quantum mechanics, similar equations can be written in the Heisenberg picture in terms of operators:
\[ \dot{x}_t^a = \dot{x}_t - \dot{d}(t) \]  
\[ \dot{p}_t^a = \dot{p}_t - m \dot{d}(t) \]  

(3.25.a)  
(3.26.a)

where \( \dot{x}_t^a \) and \( \dot{x}_t \) are the position operators and \( \dot{p}_t^a \) and \( \dot{p}_t \) are the momentum operators in the \( F^a \) and \( F \) frames of reference respectively. The transformation between the two frames of reference is effected by the unitary operator:

\[ U_{ta}(t) = \exp(i(m\dot{d}(t) \cdot \dot{x}_t - \dot{d}(t) \cdot \dot{p}_t)) \]  

(3.27)

which preserves the commutation relations. We shall prove the appropriateness of this choice by showing that the Heisenberg position and momentum operators are indeed transformed according to (3.4):

\[ \dot{x}_t^a = U_{ta}(t) \dot{x}_t U_{ta}^+(t) \]  
\[ \dot{p}_t^a = U_{ta}(t) \dot{p}_t U_{ta}^+(t) \]  

(3.25.b)  
(3.26.b)

To this purpose, we rewrite the unitary operator \( U_{ta}(t) \) in the following form:

\[ U_{ta}(t) = e^{im\dot{d}(t) \cdot \dot{x}_t} e^{-id(t) \cdot \dot{p}_t} e^{-im\dot{d}(t) \cdot \dot{d}(t)/2} \]

where we have used the identity (MER70)

\[ e^A e^B = e^{A+B+[A,B]/2} \]  

(3.28)

for operators \( A \) and \( B \) which commute with \( [A,B] \). Let \( U_{ta}(t) \) and its adjoint operate on \( \dot{x}_t \):
\[ U_{ta}(t) \dot{X}_t U_{ta}^\dagger(t) = e^{i\dot{d}(t)} \cdot \dot{X}_t e^{-i\dot{d}(t)} \cdot \dot{p}_t X_t e^{i\dot{d}(t)} \cdot \dot{p}_t e^{-i\dot{d}(t)} \cdot \dot{X}_t \]

\[ = e^{i\dot{d}(t)} \cdot \dot{X}_t (\dot{X}_t - \dot{d}(t)) e^{-i\dot{d}(t)} \cdot \dot{X}_t \]

\[ = \dot{X}_t - \dot{d}(t) = \dot{X}_t^a \]

Here we have applied another identity,

\[ e^{A} e^{-A} = B + [A,B] + \frac{1}{2!} [A,[A,B]] + \frac{1}{3!} [A,[A,[A,B]]] + \ldots \quad (3.29) \]

and the familiar commutation relations

\[ [X_t, P_{X_t}] = [Y_t, P_{Y_t}] = [Z_t, P_{Z_t}] = i \]

\[ [X_t, P_{Y_t}] = [X_t, P_{Z_t}] = [Y_t, P_{X_t}] = [Y_t, P_{Z_t}] \]

\[ = [Z_t, P_{X_t}] = [Z_t, P_{Y_t}] = 0 \]

Equation (3.26.b) can be similarly verified. Note that the unitary operator \( U_{ta}(t) \) is an operator with time-dependent parameters \( \dot{d}(t) \) and \( \dot{d}(t) \) which imply the linearly accelerated relative motion of \( F \) and \( F^a \). We can say that \( U_{ta}(t) \) is a "bridge" to connect these two frames of reference.

To obtain the corresponding unitary operator \( U_{oa}(t) \) in the Schrödinger picture, the time-development operator \( T(t) \) is introduced and applied to the unitary operator \( U_{ta}(t) \) to give

\[ U_{oa}(t) = T(t) U_{ta}(t) T^\dagger(t) \]
Therefore,

\[ U_{oa}(t) = \exp(i(m\dot{d}(t) \cdot \dot{X}_o - d(t) \cdot \dot{P}_o)) \]  \hspace{1cm} (3.30)

where we use the operator transformation relations

\[ \dot{X}_o = T(t) \dot{X}_t T(t) \]
\[ \dot{P}_o = T(t) \dot{P}_t T(t) \]  \hspace{1cm} (3.31)

according to equation (3.12). Of course, the position and momentum operators preserve their commutation relations in the Schrödinger picture.

In order to obtain an equation for the operator \( S(t) \), we now construct equation (3.21). We rewrite equation (3.30) as follows

\[ U_{oa}(t) = e^{im\dot{d}(t) \cdot \dot{X}_o} e^{-i\dot{d}(t) \cdot \dot{P}_o} e^{-im\dot{d}(t) \cdot \dot{d}(t)/2} \]  \hspace{1cm} (3.30.a)

and then take the adjoint,

\[ U_{oa}^\dagger(t) = e^{im\dot{d}(t) \cdot \dot{d}(t)/2} e^{i\dot{d}(t) \cdot \dot{P}_o} e^{-im\dot{d}(t) \cdot \dot{X}_o} \]  \hspace{1cm} (3.30.b)

where we utilize the identity (3.28). Differentiating \( U_{oa}^\dagger(t) \) with respect to time and multiplying the result from the left by \( U_{oa}(t) \) yields

\[
\frac{\partial U_{oa}(t)}{\partial t} = \frac{1}{2}m\ddot{d}(t) \cdot \ddot{d}(t) + \frac{1}{2}m(\dot{d}(t))^2 + e^{im\dot{d}(t) \cdot \dot{X}_o} (i\dot{d}(t) \cdot \dot{P}_o) e^{-im\dot{d}(t) \cdot \dot{X}_o} - im\dot{d}(t) \cdot \dot{X}_o
\]
From the identity (3.29) and the commutation relations between the Schrödinger position and momentum operators, we may write

$$U_{\alpha\alpha}(t) \frac{\partial U_{\alpha\alpha}^\dagger(t)}{\partial t} = \frac{1}{2} m \ddot{\mathbf{d}}(t) \cdot \ddot{\mathbf{d}}(t) - \frac{1}{2} m(\dot{\mathbf{d}}(t))^2 + i \dot{\mathbf{d}}(t) \cdot \mathbf{P}_0 - im \ddot{\mathbf{d}}(t) \cdot \mathbf{X}_0$$

(3.32)

Substituting (3.32) into (3.21) we get the differential equation for operator $S_{\alpha}(t)$

$$\frac{dS_{\alpha}(t)}{dt} = S_{\alpha}(t) \left( iG(T) + i \ddot{\mathbf{d}}(t) \cdot \mathbf{P}_0 - im \ddot{\mathbf{d}}(t) \cdot \mathbf{X}_0 \right)$$

(3.33)

where we define

$$G(t) = m \left( \ddot{\mathbf{d}}(t) \cdot \ddot{\mathbf{d}}(t) - (\dot{\mathbf{d}}(t))^2 \right)/2$$

(3.33) is the equation which we must solve to obtain the transformation between the two frames of reference.

In order to solve (3.33), we first integrate (3.33) and apply the initial conditions. Then we derive the solution of $S_{\alpha}(t)$ at time $t$

$$S_{\alpha}(t) = \exp( i \int_0^t dt' \left( iG(t') + i \ddot{\mathbf{d}}(t') \cdot \mathbf{P}_0 - im \ddot{\mathbf{d}}(t') \cdot \mathbf{X}_0 \right)$$

$$= \exp( iF'(t) + i(\ddot{\mathbf{d}}(t) - \ddot{\mathbf{d}}(0)) \cdot \mathbf{P}_0 - im(\dot{\mathbf{d}}(t) - \dot{\mathbf{d}}(0)) \cdot \mathbf{X}_0)$$

$$= e^{iF(t)} e^{i(\ddot{\mathbf{d}}(t) - \ddot{\mathbf{d}}(0)) \cdot \mathbf{P}_0} e^{-im(\dot{\mathbf{d}}(t) - \dot{\mathbf{d}}(0)) \cdot \mathbf{X}_0}$$

(3.34)

where $F(t)$ is defined as
\[
F(t) = \int_0^t dt' G(t') + m(\ddot{d}(t) - \ddot{d}(0)) \cdot (\dot{d}(t) - \dot{d}(0))/2
\]

and identity (3.28) is used again here.

We now seek to demonstrate how the operators, \( U_{Oa}(t) \) and \( S_a(t) \), transform the quantities in the Schrödinger picture from the \( F \) frame to the \( F^a \) frame of reference. To show this we consider Table 3-1 in a clockwise direction. First, we relate the Schrödinger position and momentum operators, \( \hat{X}_O^a \) and \( \hat{P}_O^a \), in \( F^a \) back to the corresponding Schrödinger operators in the \( F \) frame of reference. According to equations (3.13), (3.25a) and (3.12), we obtain

\[
\hat{X}_O^a = T^a(t) (\hat{X}_L - \ddot{d}(t)) T^{a\dag}(t)
\]

\[
= S_a(t) \hat{X}_O S_a^\dagger(t) - \ddot{d}(t)
\]

(3.35a)

for the position operator and

\[
\hat{P}_O^a = T^a(t) (\hat{P}_L - m\ddot{d}(t)) T^{a\dag}(t)
\]

\[
= S_a(t) \hat{P}_O S_a^\dagger(t) - m\ddot{d}(t)
\]

(3.36a)

for the momentum operator, where \( S_a(t) \) is defined as (3.20) and \( T^a(t) \) is used as a unitary operator. Then substituting (3.34) in the above equations, we may write

\[
\hat{X}_O^a = e^{i(\ddot{d}(t) - \ddot{d}(0)) \cdot \hat{P}_0} \hat{X}_O e^{-i(\ddot{d}(t) - \ddot{d}(0)) \cdot \hat{P}_0} - \ddot{d}(t)
\]

\[
= \hat{X}_O + i(\ddot{d}(t) - \ddot{d}(0)) \cdot [\hat{P}_O, \hat{X}_O] - \ddot{d}(t)
\]

\[
\hat{P}_O^a = \hat{X}_O - \ddot{d}(0)
\]

(3.35b)
and

\[ \hat{\mathbf{p}}_0^a = e^{-i \mathbf{d}(t) \cdot \mathbf{d}(0)} \cdot \hat{x}_0^a \mathbf{p}_0 \cdot e^{i \mathbf{d}(t) \cdot \mathbf{d}(0)} \cdot \hat{x}_0^a \cdot \hat{\mathbf{p}}_0 = \hat{\mathbf{p}}_0 - i \mathbf{d}(t) \cdot [\hat{x}_0^a, \hat{\mathbf{p}}_0] \cdot \hat{\mathbf{d}}(t) \]

\[ \hat{\mathbf{p}}_0^a = \hat{\mathbf{p}}_0 - \hat{\mathbf{m}} \hat{\mathbf{d}}(0) \]  

(3.36.b)

Here we again use the identity (3.29) and \([\hat{x}_0^a, \hat{\mathbf{p}}_0] = i\).

These relations look almost trivial but nevertheless deserve some comments.

As shown in equation (3.35.b) and (3.36.b), our general transformation theory leaves the Schrödinger position and momentum operators time-independent. \(\hat{x}_0^a\) and \(\hat{\mathbf{p}}_0^a\) are determined at time zero and are not changing as time proceeds, as is characteristics of Schrödinger operators. But \(\hat{x}_0^a\) and \(\hat{\mathbf{p}}_0^a\) are of course observables, and their expectation values are the values of position and momentum in the classical limit and changing with time. (Momentum is here defined as a product of mass with velocity).

Where does the time-dependence in the expectation values come from? In the Schrödinger picture, the time-dependence is embedded in the state vectors and we will show in a later section how the state vectors or wave functions are changing with time.

Second, we consider how the Hamiltonian is transformed in the accelerating reference frame. Substituting (3.32) into (3.22), we may write

\[ \hat{\mathbf{h}}_0^a = S_a(t) H_0 S_a^\dagger(t) - \frac{1}{2} \mathbf{m} \hat{\mathbf{d}}(t) \cdot \hat{\mathbf{d}}(t) + \frac{1}{2} \mathbf{m} \hat{\mathbf{d}}(t) \cdot \hat{\mathbf{d}}(t) \]

\[ \cdot \hat{\mathbf{d}}(t) \cdot (S_a(t) \hat{\mathbf{p}}_0 S_a^\dagger(t)) + \mathbf{m} \hat{\mathbf{d}}(t) \cdot (S_a(t) \hat{x}_0 S_a^\dagger(t)) \]
Let us assume that \( H_o \) is a function of \( \dot{x}_o, \dot{p}_o \) and \( t \) and consists of kinetic energy and potential energy as usual,

\[
H_o = H(\dot{x}_o, \dot{p}_o, t) = (\dot{p}_o)^2/2m + V(\dot{x}_o, t) \tag{3.37}
\]

Since \( S_a(t) \) transforms \( \dot{x}_o \) and \( \dot{p}_o \) into \( \dot{x}_a \) and \( \dot{p}_a \) according to (3.35.b) and (3.36.b), we now rewrite \( \hat{H}_o^a \) as follows:

\[
\hat{H}_o^a = H(\dot{x}_a^a + \dot{a}(t), \dot{p}_a^a + m\dot{a}(t), t) - m\ddot{a}(t)\cdot \dot{a}(t)/2 + m\dddot{a}(t)\cdot \dddot{a}(t)/2
\]

\[
- \dddot{a}(t)\cdot (\dot{p}_a^a + m\ddot{a}(t)) + m\ddot{a}(t)\cdot (\dot{x}_a^a + \dot{a}(t))
\]

\[
= (\dot{p}_o^a + m\ddot{a}(t)) \cdot (\dot{p}_o^a + m\ddot{a}(t))/2m + V(\dot{x}_o^a + \dot{a}(t), t) - \dddot{a}(t) \cdot \dot{p}_o^a
\]

\[
+ m\ddot{a}(t) \cdot \dot{x}_o^a + m\ddot{a}(t) \cdot \dddot{a}(t)/2 - m\dddot{a}(t) \cdot \dddot{a}(t)/2
\]

so that

\[
\hat{H}_o^a = (\dot{p}_o^a)^2/2m + V(\dot{x}_o^a, \dot{a}(t), t) + m\ddot{a}(t)\dot{x}_o^a - m\dddot{a}(t)\dddot{a}(t)/2 \tag{3.38}
\]

It is interesting to see that the new Hamiltonian \( \hat{H}_o^a \) in the \( F^a \) frame of reference which describes the same physical system as in \( F \) frame of reference, has two additional terms besides the kinetic and potential energy terms. These last two terms in (3.38) are explicitly products of force with distance. For a non-inertial frame of reference \( F^a \), there must be an external force which accelerates the particle. If the frame of reference \( F^a \) is not accelerating but rather moving with constant velocity with respect to the \( F \) frame of reference, the last two terms in (3.38) vanish and the Hamiltonian satisfies the following relation:
\[ \hat{H}_O = U_O(t)H_0U_O^\dagger(t) = U_O(t)H(\hat{x}_O, \hat{p}_O, t)U_O^\dagger(t) = H(\hat{x}_O, \hat{p}_O, t) \]

This special case is classified as a Galilean transformation. Another important aspect to emphasize is that the Hamiltonian in equation (3.38) also satisfies Hamilton's equations of motion as it should in classical dynamics, even though the dynamical variables now refer to an accelerated frame of reference. It asserts the correctness of our general transformation theory. There is another transformation theory published in the literature (SCH77) in which the Hamiltonian does not satisfy Hamilton's equations of motion and the Schrödinger operators are time varying. These formulations of the theory are less satisfactory from our point of view.

Equation (3.19.b) gives the relationship between state vectors in frames of reference \( F \) and \( F^a \). We will demonstrate how the wave functions are related with each other in the next section.

3.5 Wave Functions and Translation Factors.

In scattering theory, we are interested in excitation and ionization probabilities which are calculated from wave functions. Most of scattering theory deals with wave packets which describe the propagation of a physical system in space and time. Here we represent the state vector in a coordinate representation, which is spanned by the eigenvectors of the position operator. Before we investigate how the wave functions in \( F \) and \( F^a \) are related, we introduce a new notation for bra/ket vectors. We shall write

\[ \hat{x}_O \mid \hat{x}_O; \hat{r} \rangle = \hat{r} \mid \hat{x}_O; \hat{r} \rangle \]  

(3.39)
where $\hat{x}_O$ is the position operator, and $\hat{r}=(x,y,z)$ is the eigenvalue of the eigenvector $|\hat{x}_O;\hat{r}\rangle$. $\hat{r}$ is a vector whose components are real numbers and is called the coordinate. (In Dirac's old terminology, $\hat{x}_O$ is a q-number, and $\hat{r}$ is a c-number.) The notation, which may appear redundant, is chosen in order to distinguish two or more eigenvectors of different operators but with the same eigenvalue. In this case the usual specification of an eigenvector by its corresponding eigenvalue is not adequate and sufficient. Let us first discuss some of the properties of this coordinate eigenvector.

Similar to equation (3.39), we also write

$$\hat{x}_O |\hat{x}_O;\hat{r}\rangle = \hat{r} |\hat{x}_O;\hat{r}\rangle$$  \hspace{1cm} (3.40)

$\hat{r}$ represents the coordinate in the frame of reference $F^a$. From the discussion of the transformation theory in the previous sections and equations (3.13), (3.4), and (3.12), we may write

$$\hat{x}_O |\hat{x}_O;\hat{r}\rangle = T^a(t)U(t)T^a(t)\hat{x}_O |\hat{x}_O;\hat{r}\rangle$$

$$= T^a(t)U(t)T^a(t)\hat{x}_O |\hat{x}_O;\hat{r}\rangle$$

$$= T^a(t)U(t)T^a(t)|\hat{x}_O;\hat{r}\rangle$$

Therefore

$$\hat{x}_O |\hat{x}_O;\hat{r}\rangle = \hat{r} (T^a(t)U(t)T^a(t)|\hat{x}_O;\hat{r}\rangle)$$  \hspace{1cm} (3.41)

where we have used the transformation between operators $\hat{x}_O$ and $\hat{x}_O^a$. Comparing the above equation with (3.40), we
can see that $|\hat{X}_O;\tau_a> \text{ and } (T^a(t)U_{ta}(t)T^a(t)|\hat{X}_O;\tau_a>)$ have the same eigenvalue, $\tau_a$, and we obtain the relation between vectors $|\hat{X}_O;\tau_a>$ and $|\hat{X}_O;\tau_a>$:

$$|\hat{X}_O;\tau_a> = S_a(t)U_{oa}(t) |\hat{X}_O;\tau_a>$$ (3.42)

where we have used

$$U_{oa}(t) = T(t)U_{ta}(t)T^a(t)$$

and equation (3.20).

To link the wave function with the state vector explicitly, we use the eigenvectors of the position operator $\hat{X}_O$ as a basis and write

$$\psi(\tau; t) = <\hat{X}_O;\tau|\psi(t)>$$ (3.43)

Similarly, in frame of reference $\mathbb{F}^a$,

$$\psi^a(\tau^a; t) = <\hat{X}_O;\tau^a|\psi^a(t)>$$ (3.44)

Substituting (3.19.b), (3.42) and (3.30.b) into (3.44), we get

$$\psi^a(\tau^a; t) = <\hat{X}_O;\tau^a|U_{oa}(t)S_a(t)S_a(t)|\psi(t)>$$

$$= <\hat{X}_O;\tau^a|U_{oa}(t)|\psi(t)>$$

$$= <\hat{X}_O;\tau^a|e^{-i\tilde{d}(t)\cdot\hat{P}_O} e^{i\tilde{d}(t)\cdot\hat{P}_O} e^{-im\tilde{d}(t)\cdot\hat{X}_O}|\psi(t)>$$

because $S_a(t)$ is a unitary operator. Since $\exp(-i\tilde{d}(t)\cdot\hat{P}_O)$ is a translation operator, whose effect is to shift the
coordinate by a constant distance \( \dot{d}(t) \), i.e.,

\[
e^{-i\dot{d}(t) \cdot \hat{p}_0} |\hat{x}_0; \hat{r}\rangle = |\hat{x}_0; \hat{r} + \dot{d}(t)\rangle \tag{3.45}
\]

we may write

\[
\psi_a(\hat{r}^a, t) = e^{i\dot{d}(t) \cdot \hat{r}^a(t) / 2} \langle \hat{x}_0; \hat{r}^a + \dot{d}(t) | e^{-i\dot{d}(t) \cdot \hat{x}_0} \psi(t) \rangle
\]

\[
= e^{i\dot{d}(t) \cdot \dot{x}_0} / 2 e^{-i\dot{d}(t) \cdot \hat{r}} \langle \hat{x}_0; \hat{r} | \psi(t) \rangle
\]

where we have used equation (3.23) and

\[
e^{i\dot{d}(t) \cdot \dot{x}_0} |\hat{x}_0; \hat{r}\rangle = e^{i\dot{d}(t) \cdot \hat{r}} |\hat{x}_0; \hat{r}\rangle
\]

we may therefore write the relation between wave functions \( \psi_a(\hat{r}^a, t) \) and \( \psi(\hat{r}, t) \):

\[
\psi_a(\hat{r}^a, t) = e^{i\dot{d}(t) \cdot \dot{x}_0} / 2 e^{-i\dot{d}(t) \cdot \hat{r}} \psi(\hat{r}, t) \tag{3.46.a}
\]

or conversely since \( \hat{r} = \hat{r}^a + \dot{d}(t) \),

\[
\psi(\hat{r}, t) = e^{i\dot{d}(t) \cdot \dot{x}_0} / 2 e^{-i\dot{d}(t) \cdot \hat{r}} \psi_a(\hat{r}^a, t) \tag{3.46.b}
\]

The phase factors in (3.46.a) and (3.46.b) are termed translation factors and have been the subject of much discussion in the literature of atomic collision physics. They provide the relation between the wave propagation in two different frames of reference F and \( F^a \) which are in relative motion. If there is no relative velocity between F and \( F^a \), i.e. \( \dot{d}(t) = 0 \), wave packets propagate the same in both frames. If there is a constant relative velocity,
i.e. \( \dot{\mathbf{d}}(t) = \mathbf{v} \) or \( \ddot{\mathbf{d}}(t) = \mathbf{v} t \), the translation factor becomes \( \exp\left( \imath m (v^2 t^2 / 2 - \mathbf{v} \cdot \mathbf{r}) \right) \), corresponding to a Galilean transformation.

3.6 Rotating Frames

In this section, we will consider another frame of reference \( F^R \) which rotates around a fixed axis (y-axis) with angular velocity \( \hat{\theta}(t) \) relative to frame of reference \( F \) as shown in Figure 3-2. In classical mechanics, the coordinates \( \mathbf{r}=(x,y,z) \) and \( \mathbf{r}^R=(x^R,y^R,z^R) \) in the frames of reference \( F \) and \( F^R \) respectively are related as follows:

\[
\begin{align*}
x^R &= -z \sin\theta(t) + x \cos\theta(t) \\
y^R &= y \\
z^R &= z \cos\theta(t) + x \sin\theta(t)
\end{align*}
\]

where \( \theta(t) \) is the angle of rotation.

In quantum mechanics, the relationship between the corresponding Heisenberg position operators \( \hat{\mathbf{x}}_t=(X_t,Y_t,Z_t) \) and \( \hat{\mathbf{x}}^R_t=(X^R_t,Y^R_t,Z^R_t) \) in frames \( F \) and \( F^R \) may be written as

\[
\begin{align*}
x^R_t &= -Z_t \sin\theta(t) + X_t \cos\theta(t) \\
y^R_t &= Y_t \\
x^R_t &= Z_t \cos\theta(t) + X_t \sin\theta(t)
\end{align*}
\]

The relation between the momentum operators are
\[ P_{x}^{R} = -P_{z} \sin \theta(t) + P_{x} \cos \theta(t) \]
\[ P_{y}^{R} = P_{y} \]
\[ P_{z}^{R} = P_{z} \cos \theta(t) + P_{x} \sin \theta(t) \]  \hspace{1cm} (3.49)

Here we choose a particular canonical transformation such that the position and momentum operators in the \( F^{R} \) frame are the position and momentum values projected from the frame of reference \( F \). Therefore, the transformations (3.48) and (3.49) are merely geometric and leave the kinematic relationships unchanged. In particular, the momentum is defined in terms of the motion with respect to the frame \( F \).

In contrast, in the case of transformation to a linear accelerating frame, we made a kinematic transformation and introduced momenta with respect to \( F^{A} \) as well as \( F \). An analogous canonical transformation between \( F \) and \( F^{R} \) in terms of rotating Cartesian coordinates is not possible, since the commutation relations cannot be preserved. Using polar coordinates, it is possible to construct such a canonical transformation and derive a Hamiltonian, but the formula are so complicated that it seems no practical use for the present purpose.

Corresponding to the transformations (3.48) and (3.49), a unitary rotation operator can be constructed in the form

\[ U_{tR}(t) = \exp (-i \, \hat{j} \cdot \hat{L}_{t} \theta(t)) \]  \hspace{1cm} (3.50.a)

where \( \hat{j} \) is the unit vector defining the axis of rotation and \( \hat{L}_{t} \) is the angular momentum operator. We may rewrite (3.50.a) as

\[ U_{tR}(t) = \exp (-i \, L_{y} \theta(t)) \]  \hspace{1cm} (3.50.b)
It is easy to show that this unitary operator accomplishes the transformation (3.48) and (3.49).

If we apply the time-development operator to $U_{OR}(t)$, we obtain the rotation operator in the Schrödinger picture

$$U_{OR}(t) = \exp(-iL\gamma_0 \theta(t))$$  (3.51)

Following the same procedure as we discussed in section 3.3, we find that the derivative of $U_{OR}^\dagger(t)$ with respect to time is

$$\frac{\partial U_{OR}^\dagger(t)}{\partial t} = i L\gamma_0 \dot{\theta}(t) \exp(iL\gamma_0 \theta(t))$$

so that

$$U_{OR}(t) \frac{\partial U_{OR}^\dagger(t)}{\partial t} = i L\gamma_0 \omega(t)$$  (3.52)

where $\omega(t) = \dot{\theta}(t)$ is defined as the angular velocity which is generally not constant. According to (3.21), the equation of motion of the operator $S_R(t)$ is

$$\frac{dS_R(t)}{dt} = S_R(t) (iL\gamma_0 \omega(t))$$

from which we can solve for $S_R(t)$:

$$S_R(t) = \exp \left( i L\gamma_0 (\theta(t)-\theta(0)) \right)$$

If we suppose that the two frames of reference coincide with each other at time zero, then $\theta(0)=0$, and $S_R(t)$ is
\[ S_R(t) = \exp \left( i \, L_y \, \theta(t) \right) \quad (3.53) \]

Now, applying the time-development operators to the \( x \)-components of position operators \( \hat{x}_t^R \) and \( \hat{x}_t \), we can further deduce the relation between the \( x \)-components of operators \( \hat{x}_O^R \) and \( \hat{x}_O \)

\[ x_O^R = T_R(t) x_t^R T_R^+(t) \]

\[ = T_R(t)(-Z_t \sin \theta(t) + X_t \cos \theta(t)) T_R^+(t) \]

\[ = -(T_R(t) T^+(t) Z_t T(t) T_R^+(t)) \sin \theta(t) + \]

\[ (T_R(t) T^+(t) Z_t T(t) T_R^+(t)) \cos \theta(t) \]

\[ = S_R(t) Z_O S_R^+(t)(-\sin \theta(t)) + S_R(t) X_O S_R^+(t)(\cos \theta(t)) \]

As demonstrated before, the results of the operator \( S_R(t) \) operating on \( Z_O \) and \( X_O \) are

\[ S_R(t) Z_O S_R^+(t) = Z_O \cos \theta(t) - X_O \sin \theta(t) \]

\[ S_R(t) X_O S_R^+(t) = X_O \cos \theta(t) + Z_O \sin \theta(t) \]

Therefore we conclude that

\[ x_O^R = x_O \]

This equation again exhibits the time-independence of the Schrödinger operators.

Now we can examine how the Schrödinger Hamiltonians in frames \( F \) and \( F_R \) differ from each other. Let us assume that the Hamiltonian \( H_O \) is a function of \( \hat{x}_O, \hat{p}_O \) and \( t \), as
expressed in equation (3.37). According to (3.22) and (3.52), the Hamiltonian \( \hat{H}_O^R \) in the frame of reference \( F^R \) can be written as

\[
\hat{H}_O^R(\hat{x}_O^R, \hat{p}_O^R, t) = H(\hat{x}_O^R, \hat{p}_O^R, t) - L_{y_0} \omega(t)
\]  \( (3.54) \)

If the frame of reference \( F^R \) does not rotate, that is \( \omega(t) = 0 \), then the two Hamiltonian \( H_O \) and \( \hat{H}_O^R \) are the same.

Finally we are going to show how the wave functions in frames of reference \( F \) and \( F^R \) are related to each other. We shall use the new notation introduced in Section 3.5. In the coordinate representation, the wave functions can be written in the forms

\[
\psi(\hat{r}, t) = \langle \hat{x}_O; \hat{r} | \psi(t) \rangle
\]

\[
\psi^R(\hat{r}^R; t) = \langle \hat{x}_O^R; \hat{r}^R | \psi^R(t) \rangle
\]

From the previous discussion, we may derive

\[
\psi^R(\hat{r}^R; t) = \langle \hat{x}_O^R; \hat{r}^R | S_R(t) | \psi(t) \rangle
\]

\[
= \langle \hat{x}_O^R; \hat{r}^R | U_{OR}(t) \psi(t) \rangle
\]

\[
= \langle \hat{x}_O; \hat{r} | \psi(t) \rangle
\]

or

\[
\psi^R(\hat{r}^R; t) = \psi(\hat{r}; t)
\]  \( (3.55) \)

Here we have used equation (3.19.b),

\[
| \hat{x}_O^R; \hat{r}^R \rangle = S_R(t) U_{OR}(t) | \hat{x}_O; \hat{r}^R \rangle
\]
which is similar to equation (3.42), and the fact that
\[ U_{0R}(t) = \exp(-iL_y \theta(t)) \]
will rotate the coordinates \( \vec{r}_R = (x_R, y_R, z_R) \) clockwise by an angle \( \theta(t) \).

Equation (3.55) has no rotation factor between wave functions. Also, as seen equation (3.54), the two Hamiltonians differ only by \( L_y \theta(t) \), whereas in the case of translations there are two correction terms. This result is explained by the purely geometric nature of the canonical transformation.
CHAPTER 4

NUMERICAL METHODS AND COMPUTATIONAL RESULTS

4.1 Determination of Matrix Elements

Armed with the theory developed in the last two chapters, we now set out to perform the actual calculation. Before solving the coupled equations, the matrix elements (2.18) must be evaluated first. The matrix elements are expressed in the C. M. frame of reference. As we discussed in Chapter 3, if we use different frames of reference, i.e., the target or the projectile frames of reference, and make the proper transformation, we may use the hydrogenic wave functions to evaluate the matrix elements.

To evaluate the direct coupling matrix elements $H_{mn}(A)$, we will use the projectile frame of reference. Applying equation (3.19.b) to transform the state vectors from the C.M. frame to the projectile frame, we get

$$H_{mn}(A) = \langle \phi_m(A,t) | H_0 - \frac{i}{\hbar} \frac{\partial}{\partial t} | \phi_n(A,t) \rangle$$

$$= \langle \phi^A_m(A,t) | S_A(t)(H_0 - \frac{i}{\hbar} \frac{\partial}{\partial t})S_A^+(t) | \phi^A_n(A,t) \rangle$$

where $| \phi^A_m(A,t) \rangle$ and $| \phi^A_n(A,t) \rangle$ are the Schrödinger state vectors in the projectile frame of reference. From equations (3.21) and (3.22), we obtain
\[
\begin{align*}
H_{mn}(A) &= \langle \phi^A_m(A, t) | \hat{\mathcal{H}}^A_O - i S_A(t) U_{OA}(t) \frac{\partial u^A_{OA}(t)}{\partial t} S^A_1(t) - i S_A(t) \frac{\partial s^A_1(t)}{\partial t} \rangle - i \hat{\mathcal{D}}^A_{n}(A, t) \\
&= \langle \phi^A_m(A, t) | \hat{\mathcal{H}}^A_O - i \left( \frac{dS_A(t)}{dt} S_A^\dagger(t) + S_A(t) \frac{ds_A^\dagger(t)}{dt} \right) - i \hat{\mathcal{D}}^A_{n}(A, t) \rangle
\end{align*}
\]

So
\[
H_{mn}(A) = \langle \phi^A_m(A, t) | \hat{\mathcal{H}}^A_O - i \frac{\partial}{\partial t} | \phi^A_n(A, t) \rangle \tag{4.1}
\]

Here we have used the unitary property:

\[
S_A(t) S_A^\dagger(t) = 1
\]

and

\[
\frac{dS_A(t)}{dt} S_A^\dagger(t) + S_A(t) \frac{ds_A^\dagger(t)}{dt} = 0
\]

The Schrödinger Hamiltonian \( \hat{\mathcal{H}}^A_O \) in equation (4.1) is in the projectile frame of reference. Since the projectile frame is linear accelerating and rotating with respect to the C. M. frame of reference in the coordinate representation, the Hamiltonian becomes according to (3.38) and (3.54)

\[
\hat{\mathcal{H}}^A_O = \frac{1}{2} (v^A)^2 - \frac{Z_1}{r_A} - \frac{Z_2}{|\vec{r}_A + \vec{R}(t)|} \dot{R}^A(t) \cdot \dot{R}^A(t) + \frac{e^A}{r_A} \cdot \frac{e^A}{r_A} + \frac{1}{2} \dot{r}_A(t) \cdot \dot{r}_A(t) - L^2 \omega(t)
\]

Here there are three correction terms: two corresponding to linear acceleration and one to rotation.

Similarly, if we use the target frame to evaluate the
direct coupling matrix elements $\overline{H}_{mn}(B)$, we write

$$
\overline{H}_{mn}(B) = \langle \phi_m(B,t) | H_O - i \frac{\partial}{\partial t} | \phi_n(B,t) \rangle
$$

$$
= \langle \phi_m^B(B,t) | \hat{H}_O^B - i \frac{\partial}{\partial t} | \phi_n^B(B,t) \rangle
$$

(4.3)

where $\hat{H}_O^B$ is defined as

$$
\hat{H}_O^B = \frac{Z_1}{2Z_2} \left[ \vec{r}_B \cdot \vec{r}_B^* - \frac{Z_2}{\vec{r}_B} \right] - \frac{Z_2}{\vec{r}_B} \left[ \vec{r}_B \cdot \vec{r}_B^* - \frac{Z_2}{\vec{r}_B} \right] - L^B \omega(t)
$$

(4.4)

and $|\phi_m^B(B,t)\rangle$ and $|\phi_n^B(B,t)\rangle$ are Schrödinger state vectors in the target frame of reference.

Finally, in the coupled equations, the exchange coupling matrix elements and the overlap matrix elements are defined as

$$
K_{mn}(A,B) = \langle \phi_m^B(A,t) | \hat{H}_O^B - i \frac{\partial}{\partial t} | \phi_n^B(B,t) \rangle
$$

(4.5)

$$
\overline{K}_{mn}(B,A) = \langle \phi_m^A(B,t) | \hat{H}_O^B - i \frac{\partial}{\partial t} | \phi_n^A(A,t) \rangle
$$

(4.6)

$$
S_{mn} = \langle \phi_m^A(A,t) | \phi_n^A(B,t) \rangle
$$

(4.7)

$$
S_{mn}^\dagger = \langle \phi_m^B(B,t) | \phi_n^B(A,t) \rangle
$$

(4.8)

The next consideration to arise is the total number of states to be used in the expansion. The more states we include, the more accurate the calculation will be, but the number of coupled equations that need to be solved
increases correspondingly. We decide to consider states with principal quantum number n=1 and n=2, which are a total of five states. We express the state vectors in the projectile and target frames of reference, both of which share a common y-axis perpendicular to the plane of projectile motion. Because of the reflection symmetry about the plane of motion, the Hamiltonians $\hat{H}_0^A$ and $\hat{H}_0^B$ both are even operators under the reflection $y \rightarrow -y$, and the five states break up into two groups. The four of these states, 1S, 2S, 2P_z, and 2P_x are even with respect to transformation from $y$ to $-y$. Only the state 2P_y is odd with respect to $y$ reflection, and because of selection rules, does not interact with the other four states. Therefore, we include only the 1S, 2S, 2P_z, and 2P_x hydrogenic states centered both on the target nucleus and the projectile ion. We write the general form of the wave function as

$$\langle \mathbf{r} | n \rangle = \psi_n(\mathbf{r}) \exp(-iE_n t) \quad (4.9.a)$$

where $E_n$ is the energy level of a hydrogenic atom with nuclear charge $z$.

$$E_n = \frac{z^2}{(2n^2)} \quad (4.9.b)$$

Then the four hydrogenic wave functions are

$$\psi_{1S}(\mathbf{r}) = R_{10} Y_0^0(\theta, \phi) = \frac{1}{\sqrt{\pi}} \ z^{3/2} \cdot \exp(-z r)$$

$$\psi_{2S}(\mathbf{r}) = R_{20}(r) Y_0^0(\theta, \phi) = \frac{1}{4\sqrt{2\pi}} \ z^{3/2}(2-zr)\exp(-z r/2)$$

$$\psi_{2P_z}(\mathbf{r}) = R_{21}(r) Y_1^0(\theta, \phi) = \frac{1}{4\sqrt{2\pi}} \ z^{3/2}(rz) \cdot \cos \theta \ \exp(-z r/2)$$

$$\psi_{2P_x}(\mathbf{r}) = R_{21}(r) Y_1^1(\theta, \phi) = \frac{1}{4\sqrt{2\pi}} \ z^{3/2}(rz) \cdot \sin \theta \ \exp(-z r/2)$$
\[ \phi_{2p}^x = \frac{1}{\sqrt{2}} R_{21}(r)(Y_{1}^{-1}(\theta, \phi) - Y_{1}^{1}(\theta, \phi)) \]

\[ = \frac{1}{4\sqrt{2}\pi} Z^{3/2} \sin \theta \cos \phi \exp(-Zr/2) \]

where \( R_{n\ell}(r) \) is the radial wave function and \( Y_{\ell}^{m} \) is a spherical harmonic (MOT65). \( Z \) is the charge of the nucleus, and \((r, \theta, \phi)\) are the polar coordinates of the electron. Therefore the equations (2.17) become 16 coupled equations. In our case, there is a complete lack of orthogonality within the expansion, because at finite time \( t \) eigenfunctions centered on different nuclei are generally not orthogonal. Only at \( t=+\infty \) and \( t=-\infty \), do the hydrogenic wave functions centered on different nuclei approach orthogonality. The hydrogenic wave functions centered on the same nuclei remain orthogonal through the whole time interval.

Explicit expressions for the direct coupling, exchange coupling, and overlap matrix elements are derived in Appendix A and B respectively.
4.2 Numerical Method

The coupled differential equations (2.17) can be written as

\[ \begin{bmatrix} a_{1s} \\ a_{2s} \\ a_{2p_z} \\ a_{2p_x} \\ a_{1si} \\ a_{2si} \\ a_{2p_{zi}} \\ a_{2p_{xi}} \\ b_{1s} \\ b_{2s} \\ b_{2p_z} \\ b_{2p_x} \\ b_{1si} \\ b_{2si} \\ b_{2p_{zi}} \\ b_{2p_{xi}} \end{bmatrix} \]

\[ = \begin{bmatrix} a_{1s} \\ a_{2s} \\ a_{2p_z} \\ a_{2p_x} \\ a_{1si} \\ a_{2si} \\ a_{2p_{zi}} \\ a_{2p_{xi}} \\ b_{1s} \\ b_{2s} \\ b_{2p_z} \\ b_{2p_x} \\ b_{1si} \\ b_{2si} \\ b_{2p_{zi}} \\ b_{2p_{xi}} \end{bmatrix} \]

\[ \begin{bmatrix} a_{1s} \\ a_{2s} \\ a_{2p_z} \\ a_{2p_x} \\ a_{1si} \\ a_{2si} \\ a_{2p_{zi}} \\ a_{2p_{xi}} \\ b_{1s} \\ b_{2s} \\ b_{2p_z} \\ b_{2p_x} \\ b_{1si} \\ b_{2si} \\ b_{2p_{zi}} \\ b_{2p_{xi}} \end{bmatrix} \]

(4.12)

and \( T \) and \( V \) are two 16 by 16 matrices:

\[
T = \begin{pmatrix}
H_r(A) & -H_i(A) & K_r(A,B) & -K_i(A,B) \\
H_i(A) & H_r(A) & K_i(A,B) & K_r(A,B) \\
K_r(B,A) & -K_i(B,A) & H_r(B) & -H_i(B) \\
K_i(B,A) & K_r(B,A) & H_i(B) & H_r(B)
\end{pmatrix}
\]

(4.13a)
\[ V = \begin{pmatrix} 0 & -I & -S_i & -S_r \\ I & 0 & S_r & -S_i \\ -S_i^\dagger & -S_r^\dagger & 0 & -I \\ S_r^\dagger & -S_i^\dagger & I & 0 \end{pmatrix} \] (4.13.b)

The subscripts 'r' and 'i' in the above matrix elements denote the real and imaginary parts. The matrix elements \( H(A), \overline{H}(B), K(A,B), \overline{K}(B,A) \), \( S \) and \( S^\dagger \) are themselves 4 by 4 matrices which are defined in equation (2.18).

We may rewrite equation (4.11) as

\[ \frac{d \hat{a}}{dt} = V^{-1} T \hat{a} = W \hat{a} \] (4.11.b)

This is the coupled differential equations which we want to solve.

The matrix elements of \( T \) and \( V \) in equation (4.13) are combinations of volume integrals over the electron coordinates. As shown in Table B-1, these volume integrals are double integrations over 'r' and 'θ' with the following format:

\[ D_{j\pm}(n_1,n_2,k,\ell,m,n) = \int_0^\infty dr \int_0^1 dx \, r^k e^{-\eta_1 r} \int_0^1 dx \, e^{-\eta_2 y_{\pm}(r,x)} e^{i R(t) x} \]

\[ J_j(\lambda r(1-x^2)^{\frac{1}{2}})(y_{\pm}(r,x))^\ell x^m (1-x^2)^{n/2} \]

where

\[ x = \cos \theta \]

\[ \lambda = v_1 b/R(t) \]

\[ y_{\pm}(r,x) = (r^2 + R(t)^2 \pm 2 R(t) r \cos \theta)^{\frac{1}{2}} \]
We evaluate these double integrals numerically. There are 21 different complex angular integrals (integrals over cosθ) and a total of 57 complex double integrals. We compute real and imaginary parts separately. For the angular integrals, we use the Clenshaw-Curtis quadrature (GEN72) which applies the Chebyshev series expansion. A fast Fourier transform algorithm is used to calculate the cosine transform coefficients from which the angular integrals can be derived. For the radial integrations, we use the modified Romberg quadrature method (FAI69). The range of integrations from 0 to ∞ for the radial coordinate 'r' is divided into subintervals:

\[ D_{j\pm}(\eta_1, \eta_2, k, \ell, m, n) = \int_{0}^{d_1} dr \int_{0}^{2d_2} dr' r^{k-\eta_1} r' F_{j\pm}(r, \eta_2, \ell, m, n) + \ldots \]

(4.14)

The integration procedure is terminated whenever the relative value of the contributions from a subinterval with respect to the total sum of the integral is less than the required tolerance. 'ELECAP' is the program which evaluates these double integrals. For an internuclear distance R(t), it takes an average of 10 minutes CPU time to evaluate all the exchange matrix elements (K_{mn}, \overline{K}_{mn}, S_{mn} and S^\dagger_{mn}) by using the IBM 360/370 optimization compiler (H-compiler).

After we compute all the matrix elements, we solve the coupled equations (4.11) by dividing the trajectory into three regions. In region I and III, the internuclear distance R(t) is the independent variable. Equation (4.11.b) becomes

\[ \frac{d\hat{a}}{dR} = (\dot{R}(t))^{-1} W \hat{a} \]

(4.15.a)
In region II, the angle $\phi$ is the independent variable. We may write equation (4.11.b) as

$$\frac{d\hat{a}}{d\phi} = (\dot{\phi}(t))^{-1} W \hat{a} = \frac{R^2(t)}{b\nu_1} W \hat{a} \quad (4.15.b)$$

Here we have used equation (2.3).

We used GEARS method or the variable-order ADAMS predictor-corrector method (GEA71), provided by an IMSL package, to integrate the coupled differential equations. For He$^{++}$ projectiles scattered from hydrogen atoms, we started our time integration at $R(t)=6$ a.u. at a negative time ($t<0$) and ended at $R(t)=6.0$ a.u. at a positive time ($t>0$). The integration variables $R(t)$ and $\phi(t)$ were exchanged at $R(t)=1.0\times10^{-2}$ a.u. It took about 400 steps to integrate the coupled differential equations. Because it is very expensive to evaluate the double integrations, we evaluated only 22 data points for different $R(t)$ values ranging from 10 a.u. to $2.074\times10^{-3}$ (the distance of closest approach). Then we use a cubic spline fit method to interpolate the matrix elements while solving the differential equations. The direct transition matrix elements, as shown in Table A-1, have explicit series expansions. So they are explicitly evaluated during the time integrations. The whole time integration process took about 6 minutes CPU time for IBM 360/370 optimization compiler (H-compiler).

A complete program listing is in Appendix D which includes three parts: ELECAP, TRANMAX, and DCOUPL.
4.3 Computational Results and Numerical Errors

Although the main purpose of this dissertation was to establish a method for evaluating the excitation and charge transfer probabilities with as few approximations as possible, one set of data was calculated as an illustration of the method for this dissertation. From the discussion of the previous section, it is clear that the numerical computation procedure is complex, lengthy, and expensive, but from the preliminary sample calculation we may understand the problem better and find out where difficulties lie. Further work with more realistic assumptions may be performed using the lessons of this dissertation.

Most previous calculations have assumed that the ion-atom collision can be adequately described by uniform motion on a straight-line trajectory. On the other hand, as discussed in the beginning of Chapter 2, the collision process of interest in this dissertation is characterized by two conditions: (1) the scattering angle is greater than 10 degrees and (2) the slowing down factor is about 10% [slowing down factor: \( \Delta v = (v_1 - v) / v_1 \), where \( v_1 \) is the incident projectile velocity and \( v \) is the relative velocity at the distance of closest approach]. From (2.2.e), the first condition gives

\[
\pi - 2 \tan^{-1}(b/a) = \theta > 10^\circ \quad (4.16)
\]

The second condition gives

\[
v/v_1 = b/(a+c) < 90\%
\]

(4.17)

Combining (4.16) and (4.17), we conclude that if the ratio of the impact parameter and the distance of closest approach is in the range:

\[0.9 < b/a < 9\]
the two conditions will be satisfied.

For the sample calculation, the following initial conditions were chosen:

\[ E = 100 \text{ Kev/amu} \]
\[ b = 5a = 1.04 \times 10^{-2} \text{ a.u.} \]
\[ z_1 = 2 \]
\[ z_2 = 1 \]

The initial values for the amplitudes were specified as in equation (2.19). Therefore this calculation examines the collision process for He$^{+}$ ions incident on hydrogen atoms at moderate velocities and relatively small but non-zero impact parameters.

Figures 4-1 to 4-4 represent the typical behavior of the matrix elements which always damp off at large internuclear distance $R$. The accuracy of the double integration was verified by checking the overlap matrix elements to determine if they satisfy the relation: $S_{\text{nm}}^* = S_{\text{mn}}^+$. For internuclear distance less than 1 a.u., the matrix elements satisfy this relation accurately. But at large internuclear distance, the imaginary parts of the matrix elements tend to deviate from the expected equality. These errors are attributable to errors in the double integration which occur for large internuclear distance $R$ when the matrix elements are numerically small. These errors may be reduced by changing the integration intervals while integrating over the electronic radial coordinate at large internuclear distance.

Figures 4-5 and 4-6 show the normalized excitation probabilities ($|b_{2S}|^2$, $|b_{2P_z}|^2$, and $|b_{2P_x}|^2$) and the normalized charge exchange probabilities ($|a_{1S}|^2$, $|a_{2S}|^2$, $|a_{2P_z}|^2$, and $|a_{2P_x}|^2$) versus the internuclear distance and time.
The internuclear distance is plotted on a logarithmic scale, so that there is a discontinuity at the distance of closest approach which corresponds to \( t=0 \). The general behavior of each probability is reasonable and shows that for this particular impact parameter the charge transfer to the 1S state is the largest and the most important of these probabilities.

It should be noted, however, that this result does not imply that the charge transfer to the 1S state is most probable when one sums over all impact parameters. On the qualitative argument that charge transfer is most probable to states with the least change in binding energy, one expects the total cross sections for transfer to the 2S state in \( \text{He}^{++)} \) to be dominant. (SHG74)(SHG76)(RAP73) It should be interesting to make the necessary calculations for a range of impact parameters to be able to estimate total cross sections.

As the collision proceeds from \( t=-\infty \), the total inelastic transition probability for removal of the electron from the 1S orbit of the target hydrogen atoms increases rapidly toward 30-40\%. Eventually, at the end of the collision as \( t=+\infty \), it approaches 50\%, but the calculation was not accurate enough to show satisfactory convergence.

As proved in Appendix C, in spite of the truncation of the coupled equations of motion and the use of basis states which are not linearly independent, the approximate wave functions obtained in this calculation satisfy in principle the condition of conservation of probability. However, because of numerical inaccuracy, the sample calculation gives an error as large as 60\% at the end of the collision as \( t=+\infty \). This error is accumulated from three sources:

1. the error from the matrix elements at large internuclear distance \( R \), as discussed above,
2. the error caused by interpolating the matrix elements, and
(3) the error accumulated while integrating the coupled differential equations.

In Chapter 5, some methods will be discussed to correct or reduce these errors.
Figure 4-1
$K_{2S1S}$ MATRIX ELEMENT

Figure 4-2
$S_{2Pz2Pz}$ MATRIX ELEMENT
$K_{1S2PZ}$ MATRIX ELEMENT

Figure 4-4
Figure 4-5

Normalized Excitation Probabilities

- $|b_{2s}|^2$
- $|b_{2P_z}|^2$
- $|b_{2P_x}|^2$

Internuclear Distance $R$ (a.u.)

Time $t$ (a.u.)
CHAPTER 5

CONCLUSION

The goal of this dissertation was to establish a method to solve the problem of inelastic inner-shell ion-atom collision with only truncation errors, by using a finite basis set, and to evaluate the excitation and charge exchange probabilities. Only one sample calculation was performed to demonstrate the procedure. This preliminary calculation will give us enough information to examine the collision procedure, reveal numerical difficulties, and seek better methods to improve the results. In this concluding chapter, we briefly discuss some possible future improvements, using the general approach developed here.

As discussed in Chapter 4, the sample calculation accumulates a considerable amount of error. There are three ways to improve the results:

(1) A smaller integration interval $d_r$ in equation (4.14) should be used for large internuclear distance $R$. But this change will require more steps to integrate, and will be more expensive.

(2) A different interpolation method should be used for different sets of matrix elements because the relationship between the matrix elements and the internuclear distance is different for each set of matrix elements. For the sample calculation, an interpolation routine handles four cases. A more complicated routine may be written and tested for more accurate interpolation. Of course, the modification will increase the complexity.

(3) A high-accuracy method (i.e. extrapolation method in IMSL packages) to solve the coupled differential equations
may be used to reduce the accumulated error. In the sample calculation, the GEARs method is of low-accuracy, but able to solve both nonstiff and stiff (i.e. rapidly and not-rapidly converging) coupled differential equations. Since the behavior of the coupled equations (2.17) is unknown, the GEARs method was chosen to produce a quick and preliminary evaluation for these excitation and charge exchange amplitudes. The sample calculation only took about 400 steps to integrate the coupled differential equations. It shows that this procedure is a non-stiff case. A high-accuracy, but costly, method may be used to eliminate the errors. (BUL66) (GRA65)

Investigation of the above improvements will consume about one-half man-year. To solve these numerical difficulties requires an understanding of numerical analysis and different 'experiments' to examine and compare the results to produce a more satisfactory result than the sample calculation. More discussion on this topic is beyond the scope of this dissertation.

As far as the physics is concerned, the basic method developed in this dissertation is a general one. Further work may be accomplished in the following directions:

1. A different basis set from the hydrogenic states may be chosen for the wave function expansion such that the matrix elements may have simple expressions. For much lower energy collisions, a molecular basis set may be used (PIA74). If a different basis set is used, the analytic expressions for all the matrix elements will change.

2. An effective charge may be used for the hydrogenic wave functions and for the nuclear potentials for partially stripped ion-ion collisions. For such a choice, all the expressions for the matrix elements derived in this dissertation can be used, and the only change is in using effective charges for $Z_1$ and $Z_2$. 
Only one value of the impact parameters was considered in this dissertation. Since most of the experimental data measure the total cross sections for the charge exchange process, probabilities for a range of impact parameters may be calculated as an extension of the present work. The total cross sections can be then derived by integrating over the impact parameters and compared with the experimental results. More experimental work in measuring impact-parameter dependent charge transfer probabilities is likely be performed in the future, affording opportunities for direct comparisons between the experimental data and the detailed calculation initiated in this dissertation.
References


FAI69  G. Fairweather, COMM, ACM 12 (June, 1969), 324


GAR73  J. D. Garcia, D. J. Portner, and T. M. Kavanaugh, Rev. Mod. Phys. 45, 111 (1973)


GEN72  W. M. Gentleman, COMM. ACM 15 (May 1972), 337


RAP73  D. Rapp, J. Chem. Phys. 58, 2043 (1973)


STO73  D. Storm, Phys. Rev. A, 8, 1765,(1979)


APPENDIX A

The Direct Coupling Matrix Elements

We will drive the direct coupling matrix elements $H_{mn}(A)$ and $\tilde{H}_{mn}(B)$ here. We rewrite equation (4.1) in an integral form so that we may evaluate the integrals:

$$H_{mn}(A) = \int d^3r_A \phi_m^A(A, \mathbf{r}_A) \left( \frac{Z_2}{r_B} + \frac{\mathbf{r}_A(t) \cdot \mathbf{r}_A}{2} + \frac{1}{r_B^2} \right) \phi_n^A(A, \mathbf{r}_A) e^{i(E_m^A - E_n^A)t}$$  \hspace{1cm} (A-1)

where we have used the coordinate representation for the state vectors and expanded in the projectile frame of reference:

$$\langle \mathbf{r}_A | \phi_n^A(A, t) \rangle = \phi_n^A(A, \mathbf{r}_A) e^{-iE_n^A t}$$

where $\mathbf{r}_A$ is the position vector of the active electron from the projectile $A$. $E_n^A$ are the energy levels of the projectile atom. The argument 'A' of $\phi_n^A(A, \mathbf{r}_A)$ denotes that the wave function is centered at the projectile. The superscript 'A' denotes the projectile frame of reference. $\phi_m^A(A, \mathbf{r}_A)$ and $\phi_m^B(B, \mathbf{r}_B)$ represent the projectile and target hydrogenic atomic wave functions respectively.

Applying equations (2.10) and (2.3), we rearrange the integral (A.1)

$$H_{mn}(A) = (-Z_2^2 D^{PA}(m,n) - \frac{Z_1 Z_2}{M_A R^2} D^{DA}(m,n) + \frac{M_B Z_1 Z_2}{M_A M_R} \delta_{mn} - \frac{V_1 b}{R^2} D^{RA}(m,n)) e^{i(E_m^A - E_n^A)t}$$ \hspace{1cm} (A-2)
where we introduce three new notations which are defined as follows:

\[
\text{DP}^A(m,n) = \int d^3 \tau_A \, \phi^*_m(A, \vec{r}_A) \frac{1}{r_B} \phi^A_n(A, \vec{r}_A) \tag{A-3}
\]

\[
\text{DD}^A(m,n) = \int d^3 \tau_A \, \phi^*_m(A, \vec{r}_A) r_A \cos \theta_A \phi^A_n(A, \vec{r}_A) \tag{A-4}
\]

\[
\text{DR}^A(m,n) = \int d^3 \tau_A \, \phi^*_m(A, \vec{r}_A) L^\phi_A \phi^A_n(A, \vec{r}_A) \tag{A-5}
\]

and \( \delta_{mn} \)

\[
\delta_{mn} = \begin{cases} 
0 & \text{if } m \neq n \\
1 & \text{if } m = n
\end{cases}
\]

(A-3), (A-4) and (A-5) are the so-called "potential" (or "radial"), "dipole", and "rotational" coupling terms.

Similar equations may be derived for \( \bar{H}_{mn}(B) \)

\[
\bar{H}_{mn}(B) = (-2\text{DP}^B(m,n) - \frac{Z_1 Z_2}{M_B R^2} \text{DD}^B(m,n) + \frac{M_A Z_1 Z_2}{2M_B M_R} \delta_{mn} - \frac{v_1 b}{R^2} \text{DR}^B(m,n)) \\
\times e^{i(E^B_m - E^B_n)t} \tag{A-6}
\]

where \( \text{DP}^B \), \( \text{DD}^B \), and \( \text{DR}^B \) are defined as

\[
\text{DP}^B(m,n) = \int d^3 \tau_B \phi^*_m(B, \vec{r}_B) \frac{1}{r_A} \phi^B_n(B, \vec{r}_B) \tag{A-7}
\]

\[
\text{DD}^B(m,n) = \int d^3 \tau_B \phi^*_m(B, \vec{r}_B) r_B \cos \theta_B \phi^B_n(B, \vec{r}_B) \tag{A-8}
\]

\[
\text{DR}^B(m,n) = \int d^3 \tau_B \phi^*_m(B, \vec{r}_B) L^\phi_B \phi^B_n(B, \vec{r}_B) \tag{A-9}
\]
These equations are the same format as (A-3), (A-4), and (A-5), except that the subscripts and superscripts A, B are interchanged. Table A-1 gives the formula of the relevant integrals.

When we evaluate the integrals, we first use $Z=Z_1$, $M=M_A$, and $\mathbf{r}=(r_A, \theta_A, \phi_A)$ for the basis set of the projectile atom. Then we substitute $Z=Z_2$, $M=M_B$, and $\mathbf{r}=(r_B, \theta_B, \phi_B)$. Because of the relative orientation of the two frames of reference, there are some integral values which need a change of sign. We will discuss the consequence of these sign adjustments in following sections:

(a) Potential coupling integrals:

Equation (A-3) may be written in the explicit integral forms:

$$D^A_{p,m}(m',n)=\int_0^{2\pi} d\theta_A \left( \frac{1}{r_A^2 + R^2(t) - 2r_A R(t) \cos \theta_A} \right)^{\frac{1}{2}} \int d\phi_A \exp(i(m_2-m_1)\phi_A)$$

where (A-7) is

$$D^B_{p,m}(m',n)=\int_0^{2\pi} d\theta_B \left( \frac{1}{r_B^2 + R^2(t) + 2r_B R(t) \cos \theta_B} \right)^{\frac{1}{2}} \int d\phi_B \exp(i(m_2-m_1)\phi_B)$$

where we have used the trigometric cosine law for $r_A$, $r_B$, and R(t). If $D^B_{p,m}(m',n)$ is evaluated by substituting $Z_2$ and $M_B$ for $Z_1$ and $M_A$ into equation (A-10), there is a sign
factor \((-1)^{\ell_1 + m_1 + \ell_2 + m_2}\) in the front because the associate Legendre polynomial \(P_{\ell_2}^{m_2}\) has following property:

\[
P_{\ell_2}^{m_2}(-x) = (-1)^{\ell_2 + m_2}P_{\ell_2}^{m_2}(x)
\] (A-12)

Table A-2 gives a summary of sign changes for interchanging A and B.

<table>
<thead>
<tr>
<th>((m, n))</th>
<th>(n_1, \ell_1, m_1)</th>
<th>(n_2, \ell_2, m_2)</th>
<th>change (Y:yes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>((1S, 1S))</td>
<td>1, 0, 0</td>
<td>1, 0, 0</td>
<td>N</td>
</tr>
<tr>
<td>((1S, 2S))</td>
<td>1, 0, 0</td>
<td>2, 0, 0</td>
<td>N</td>
</tr>
<tr>
<td>((1S, 2P_z))</td>
<td>1, 0, 0</td>
<td>2, 1, 0</td>
<td>Y</td>
</tr>
<tr>
<td>((1S, 2P_x))</td>
<td>1, 0, 0</td>
<td>2, 1, +1</td>
<td>N</td>
</tr>
<tr>
<td>((2S, 2S))</td>
<td>2, 0, 0</td>
<td>2, 0, 0</td>
<td>N</td>
</tr>
<tr>
<td>((2S, 2P_z))</td>
<td>2, 0, 0</td>
<td>2, 1, 0</td>
<td>Y</td>
</tr>
<tr>
<td>((2S, 2P_x))</td>
<td>2, 0, 0</td>
<td>2, 1, +1</td>
<td>N</td>
</tr>
<tr>
<td>((2P_z, 2P_x))</td>
<td>2, 1, 0</td>
<td>2, 1, 0</td>
<td>N</td>
</tr>
<tr>
<td>((2P_z, 2P_x))</td>
<td>2, 1, +1</td>
<td>2, 1, +1</td>
<td>N</td>
</tr>
</tbody>
</table>

Table A-2 \(\text{DP}^A(m, n) \rightarrow \text{DP}^B(m, n)\)

(b) Dipole coupling integrals:

Equations (A-4) and (A-8) are

\[
\text{DD}^A(m, n) = \int dr_A r_A^2 R_{n_1}^{\ell_1}(r_A) R_{n_2}^{\ell_2}(r_A) \int d(\cos \theta_A) \cos \theta_A P_{\ell_1}^{m_1}(\cos \theta_A)
\]

\[
P_{\ell_2}^{m_2}(\cos \theta_A) \int d\phi_A \exp(i(m_2 - m_1)\phi_A)
\] (A-13)
and

\[ DD^B(m,n) = \int dr_B r_B^3 R_{n_1}^B(r_B) R_{n_2}^B(r_B) \int d(\cos \theta_B) \cos \theta_B p_{\ell_1}^{m_1}(\cos \theta_B) \]

\[ p_{\ell_2}^{m_2}(\cos \theta_B) \int d\Phi_B \exp(i(m_2-m_1)\phi_B) \]  

(A-14)

Comparing (A-13) and (A-14), there is no sign change for evaluating \( DD^B(m,n) \) by substituting \( z_2 \) and \( M_B \) into equation (A-13).

(c) Rotational coupling integrals:

We write (A-5) and (A-9) as

\[ DR^A(m,n) = \int dr_A d^2 \Phi_A r_A^2 \sin \theta_A \phi_{n(A,\vec{r}_A)}^{A} \frac{L_+ - L_-}{2i} \phi_{n(A,\vec{r}_A)}^{A} \]  

(A-15)

and

\[ DR^B(m,n) = \int dr_B d^2 \Phi_B r_B^2 \sin \theta_B \phi_{n(B,\vec{r}_B)}^{B} \frac{L_+ - L_-}{2i} \phi_{n(B,\vec{r}_B)}^{B} \]  

(A-16)

If we substitute \( z_2 \) and \( M_B \) for \( z_1 \) and \( M_A \) in equation (A-15), we get \( DR^B(m,n) \) with no sign change.
\[
G(n, a) = \int_0^\infty \frac{r^n}{R(t)} \left(r+R(t) - |r-R(t)|\right) e^{-ar} \]
\[
= \frac{2(n!)}{a^{n+1}} \left[\frac{n+1}{aR(t)} - e^{-aR(t)} \left(\sum_{m=0}^{n-1} \frac{(n-m)}{aR(t)m!} + \frac{1}{m!} \right)\right] \]
\[
B(n, a) = \int_0^\infty \frac{r^n}{R(t)} \left(r+R(t) + |r-R(t)|e^{-ar}\right) \]
\[
= \frac{2(n!)}{a^{n+1}} \left[1 + \frac{e^{-aR(t)}}{aR(t)} \sum_{m=0}^{n} \frac{(n-m+1)}{m!} \right] \]
\[
DP^A(1S, 1S) = 2z_1^3 G(1, 2z_1) \]
\[
DP^A(1S, 2S) = \frac{z_1^3}{2\sqrt{2} \left[2G(1, 3z_1/2) - z_1G(2, 3z_1/2)\right]} \]
\[
DP^A(1S, 2P_z) = \frac{z_1^4}{6\sqrt{2}} \left[\frac{1}{R(t)} G(3, \frac{3z_1}{2}) + R(t) G(1, \frac{3z_1}{2}) - B(2, \frac{3z_1}{2})\right] \]
\[
DP^A(2S, 2P_z) = \frac{z_1^4}{48} \left[- \frac{Z_1}{R(t)} G(4, Z_1) + \frac{2}{R(t)} G(3, Z_1) - Z_1 R(t) G(2, Z_1) \right]
+ 2R(t) G(1, Z_1) - 2B(2, Z_1) + Z_1 B(3, Z_1) \]
\[
DP(2S, 2S) = \frac{z_1^3}{16} \left[4G(1, Z_1) - 4Z_1 G(2, Z_1) + z_1^2 G(3, Z_1)\right] \]
Table A-1 (cont't)

\[ DP^A(2P_z, 2P_z) = \frac{Z_1^5}{240} \left[ \frac{2}{R(t)^2} G(5, Z_1) + 2R(t)^2 G(1, Z_1) + 7G(3, Z_1) \right. \\
\left. - \frac{2}{R(t)} B(4, Z_1) - 2R(t) B(2, Z_1) \right] \]

\[ DP^A(2P_x, 2P_x) = \frac{Z_1^5}{240} \left[ 4G(3, Z_1) - \frac{1}{R(t)^2} G(5, Z_1) - R(t)^2 G(1, Z_1) \right. \\
\left. + \frac{1}{R(t)} B(4, Z_1) + R(t) B(2, Z_1) \right] \]

\[ DP^A(2S, 1S) = DP^A(1S, 2S) \]
\[ DP^A(2P_z, 2S) = DP^A(2S, 2P_z) \]

\[ DP^A(2P_z, 1S) = DP^A(1S, 2P_z) \]

All the other direct potential coupling terms are zero.

\[ DD^A(1S, 2P_z) = DD^A(2P_z, 1S) = \frac{128\sqrt{2}}{243Z_1} \]
\[ DD^A(2S, 2P_z) = DD^A(2P_z, 2S) = -\frac{3}{Z_1} \]

All the other direct dipole coupling terms are zero.

\[ DR^A(2P_z, 2P_x) = -i \quad DR^A(2P_x, 2P_z) = i \]

All the other direct radial coupling terms are zero.
APPENDIX B

The Exchange Coupling Matrix Elements

The evaluations of the exchange coupling matrix elements are more complicated than the ones of the direct coupling matrix elements discussed in Appendix A. Before we start to investigate the $K_{mn}(A,B)$ and $\bar{K}_{mn}(B,A)$ matrices, we must prepare ourselves by fixing the notation. If we use target frame of reference, the target coordinate representations of the state vectors are:

\[
\begin{align*}
\langle \mathbf{r}_{B} \mid \phi_{m}^{B}(A,t) \rangle &= \phi_{m}^{B}(A,\mathbf{r}_{B}) \exp(-i\mathbf{r}_{m}^{A}t) \\
\langle \mathbf{r}_{B} \mid \phi_{n}^{B}(B,t) \rangle &= \phi_{n}^{B}(B,\mathbf{r}_{B}) \exp(-i\mathbf{r}_{n}^{B}t)
\end{align*}
\]

(B-1)

where $\mathbf{r}_{B}$ is the position vector of the active electron from the target B. The subscript and superscript definitions please refer to Appendix A. Here $\phi_{n}^{B}(B,\mathbf{r}_{B})$ does represent the hydrogenic basis of the target atom which have the wave functions expressed in (4.9). $\phi_{m}^{B}(A,\mathbf{r}_{B})$ needs phase factors to transform to the hydrogenic basis of the projectile atom. Using the theory developed in Chapter 3 and equation (3.46), we relate the two wave functions as

\[
\phi_{m}^{B}(A,\mathbf{r}_{B}) = \exp(i\mathbf{r}(t) \cdot \mathbf{R}(t)/2) \exp(i\mathbf{r}(t) \cdot \mathbf{r}_{A}) \phi_{m}^{A}(A,\mathbf{r}_{A})
\]

or

\[
(B-2)
\]

\[
\phi_{m}^{A}(A,\mathbf{r}_{A}) = \exp(i\mathbf{r}(t) \cdot \mathbf{R}(t)/2) \exp(-i\mathbf{r}(t) \cdot \mathbf{r}_{B}) \phi_{m}^{B}(A,\mathbf{r}_{B})
\]
since $\mathbf{r_B} = \mathbf{r_A} + \dot{\mathbf{R}}(t)$. Now we are ready to examine the exchange coupling matrix.

Applying equations (4.4), (2.10), and (2.3) to (4.5), we get

$$K_{mn}(A,B) = (-Z_1 IP(A,B)_{mn} - \frac{Z_1 Z_2}{N_{BR}^2(t)} ID(A,B)_{mn} + \frac{MAZ_1 Z_2}{2M_{BR}^2(t)} S_{mn} - \frac{v_1 b}{R^2(t)} IR(A,B)_{mn} \exp(i(E_m - E_n) t))$$

(B-3)

where we again introduce four functions which are defined as following integrals:

$$IP(A,B)_{mn} = \langle \phi_m^B(A,t) | \frac{1}{\mathbf{r_A}} | \phi_n^B(B,t) \rangle$$

(B-4)

$$= \int d^3 r_A \phi_m^A^\ast(A,\mathbf{r_A}) e^{-i \mathbf{r}(t) \mathbf{R}(t)/2} e^{-i \mathbf{r}(t) \mathbf{r_A}} \frac{1}{\mathbf{r_A}} \phi_n^B(B,\mathbf{r_B})$$

$$ID(A,B)_{mn} = \langle \phi_m^B(A,t) | r_B \cos \theta_B | \phi_n^B(B,t) \rangle$$

(B-5)

$$= \int d^3 r_A \phi_m^A^\ast(A,\mathbf{r_A}) e^{i \mathbf{r}(t) \mathbf{R}(t)/2} e^{-i \mathbf{r}(t) \mathbf{r_B}} \cos \theta_B \phi_n^B(B,\mathbf{r_B})$$

$$S_{mn} = \int d^3 r_B \phi_m^A^\ast(A,\mathbf{r_A}) e^{i \mathbf{r}(t) \mathbf{R}(t)/2} e^{-i \mathbf{r}(t) \mathbf{r_B}} \phi_n^B(B,\mathbf{r_B})$$

(B-6)

$$IR(A,B)_{mn} = \langle \phi_m^B(A,t) | L_{YB}^- | \phi_n^B(B,t) \rangle$$

(B-7)

$$= \int d^3 r_B \phi_m^A^\ast(A,\mathbf{r_A}) e^{i \mathbf{r}(t) \mathbf{R}(t)/2} e^{-i \mathbf{r}(t) \mathbf{r_B}} \frac{L_{YB}^- L_{YB}^-}{2i} \phi_n^B(B,\mathbf{r_B})$$

(B-4), (B-5), and (B-7) are the "potential" (or "radial"), "dipole", and "rotational" coupling terms for the charge
exchange. (B-6) is the overlap integral.

Similar equations may be derived for $\bar{K}_{mn}(B,A)$:

$$
\bar{K}_{mn}(B,A) = (-Z_2 I P(B,A))_{mn} - \frac{Z_1 Z_2}{M_A R^2(t)} \text{ID}(B,A)_{mn} +
$$

$$
\frac{M_B Z_1 Z_2}{2 M_A M R(t)} S^+_{mn} - \frac{\nu_{1b}}{R^2(t)} \text{IR}(B,A)_{mn} \exp(i(E^B_m - E^A_n)t)
$$

(B-8)

where we introduce four functions which are defined as following integrals:

$$
I P(B,A)_{mn} = \langle \phi^A_m(B,t) | \frac{1}{r_B} | \phi^A_n(A,t) \rangle
$$

(B-9)

$$
= \int d^3 \tau_B \phi^B_m(B, \hat{r}_B) e^{-i \hat{R}(t) \hat{R}(t)/2} e^{i \hat{R}(t) \hat{r}_B} \frac{1}{r_B} \phi^A_n(A, \hat{r}_A)
$$

$$
I D(B,A)_{mn} = \langle \phi^A_m(B,t) | r_A \cdot \cos \theta_A | \phi^A_n(A,t) \rangle
$$

(B-10)

$$
= \int d^3 \tau_A \phi^B_m(B, \hat{r}_B) e^{i \hat{R}(t) \hat{R}(t)/2} e^{i \hat{R}(t) \hat{r}_A} \cos \theta_A \phi^A_n(A, \hat{r}_A)
$$

$$
S^+_{mn} = \int d^3 \tau_A \phi^B_m(B, \hat{r}_B) e^{i \hat{R}(t) \hat{R}(t)/2} e^{i \hat{R}(t) \hat{r}_A} \phi^A_n(A, \hat{r}_A)
$$

(B-11)

$$
I R(B,A)_{mn} = \langle \phi^A_m(B,t) | L^+ \cdot \phi^A_n(A,t) \rangle
$$

(B-12)

$$
= \int d^3 \tau_A \phi^B_m(B, \hat{r}_B) e^{i \hat{R}(t) \cdot \hat{R}(t)/2} e^{i \hat{R}(t) \cdot \hat{r}_A} \frac{L^+ \cdot L^-}{2i} \phi^A_n(A, \hat{r}_A)
$$
These equations have the same format as (B-4) to (B-7), except that the subscripts and superscripts A, B are interchanged. Table B-1 gives the formula of the relevant integrals.

We evaluate these matrix elements, IP(A,B)\(_{mn}\), ID(A,B)\(_{mn}\), S\(_{mn}(A)\), and IR(A,B)\(_{mn}\), by first substituting \(Z_1, Z_2\) and \(M_A, M_B\) into the expressions in Table B-1. Then we substitute exchanged \(Z_1\) and \(Z_2\), \(M_A\) and \(M_B\) values into the expressions in Table B-1 and change the integral variable \(\cos\theta\) into 

\(-\cos\theta\) to evaluate the IP(B,A)\(_{mn}\), ID(B,A)\(_{mn}\), S\(_{mn}(B)\), and IR(B,A)\(_{mn}\) matrix elements. There is a sign adjustment from this operation. For instance, the potential coupling integrals (B-4) and (B-9) may be written

\[
\text{IP(A,B)}_{mn} = 2\pi e^{-iR(t)\hat{R}(t)/2} \int dr_A r_A R_{n_1}^*(r_A) \int d(cos\theta_A) P_{\ell_1}^{|m_1|}(cos\theta_A) \\
e^{i\hat{R}(t)\hat{r}_A\cos\theta_B} J_{m_1-m_1_1} (v_1 r_A \sin\theta_A / R(t)) \\
P_{\ell_2}^{|m_2|}(cos\theta_B) R_{n_2}^*(r_A) \left[ (r_A^2 + R(t)^2 - 2r_A R(t) \cos\theta_A)^{1/2} \right] \quad (B-13)
\]

and

\[
\text{IP(B,A)}_{mn} = 2\pi e^{-iR(t)\hat{R}(t)/2} \int dr_B r_B R_{n_2}^*(r_B) \int d(cos\theta_B) P_{\ell_1}^{|m_1|}(cos\theta_B) \\
e^{-i\hat{R}(t)\hat{r}_B\cos\theta_A} J_{m_2-m_2} (v_1 r_B \sin\theta_B / R(t)) \\
P_{\ell_2}^{|m_2|}(cos\theta_A) R_{n_2}^*(r_B) \left[ (r_B^2 + R(t)^2 + 2r_B R(t) \cos\theta_B)^{1/2} \right] \quad (B-14)
\]

If we interchange A and B and transform

\[
r_A \rightarrow r_B \\
cos\theta_A \rightarrow -\cos\theta_B \\
\sin\theta_A \rightarrow \sin\theta_B \\
\phi_A \rightarrow \phi_B
\]
and from the Bessel function's property

\[ J_n(-x) = (-1)^n J_n(x) \]

(B-14) will have the same expression as (B-13) except a sign factor \((-1)^{m_1 - m_2}\) in the front. Therefore, whenever an odd order of Bessel function appears in the integrand, a sign should be adjusted.
Table B-1

\[ F_{0\pm}(r, n_2, \lambda, m, n) = \int dx \ e^{-\eta_2 y_{\pm}(r, x)} e^{i r \hat{R}(t)} x \ J_0(r \lambda (1-x^2)^{1/2}) \]
\[ (y_{\pm}(r, x))^\ell x^m (1-x^2)^{n/2} \]

\[ F_{1\pm}(r, n_2, \lambda, m, n) = \int dx \ e^{-\eta_2 y_{\pm}(r, x)} e^{i r \hat{R}(t)} x \ J_1(r \lambda (1-x^2)^{1/2}) \]
\[ (y_{\pm}(r, x))^\ell x^m (1-x^2)^{n/2} \]

\[ F_{2\pm}(r, n_2, \lambda, m, n) = \int dx \ e^{-\eta_2 y_{\pm}(r, x)} e^{i r \hat{R}(t)} x \ J_2(r \lambda (1-x^2)^{1/2}) \]
\[ (y_{\pm}(r, x))^\ell x^m (1-x^2)^{n/2} \]

\[ D_{0\pm}(n_1, n_2, k, \lambda, m, n) = \int dr \ r^k e^{-\eta_1 r} F_{0\pm}(r, n_2, \lambda, m, n) \]

\[ D_{1\pm}(n_1, n_2, k, \lambda, m, n) = \int dr \ r^k e^{-\eta_1 r} F_{1\pm}(r, n_2, \lambda, m, n) \]

\[ D_{2\pm}(n_1, n_2, k, \lambda, m, n) = \int dr \ r^k e^{-\eta_1 r} F_{2\pm}(r, n_2, \lambda, m, n) \]

where \( \lambda = v_1 b / R(t) \)

\[ y_{\pm}(r, x) = (r^2 + x^2 \pm 2r R(t)x)^{1/2} \]

and all above equations are functions of \( R(t) \), the internuclear distance.
Table B-1 (con't)

\[ \text{IP(A,B)}_{1S1S} = 2(z_1 z_2)^{3/2} \text{D}_{0-}(z_1, z_2, 1, 0, 0, 0) \]

\[ \text{IP(A,B)}_{1S2S} = \frac{(z_1 z_2)^{3/2}}{2 \sqrt{2}} [2\text{D}_{0-}(z_1, z_2/2, 1, 0, 0, 0) - z_2 \text{D}_{0-}(z_1, z_2/2, 1, 1, 0, 0)] \]

\[ \text{IP(A,B)}_{1S2P_z} = \frac{(z_1 z_2)^{3/2} z_2}{2 \sqrt{2}} [\text{D}_{0-}(z_1, z_2/2, 2, 0, 1, 0) - R(t)\text{D}_{0-}(z_1, z_2/2, 1, 0, 0, 0)] \]

\[ \text{IP(A,B)}_{1S2P_x} = \frac{(z_1 z_2)^{3/2} z_2}{2 \sqrt{2}} i [\text{D}_{1-}(z_1, z_2/2, 2, 0, 0, 1)] \]

\[ \text{IP(A,B)}_{2S1S} = \frac{(z_1 z_2)^{3/2}}{2 \sqrt{2}} [2\text{D}_{0-}(z_1/2, z_2, 1, 0, 0, 0) - z_1 \text{D}_{0-}(z_1/2, z_2, 2, 0, 0, 0)] \]

\[ \text{IP(A,B)}_{2S2S} = \frac{(z_1 z_2)^{3/2}}{16} [4\text{D}_{0-}(z_1/2, z_2/2, 1, 0, 0, 0) - 2z_1 \text{D}_{0-}(z_1/2, z_2/2, 2, 0, 0, 0) - 2z_2 \text{D}_{0-}(z_1/2, z_2/2, 1, 1, 0, 0) + z_1 z_2 \text{D}_{0-}(z_1/2, z_2/2, 2, 1, 0, 0)] \]

\[ \text{IP(A,B)}_{2S2P_z} = \frac{(z_1 z_2)^{3/2} z_2}{16} [2\text{D}_{0-}(z_1/2, z_2/2, 2, 0, 1, 0) - z_1 \text{D}_{0-}(z_1/2, z_2/2, 3, 0, 1, 0) - 2R(t)\text{D}_{0-}(z_1/2, z_2/2, 2, 1, 0, 0) + z_1 R(t)\text{D}_{0-}(z_1/2, z_2/2, 2, 0, 0, 0)] \]

\[ \text{IP(A,B)}_{2S2P_x} = \frac{z_2 (z_1 z_2)^{3/2}}{16} i [2\text{D}_{1-}(z_1/2, z_2/2, 2, 0, 0, 1) - z_1 \text{D}_{1-}(z_1/2, z_2/2, 3, 0, 0, 1)] \]
\begin{align*}
\text{IP}(A,B)_{2P_{z1S}} &= \frac{(z_1 z_2)^{3/2} z_1}{2 \sqrt{2}} [D_{2D}(z_1/2,z_2/2,0,0,1,0)] \\
\text{IP}(A,B)_{2P_{2S}} &= \frac{(z_1 z_2)^{3/2} z_1}{16} [2D_{2D}(z_1/2,z_2/2,1,1,0)] \\
\text{IP}(A,B)_{2P_{2P}} &= \frac{(z_1 z_2)^{5/2}}{16} [D_{2D}(z_1/2,z_2/2,0,0,1,0)] \\
\text{IP}(A,B)_{2P_{2P}} &= \frac{(z_1 z_2)^{5/2}}{16} [D_{2D}(z_1/2,z_2/2,0,0,1,0)] \\
\text{IP}(A,B)_{2P_{1S}} &= \frac{(z_1 z_2)^{3/2} z_1}{2 \sqrt{2}} [D_{2D}(z_1/2,z_2/2,0,0,1,0)] \\
\text{IP}(A,B)_{2P_{2S}} &= \frac{(z_1 z_2)^{3/2} z_1}{16} [2D_{2D}(z_1/2,z_2/2,1,1,0)] \\
\text{IP}(A,B)_{2P_{2P}} &= \frac{(z_1 z_2)^{5/2}}{16} [D_{2D}(z_1/2,z_2/2,0,0,1,0)] \\
\text{IP}(A,B)_{2P_{2P}} &= \frac{(z_1 z_2)^{5/2}}{32} [D_{2D}(z_1/2,z_2/2,0,0,2)] \\
\end{align*}
Table B-1 (cont’d)

\[ \text{ID}(A,B)_{1S1S} = 2(Z_1 Z_2)^{3/2} D_{0+}(Z_2/Z_1, 3, 0, 1, 0) \]

\[ \text{ID}(A,B)_{1S2S} = \frac{(Z_1 Z_2)^{3/2}}{2 \sqrt{2}} \frac{1}{[2D_{0+}(Z_2/Z_1, 3, 0, 1, 0) - Z_2 D_{0+}(Z_2/Z_1, 4, 0, 1, 0)]} \]

\[ \text{ID}(A,B)_{1S2P_z} = \frac{(Z_1 Z_2)^{3/2} Z_2}{2 \sqrt{2}} \frac{1}{[D_{1+}(Z_2/Z_1, 4, 0, 1, 1)]} \]

\[ \text{ID}(A,B)_{1S2P_x} = \frac{(Z_1 Z_2)^{3/2} Z_2}{2 \sqrt{2}} \frac{1}{i[D_{1+}(Z_2/Z_1, 4, 0, 1, 1)]} \]

\[ \text{ID}(A,B)_{2S1S} = \frac{(Z_1 Z_2)^{3/2}}{2 \sqrt{2}} \frac{1}{[2D_{0+}(Z_2/Z_1, 2, 3, 0, 1, 0) - Z_1 D_{0+}(Z_2/Z_1, 2, 3, 1, 1, 0)]} \]

\[ \text{ID}(A,B)_{2S2S} = \frac{(Z_1 Z_2)^{3/2}}{16} \frac{1}{[4D_{0+}(Z_2/Z_1, 2, 3, 0, 1, 0) - 2Z_1 D_{0+}(Z_2/Z_1, 2, 3, 1, 1, 0) - 2Z_2 D_{0+}(Z_2/Z_1, 2, 4, 0, 1, 0) + Z_1 Z_2 D_{0+}(Z_2/Z_1, 2, 4, 1, 1, 0)]} \]

\[ \text{ID}(A,B)_{2S2P_z} = \frac{(Z_1 Z_2)^{3/2}}{16} \frac{1}{[2D_{0+}(Z_2/Z_1, 2, 4, 0, 2, 0) - Z_1 D_{0+}(Z_2/Z_1, 2, 4, 1, 2, 0)]} \]

\[ \text{ID}(A,B)_{2S2P_x} = \frac{(Z_1 Z_2)^{3/2}}{16} \frac{1}{i[2D_{1+}(Z_2/Z_1, 2, 4, 0, 1, 1) - Z_1 D_{1+}(Z_2/Z_1, 2, 4, 1, 1, 1)]} \]
Table B-1 (con't)

\[ \text{ID}(A,B)_{2P,1S} = \frac{(Z_1 Z_2)^{3/2} Z_1}{2\sqrt{2}} [D_{0+}(Z_2/Z_1/2,4,0,2,0)+R(t)D_{0+}(Z_2/Z_1/2,3,0,1,0)] \]

\[ \text{ID}(A,B)_{2P,2S} = \frac{(Z_1 Z_2)^{3/2} Z_1}{16} [2D_{0+}(Z_2/Z_1/2,4,0,2,0)-Z_2 D_{0+}(Z_2/Z_1/2,5,0,2,0) \]
\[ + R(t)(2D_{0+}(Z_2/Z_1/2,3,0,1,0)-Z_2 D_{0+}(Z_2/Z_1/2,4,0,1,0))] \]

\[ \text{ID}(A,B)_{2P,2P} = \frac{(Z_1 Z_2)^{3/2} Z_1}{16} [D_{0+}(Z_2/Z_1/2,5,0,3,0)+R(t)D_{0+}(Z_2/Z_1/2,4,0,2,0)] \]

\[ \text{ID}(A,B)_{2P,2P} = \frac{(Z_1 Z_2)^{5/2}}{16} i[D_{1+}(Z_2/Z_1/2,5,0,2,1)+R(t)D_{1+}(Z_2/Z_1/2,4,0,1,1)] \]

\[ \text{ID}(A,B)_{2P,1S} = \frac{(Z_1 Z_2)^{3/2} Z_1}{2\sqrt{2}} iD_{1+}(Z_2/Z_1/2,4,0,0,1) \]

\[ \text{ID}(A,B)_{2P,2S} = \frac{(Z_1 Z_2)^{3/2} Z_1}{16} i[2D_{1+}(Z_2/Z_1/2,4,0,1,1)-Z_2 D_{1+}(Z_2/Z_1/2,5,0,1,1)] \]

\[ \text{ID}(A,B)_{2P,2P} = \frac{(Z_1 Z_2)^{5/2}}{16} iD_{1+}(Z_2/Z_1/2,5,0,2,1) \]

\[ \text{ID}(A,B)_{2P,2P} = \frac{(Z_1 Z_2)^{5/2}}{32} [D_{0+}(Z_2/Z_1/2,5,0,1,2)-D_{2+}(Z_2/Z_1/2,5,0,1,2)] \]
\[ S_{1S1S} = 2(z_1z_2)^{3/2} \ D_{O^+}(z_2, z_1, 2, 0, 0, 0) \]

\[ S_{1S2S} = \frac{(z_1z_2)^{3/2}}{2\sqrt{2}} [2D_{O^+}(z_2/2, z_1, 2, 0, 0, 0) - 2D_{O^+}(z_2/2, z_1, 3, 0, 0, 0)] \]

\[ S_{1S2P} = \frac{(z_1z_2)^{3/2}z_2}{2\sqrt{2}} D_{O^+}(z_2/2, z_1, 3, 0, 1, 0) \]

\[ S_{1S2P} = \frac{(z_1z_2)^{3/2}z_2}{2\sqrt{2}} i [D_{1^+}(z_2/2, z_1, 3, 0, 0, 1)] \]

\[ S_{2S1S} = \frac{(z_1z_2)^{3/2}}{2\sqrt{2}} [2D_{O^+}(z_2, z_1/2, 2, 0, 0, 0) - 2D_{O^+}(z_2, z_1/2, 2, 1, 0, 0)] \]

\[ S_{2S2S} = \frac{(z_1z_2)^{3/2}}{16} [4D_{O^+}(z_2/2, z_1/2, 2, 0, 0, 0) - 2D_{O^+}(z_2/2, z_1/2, 2, 1, 0, 0) - 2D_{O^+}(z_2/2, z_1/2, 3, 0, 0, 0) + 2D_{O^+}(z_2/2, z_1/2, 3, 1, 0, 0)] \]

\[ S_{2S2P} = \frac{(z_1z_2)^{3/2}z_2}{16} [2D_{O^+}(z_2/2, z_1/2, 3, 0, 1, 0) - 2D_{O^+}(z_2/2, z_1/2, 3, 1, 1, 0)] \]

\[ S_{2S2P} = \frac{(z_1z_2)^{3/2}}{16} i [2D_{1^+}(z_2/2, z_1/2, 3, 0, 0, 1) - 2D_{1^+}(z_2/2, z_1/2, 3, 1, 0, 1)] \]
Table B-1 (cont'

$$S_{2P \, 1S} = \frac{\langle z_1 z_2 \rangle^{3/2} \gamma_1}{2 \sqrt{2}} [D_{0+}(Z_2; z_1/2, 3, 0, 1, 0) + R(t) D_{0+}(Z_2; z_1/2, 2, 0, 0, 0)]$$

$$S_{2P \, 2S} = \frac{\langle z_1 z_2 \rangle^{3/2} \gamma_1}{16} [2D_{0+}(Z_2/2, z_1/2, 3, 0, 1, 0) - 2D_{0+}(Z_2/2, 2, 4, 0, 1, 0) + R(t) D_{0+}(Z_2/2, z_1/2, 2, 0, 0, 0) - 2D_{0+}(Z_2/2, z_1/2, 3, 0, 0, 0)]$$

$$S_{2P \, 2P} = \frac{\langle z_1 z_2 \rangle^{5/2}}{16} [D_{1+}(Z_2/2, z_1/2, 4, 0, 1, 1) + R(t) D_{1+}(Z_2/2, z_1/2, 3, 0, 0, 1)]$$

$$S_{2P \, 1S} = \frac{\langle z_1 z_2 \rangle^{3/2} \gamma_1}{2 \sqrt{2}} i D_{1+}(Z_2; z_1/2, 3, 0, 0, 1)$$

$$S_{2P \, 2S} = \frac{\langle z_1 z_2 \rangle^{3/2} \gamma_1}{16} i [2D_{1+}(Z_2/2, z_1/2, 3, 0, 0, 1) - 2D_{1+}(Z_2/2, z_1/2, 4, 0, 0, 1)]$$

$$S_{2P \, 2P} = \frac{\langle z_1 z_2 \rangle^{5/2}}{16} i D_{1+}(Z_2/2, z_1/2, 4, 0, 1, 1)$$

$$S_{2P \, 2P} = \frac{\langle z_1 z_2 \rangle^{5/2}}{32} [D_{0+}(Z_2/2, z_1/2, 4, 0, 0, 2) - D_{2+}(Z_2/2, z_1/2, 4, 0, 0, 2)]$$
Table B-1 (con't)

\[
\text{IR}(A, B)_{1S2P_z} = - \frac{(z_1 z_2)^{3/2} z_2}{2\sqrt{2}} \left[ D_{1+}(z_2/2, z_1, 3, 0, 0, 1) \right]
\]

\[
\text{IR}(A, B)_{1S2P_x} = \frac{(z_1 z_2)^{3/2} z_2}{2\sqrt{2}} (-i) \left[ D_{0+}(z_2/2, z_1, 3, 0, 1, 0) \right]
\]

\[
\text{IR}(A, B)_{2S2P_z} = \frac{(z_1 z_2)^{3/2} z_2}{16} \left[ 2D_{1+}(z_2/2, z_1/2, 2, 3, 0, 0, 1) - z_1 D_{1+}(z_2/2, z_1/2, 3, 1, 0, 1) \right]
\]

\[
\text{IR}(A, B)_{2S2P_x} = \frac{(z_1 z_2)^{3/2}}{16} i \left[ 2D_{0+}(z_2/2, z_1/2, 2, 3, 0, 1, 0) - z_1 D_{0+}(z_2/2, z_1/2, 3, 1, 1, 0) \right]
\]

\[
\text{IR}(A, B)_{2P_z 2P_z} = \frac{(z_1 z_2)^{5/2}}{16} \left[ D_{1+}(z_2/2, z_1/2, 2, 4, 0, 0, 1) + R(t) D_{0+}(z_2/2, z_1/2, 2, 3, 0, 0, 1) \right]
\]

\[
\text{IR}(A, B)_{2P_z 2P_x} = \frac{(z_1 z_2)^{5/2}}{16} i \left[ D_{0+}(z_2/2, z_1/2, 2, 4, 0, 2, 0) + R(t) D_{0+}(z_2/2, z_1/2, 2, 3, 0, 1, 0) \right]
\]

\[
\text{IR}(A, B)_{2P_x 2P_z} = \frac{(z_1 z_2)^{5/2}}{32} i \left[ D_{0+}(z_2/2, z_1/2, 4, 0, 0, 2) - D_{2+}(z_2/2, z_1/2, 2, 4, 0, 0, 2) \right]
\]

\[
\text{IR}(A, B)_{2P_x 2P_x} = \frac{(z_1 z_2)^{5/2}}{16} \left[ D_{1+}(z_2/2, z_1/2, 2, 4, 0, 1, 1) \right]
\]

All other IR terms are zero.
APPENDIX C

CONSERVATION OF PROBABILITY

The purpose of this appendix is to prove that the probability is conserved for the trial function (2.14)(GRE65). Summation convention is used here.

From (2.17), the first equation and its complex conjugates may be written

\[ a_n a_m H_{mn} (A) + a_m b_n K_{mn} (A, B) = i a_n a_m^* + i b_n a_m^* S_{mn} \]  \hspace{1cm} (C-1)

\[ a_n a_m H_{mn}^* (A) + b_n a_m K_{mn}^* (A, B) = -i a_n a_m^* - i b_n a_m^* S_{mn} \]  \hspace{1cm} (C-2)

and the second equation and its complex conjugates are

\[ a_n b_m K_{mn} (B, A) + b_m b_n H_{mn} (B) = i a_n b_m^* s_{mn}^* + i b_m b_n^* \]  \hspace{1cm} (C-3)

\[ a_n b_m K_{mn}^* (B, A) + b_m b_n H_{mn}^* (B) = -i a_n b_m^* s_{mn}^* - i b_m b_n^* \]  \hspace{1cm} (C-4)

Changing dummy indices in (C-2) and (C-4), we get

\[ a_n a_m H_{nm} (A) + b_m a_n K_{nm} (A, B) = -i a_n a_n^* - i b_m a_n^* S_{nm} \]  \hspace{1cm} (C-5)

\[ a_m b_n K_{nm} (B, A) + b_n b_m H_{nm} (B) = -i a_m b_n^* s_{nm}^* - i b_n b_n^* \]  \hspace{1cm} (C-6)

Combining (C-1), (C-5), (C-3), and (C-6) together and using the equation
we obtain

\[ i(a^*_m a_m + a_n^* a_n + b^*_m b_m + b_n^* b_n) + iS_{mn} \quad \left( b^*_m a_m^* + d^* b_n^* \right) + iS^+_{mn} \quad \left( a^*_n b_m^* + b^*_m a_n^* \right) \]

\[ = a_n a_m^* (H_{mn}(A) - H_{nm}(A)) + b_n b_m^* (\overline{H}_{mn}(B) - \overline{H}_{nm}(B)) + \]

\[ b_m^* a_n (\overline{K}_{mn}(B,A) - \overline{K}_{nm}(A,B)) + a_m^* b_n (K_{mn}(A,B) - K_{nm}(B,A)) \]

(C-7)

From the definition of the matrix elements (2.18), the following equations are valid:

\[ H_{mn}(A) = H_{nm}(A) = H^+_mn(A) \]

\[ \overline{H}_{mn}(B) = \overline{H}_{nm}(B) = \overline{H}_mn(B) \]

\[ K_{mn}(A,B) - \overline{K}_{nm}(B,A) = -i \frac{\partial}{\partial t} S_{mn} \]

\[ \overline{K}_{mn}(B,A) - K_{nm}(A,B) = -i \frac{\partial}{\partial t} S^+_{mn} \]

Hence, equation (C-7) becomes

\[ \frac{d}{dt} \left( a_m^* a_m + b_m^* b_m + S_{mn} \quad a^*_n b_n + S^+_{mn} \quad a_n b_m^* \right) = 0 \]

Q.E.D.
APPENDIX D

PROGRAM LISTING

The complete program consists of three parts:

Part I: ELECAP - Evaluate the double integrals. (p. 104)

Part II: TRANMAX - Combines the integral results to produce the matrix elements. (p. 133)

Part III: DCOUPL - Solves the coupled differential equations. (p. 140)

The JCL and the input data which produce the results shown in Figures 4-1 to 4-6 are given on page 169.
AN INCIDENT ION Z1 BOMBADES A TARGET ATOM Z2 WITH
* IMPACT PARAMETER B AND INCIDENT ENERGY E. FROM THESE
* INITIAL CONDITIONS, THE INCIDENT VELOCITY V1 AND A
* KEPLER TRAJECTORY OF THE PROJECTILE CAN BE DETERMINED.
* ALL CALCULATIONS ARE IN A.U.
*
* ALL THE CONSTANTS HERE ARE FROM A SMALL TABLE COMPILDE
* BY E. COREN, SCIENCE CENTER, ECKWELL INTERNATIONAL
* FROM PHYSICS TODAY, SEPTEMBER 1974.
*
PURPOSE---
EVALUATE THE DOUBLE INTEGRATIONS FOR MATRIX ELEMENT
CF HE++ ON H. SEE TABLE B-1.
USING BERGBER INTEGRATION OVER BO AND CLENSHAW-
CURTIS QUADRATURE OVER ANGLE.

INPUT---
WHILE: =1, READ FROM CARDS OR TERMINALS.
=12, READ FROM DISK FILE.
NCUT: =5, POINTS TO DISK FILE 5.
NBACH: =6, POINTS TO DISK FILE 6.
Z1, Z2: PROJECTILE AND TARGET CHARGE
M1, M2: MASS OF PROJECTILE, TARGET IN PROTON (APPROXIMATELY AMU)
BOTH OF THESE ARE INTEGERS.
NFUN: TOTAL NUMBER OF ANGULAR FUNCTIONS ARE EVALUATED.
NBND: TOTAL NUMBER OF RADIAL FUNCTIONS ARE EVALUATED.
LIMIT: MAXIMUM NUMBER OF SUBINTERVALS FOR ANGULAR INTEGRATION.
NLIM: MAXIMUM NUMBER OF ITERATIONS FOR DOUBLE INTEGRATION.
NCIFBO (NFUN): EACH ELEMENT IS THE NUMBER WHICH INDICATES
HOW MANY DIFFERENT POWERS OF BO CORRESPOND TO THE
SAME ANGULAR INTEGRATION.
NB: TWO-DIMENSION (16,2) ARRAY.
EACH ROW CORRESPONDS TO DIFFERENT COMBINATIONS OF
BESSEL FUNCTIONS AND TRIGONOMETRIC FUNCTIONS.
FIRST COLUMN CORRESPONDS TO ALL (EXP (-EDA2)*2) TERMS.
SECOND COLUMN CORRESPONDS TO ALL (EXP (-EDA2)) TERMS.
EL: PROJECTILE ENERGY IN KEY/AMU IN LAB SYSTEMS.
E, BA: INCIDENT-IMPACT PARAMETERS IN CM AND A.U.
VALUES ARE (1.0 AND 0.5) OR (0.5 OR 1.0)
DIFFERENT COMBINATIONS CORRESPOND TO DIFFERENT
SETS FOR CALCULATIONS.
TCLRE: REQUIRED TOLERANCE.
SMALL: =0, FOR SMALL RANGE OF R, THETA IS ACTUALLY USED TO
CALCULATE R.
=1, FOR LARGE RANGE OF R, R IS USED DIRECTLY.
THETA: FROM THIS VALUE, THE INTERNUCLEAR DISTANCE R CAN
BE DETERMINED FOR SMALL RANGE OF R.
R: INTERNUCLEAR DISTANCE IN A.U.
INPUT R MUST BE GREATER THAN THE CLOSEST APPROACH.

OUTPUT--

RHO: RADIAL RELATIVE VELOCITY OF THE PROJECTILE AND TARGET.
ECO: SCALED ELECTRON COORDINATE.
ECX, YO, Z: UPPER OR LOWER BOUNDS OF THE BO INTERVAL FOR EACH SUBINTEGRATION.
NUM: NUMBER OF FUNCTIONS LEFT FOR THE DOUBLE INTEGRATION.
MU: REDUCED MASS OF M21 AND M22 IN AU.
VI: INCIDENT VELOCITY IN AU.
V: ATOMIC UNIT OF VELOCITY IN CM/SEC
V: TANGENTIAL VELOCITY AT THE CLOSEST APPROACH.
BO: HALF OF THE COLLISION DIAMETER IN AU.
ED: ONE ATOMIC ENERGY UNIT IN EV.
SCAGL: SCATTERING ANGLE IN DEGREE
RZ: INTERNUCLEAR DISTANCE IN AU.
LAMDA: A FUNCTION OF R IF THE INCIDENT ENERGY AND THE IMPACT PARAMETER ARE GIVEN.

AT THE CLOSEST APPROACH, LAMDA EQUALS V.

FLAG: AN INDICATOR, (NFUNO) ADBAY.
(-2) MEANS TERMINATING THE INTEGRATION OVER BO FOR THE CORRESPONDING FUNCTION.

DBLINT: (NFUNO) THE RESULTS OF THE DOUBLE INTEGRATION.
UPPER; (NFUNO), EACH ELEMENT CORRESPONDS THE UPPER BOUNDS OF BO INTEGRATIONS.

* VALO(NFUNO), ESTERO(NFUNO), INDEX(NFUNO), POINTO(NFUNO)
* ARE EXPLAINED IN SUBROUTINE ROMINT
* THOSE ARE RESULTS OF BO INTEGRATION.

SUBPROGRAM USED:

ROMINT

REAL*8 VALO(114), ESTERO(114), DBLINT(114), UPPER(114)
REAL*8 RXA, RAX, RXB, BQ, C, BDCT,
1 RXEN, EDA2, RDL, EDA3, ROA, ROB, DFO,
2 RXQ, SCAGL, TLI22, THETA, TOLEBQ,
3 V, VI, VQ, Z212, PL, LAM1, LAM2, LAM3 (5)

MBRATIO: RATIO OF PROTON TO ELECTRON MASS.

REAL*8 LAMDA, M, M21A, MBRATIO/ 1.83615152E+3/

INTEGER INDEXO(114), POINTO(114), FLAG(114), NDIFRC(42, 2)
INTEGER ZT122, NANG(16, 2)

DATA E0/.-2721160885E+2/,
1 E/3.14159265358979E+0/

COMMON /CONST1, EDA2, EDA3
COMMON /CONST2, RXA, LAMDA
COMMON /INDI1, NFUNO, NFUNO, LIMIT
COMMON /INDI2, FLAG, NFUNO, NANG
COMMON /DATCL, TOLEBQ
COMMON /INTROL,INDEXA,FOINTO
COMMON /ESTROH,VALO,ESTIO

C* INPUT
ISW 0015     READ (1,1000) NFILF,NBACK,NOUT,NEON
ISW 0016     WRITE (3,1000) NFILF,NBACK,NOUT,NEON

ISW 0017     READ(1,1010) Z1,Z2,Z11,Z22,NFU,W,FUNO,LLIMT,NFLAG,
               1 (NDIFRO(J,1),NDIFRO(J,2),J=1,16)
               2 J=1,16

C* B IS THE RATIO OF IMPACT PARAMETER AND Bo.
C* THETA DETERMINES THE R VALUE.
C* I.E. THETA = 0, THE CLOSEST APPROACH.

ISW 0018     CONTINUE
ISW 0019     READ(1,1020,END=2000) EL,B,LAM1,LAM2,TOLER,SMALL,B,THETA

C* THE REQUIRED TOLERANCE OF THE SECOND (BO) INTEGRATION
C* IS 5 TIMES LARGER THAN THAT OF THE FIRST (ANGULAR)
C* INTEGRATION.
C* TOLBO=5*TOLER

ISW 0020

C* CHANGE EVERYTHING INTO A.U.
C* E IS THE REDUCED MASS, ENERGY IN A.U. IN RELATIVE COORDINATE
C* OF THE PROJECTILE AND TARGET.

ISW 0021     E=(1.0D+3)*EL/EO
ISW 0022     MU=DFLOAT(Z1*Z2)/(DFLOAT(Z1+Z2))
ISW 0023     E=MU*E

ISW 0024     Z12=Z1*Z2
ISW 0025     B0=Z12/(2*E)
ISW 0026     B1=B0
ISW 0027     V=DSQRT(Z2/WH)
ISW 0028     BA=DB2**B/BA
ISW 0029     V=V1*BA/(B0+C)
ISW 0030     SCANGL=180.0D0*(1.0D0-DBAN/(BA/B0))**2/FI
ISW 0031

C* FOR SMALL RANGE OF B, THETA IS THE INPUT
C* OTHERWISE, R IS THE INPUT

ISW 0032     IF (SMALL.EQ.0.0D0) R=BA2/(1-B0+C*DCOS(THETA))
ISW 0033     IF (SMALL.EQ.0.0D0) R=BA2/(1-B0+C*DCOS(THETA))
ISW 0034     LAMDA=V1*BA/R
ISW 0035     RDOT=DSQRT((Z-Z12/R)/MU-LAMDA**2)

C* AT THE CLOSEST APPROACH, VELOCITY ALONG THE
C* INTERNUCLEAR AXIS IS ZERO.
ISW 0036

ISW 0037     IF (SMALL.EQ.0.0D0.AND. THETA.EQ.0.0D0) RDOT=0.0D0
ISW 0038     LDA1=LAM1
ISN 0040  EDQ2=DAQ2
ISN 0041  EDQ3=DOT
ISN 0042  DBO=2*2.00/(EDQ4+EDQ2)
ISN 0043  WRITE(3,1040)E, V, Q, DOT, LAMDA, EDQ1, EDQ2, DBO
C  * CHECK TO SEE IF THE REST OF INPUT IS READ FROM DISK OR CARDS.
C
ISN 0044  IF (FILE.EQ.1) GO TO 60
ISN 0046  READ(NBACK,1030)ROA,NUM,(DBLINT(I),FLAG(I),I=1,NFUNO)
ISN 0047  GO TO 80
C  * INITIALIZE NUM. THE TWO ARRAY FLAG AND DBLINT.
C
ISN 0048  60  NUM=NFUNO
ISN 0049  DO 70 I=1,NFUNO
ISN 0050       FLAG(I)=0
ISN 0051  DBLINT(I)=0.00, DD
ISN 0052  70  CONTINUE
C
ISN 0053  IF (FQA=0.00, DD
ISN 0054  80  ROA=ROA+2ROB
ISN 0055
C  * INITIALIZE INDEXO BEFORE CALLING SUBROUTINE ROMINT.
C  * INDEXO IS AN INDICATOR SHOWING WHICH FUNCTION NEEDS
C  * INTEGRATION.
C
ISN 0056  DC 100 I=1,NFUNO
ISN 0057  IF (FLAG(I)=NFLAG)90, 95, 90
ISN 0058  INDEXO(I)=0
ISN 0059  GO TO 100
ISN 0060  INDEXO(1)=1
ISN 0061  100  CONTINUE
C
ISN 0062  CALL ROMINT(ROA,ROB,NFUNO,TOLERO,NEOM)
C
ISN 0063  DC 120 I=1,NFUNO
ISN 0064  IF (FLAG(I), EQ.,NFLAG) GO TO 120
ISN 0066  DBLINT(I)=DBLINT(I)+VALO(I)
ISN 0067  IF (DABS(VALO(I) GT. DABS(DBLINT(I)*TOLER)) GO TO 110
ISN 0069  IF (FLAG(I)=NFLAG)120,105,120
ISN 0070  105  NUM=NUM-1
ISN 0071  UPPER(I)=ROB
ISN 0072  GO TO 130
ISN 0073  IF (NUM.EQ.0) GO TO 130
ISN 0075  IF (FLAG(I)=0) CONTINUE
ISN 0076  110  FLAG(I)=0
ISN 0077  120  CONTINUE
C
ISN 0078  REWIND NBACK
ISN 0079  WRITE(NBACK,1030)ROA,NUM,(DBLINT(I),FLAG(I),I=1,NFUNO)
ISN 0080  ENDFILE NBACK
ISN 0081  ROA=ROB
ISN 0082  GO TO 80
C
ISN 0083  130  CONTINUE
  C
  C
ISN 0084   WRITE(NOUT,1010) Z1,Z2,RZ1,RZ2
ISN 0085   WRITE(NOUT,1040) M1,M2,EL,V1,BA,SCANL,R,HDOT,LAMDA,
          1   (DELM1T(1),I=1,MPUMO)
  C
ISN 0086   GO TO 10
  C
ISN 0087   1000  FORMAT(5I5)
ISN 0088   1010  FORMAT(2I5)
ISN 0089   1020  FORMAT(2D15.8)
ISN 0090   1030  FORMAT(2D25.16)
ISN 0091   1040  FORMAT(2D25.16)
  C
ISN 0092   2000  STOP
ISN 0093   END
**MODIFIED BOMBERG QUADRATURE**

This subroutine is applying modified Bomberg quadrature to integrate `nfuno` different functions from `a` to `b` (all integrals must have the same upper and lower bounds) to a required relative tolerance.

The algorithm in the reference has been rewritten in FORTRAN and modified for multi-function integrations.

This program will return valo(nfuno), estero(nfuno), pointo(nfuno) and indexo(nfuno) through the common blocks.

One can pass these variables into this subroutine through the input arguments.

Slightly change.

The maximum number of extrapolations is 15.


Subprograms Used:

DVALUE

A: LOWER BOUND.
B: UPPER BOUND.
NFUNO: NUMBER OF DIFFERENT FUNCTIONS.
N: NUMBER OF FUNCTION EVALUATIONS.
TOLER: THE DESIRED ACCURACY, RELATIVE TOLERANCE.
ESTERO: A (NFUNO) ARRAY OF ESTIMATED RELATIVE ERRORS.
VALO: A (NFUNO) ARRAY OF INTEGRATED RESULTS.
POINTO: AN (NFUNO) ARRAY TO RECORD NUMBER OF FUNCTION EVALUATIONS.
INDEXO: A (NFUNO) ARRAY INDICATOR.
INITIALLY PASSED IN BY COMMON BLOCK.
0: THE CORRESPONDENT FUNCTION NEEDS TO INTEGRATE.
1: THIS FUNCTION DOES NOT NEED TO INTEGRATE.
IF THE ESTIMATED INTEGRAL REACHES THE REQUIRED TOLERANCE, CHANGE 0 TO 1.
T: A (NFUNO) ARRAY OF THE IMPROVED TRAPEZOID VALUES.
PM: A (NFUNO, 15) TWO-DIMENSION ARRAY OF THE ROMBERG TABLE.
F: A TEMPORARY (NFUNO) ARRAY.
NUM: INITIALLY SET TO THE NUMBER OF FUNCTIONS NEED INTEGRATION.
DECREASES BY 1 WHEN ONE INTEGRAL REACHES THE TOLERANCE.
EQUALS 0 WHEN ALL THE INTEGRALS ARE ALL DONE.
S, S1, S0: (NFUNO) TEMPORARY ARRAYS OF SUMS.
DVALUE: A SUBROUTINE RETURNS THE FUNCTION EVALUATIONS.
ARGUMENTS (X, F): INPUT X AND FUNCTION VALUES F.
C REAL*8 A, B, SH, H, Z, R, BB, TOLERO
C REAL*8 F(114), T(114), R(114, 15), VALO(114),
     ESTERO(114), S(114), S1(114), S0(114)
C REAL*8 OLD(114), CGLDRE(114)
C INTEGER INDEXO(114), POINTO(114)
C COMMON /ESTERO/VALO, ESTERO
C COMMON /INDEXO, POINTO, TOLERO
C
C SH=B-A
C
C * IF TESTING THE PROGRAM, SET ITESTO=1
C ITESTO=1
C
C * INITIAL TRAPEZOID VALUE
C CALL DVALUE(A, F)
C CALL DVALUE(B, T)
C
C H=SH*.5D0

C * BY THE FOLLOWING LOOP, OBTAIN THE NUMBER OF FUNCTIONS
C * NEED TO INTEGRATE.
C NUM=0
C DC 20 I=1, NFUNO
C IF (INDEXO(I).EQ.1) GO TO 20
C T(I)=H*(F(I)+T(I))
C NUM=NUM+1
C
C 20 CONTINUE

C * INITIAL RECTANGLE VALUE
C X=(A+B)*.5D0
C CALL DVALUE(X, F)
C DC 30 I=1, NFUNO
C IF (INDEXO(I).EQ.1) GO TO 30
C BM(I, 1)=SH*T(I)
C GLD(I)=T(I)+B(R(I, 1))
C IF (OLD(I)) 24, 26, 24
C X=CLDRE(I), EO
C GO TO 30
C 26 CLDRE(I)=DABS(T(I)-BM(I, 1))
C 24 CONTINUE
C
C 30 CONTINUE

C N=2
C R=4
C DC 11 K=1, NKHM
C BB=(R*.5D0-1.0D0)/(R-1.0D0)
C * IMPROVED TRAPEZOID VALUE
C
ISN 0037 DO 40 I=1,NPDF10
ISN 0038 IF (INDEX0(I),EQ,1) GO TO 40
ISN 0040 S(I)=0
ISN 0041 T(I)=RM(I,1)+BB*(I(I)-RM(I,1))
ISN 0042 40 CONTINUE
C
C * DOUBLE NUMBER OF SUBDIVISIONS OF (A,B).
N=2*N
N=SH/DFLOAT(N)
C
ISN 0045 IF (N-32) 1,1,2
ISN 0046 1 N0=N
ISN 0047 GO TO 4
ISN 0048 2 N0=32
ISN 0049 3 IF (N-512) 4,4,5
ISN 0050 4 N1=N
ISN 0051 GO TO 6
ISN 0052 5 N1=512
C
ISN 0053 DO 9 K2=1,N,512
ISN 0054 DO 50 I=1,NPDF10
ISN 0055 50 S1(I)=0
C
ISN 0056 KK=K2+N1-1
C
ISN 0057 DO 8 K1=K2,KK,32
ISN 0058 DO 60 I=1,NPDF10
ISN 0059 60 S0(I)=0
C
ISN 0060 KKK=K1+N0-1
C
ISN 0061 DO 7 KG=K1,KKK,2
ISN 0062 X=A+DFLOAT(KG)*B
ISN 0063 CALL DVALUE(X,F)
ISN 0064 DO 70 I=1,NPDF10
ISN 0065 70 IF (INDEX0(I),EQ,1) GO TO 70
ISN 0066 S0(I)=S0(I)+F(I)
ISN 0067 70 CONTINUE
ISN 0068 70 CONTINUE
C
ISN 0070 DO 80 I=1,NPDF10
ISN 0071 IF (INDEX0(I),EQ,1) GO TO 80
ISN 0072 S1(I)=S0(I)+S1(I)
ISN 0074 80 CONTINUE
ISN 0075 80 CONTINUE
C
ISN 0076 DO 90 I=1,NPDF10
ISN 0077 IF (INDEX0(I),EQ,1) GO TO 90
ISN 0079 S(I)=S1(I)
ISN 0080 90 CONTINUE
ISN 0081 90 CONTINUE
DO 100 I=1,NFUN0
IF (INDEX0(I).EQ.1) GO TO 100
RM(I,N+1)=2.0*H*E'S I')
CONTINUE
C * END CALCULATION OF RECTANGLE VALUE.
C B=4
C * FORM BONBERG TABLE FROM RECTANGLE VALUE.
DO 10 J=1,K
L=E4+J
DO 110 I=1,NFUN0
IF (INDEX0(I).EQ.1) GO TO 110
RM(I,L)=RM(I,L+1)+(RM(I,L+1)-RM(I,L))/(E-1.DO)
CONTINUE
C B=4, DO*E
CONTINUE
C CHECK IF THE INTEGRATION REACHES THE TOLERANCE.
C OLD(I) AND OLDEB(I) RECORD THE PREVIOUS INTEGRATED
C VALUES AND ABSOLUTE ERRORS.
C VALO AND ESTERO RECORD THE CURRENT INTEGRATED
C VALUES AND ABSOLUTE ERRORS.
DO 120 I=1,NFUN0
IF (INDEX0(I).EQ.1) GO TO 120
ESTERO(I)=DABS(T(I)-RM(I,1))
IF (ESTERO(I).GT.DABS(TOLERO*VALO(I))) GO TO 160
ESTERO(I)=DABS(ESTERO(I)/VALO(I))
VALO(I)=VALO(I)*.520
GO TO 170
C IF THE CURRENT RELATIVE ERROR STARTS INCREASING,
C L.E. GREATER THAN THE PREVIOUS ONE, TAKE THE
C PREVIOUS INTEGRATED VALUE AND ERROR AND
C STOP GOING INTO FURTHER STEPS.
C ALSO PRINT OUT THIS SPECIAL CASE.
C OTHERWISE, DIVIDE MORE SUBINTERVALS AND
C INTEGRATE AGAIN.
IF (K.LE.3) GO TO 180
IF (ESTERO(I)-OLDEB(I)) 180,190,190
OLDEB(I)=LABS(OLDEB(I)/OLD(I))
VALO(I)=OLD(I)*.520
SUM=NUM-1
INDEX0(I)=1
GO TO 120
OLDEB(I)=ESTERO(I)
OLD(I)=VALO(I)
CONTINUE
C = IF TESTING THIS PROGRAM, SET ITESTO=1.
C = THEN IT WILL PRINT OUT 'I OUTPUT.'

ISH 0121       IF (ITESTO.EQ.0)GO TO 135
ISH 0123       IF (K.LT.3)GO TO 135
ISH 0125       WRITE (3,1000)K
ISH 0126       DO 130 I=1,NPUNO
ISH 0127       IF (INDEX(I).EQ.1)GO TO 130
ISH 0129       WRITE (3,1010)I,N,VALO(I),ESTERO(I)
ISH 0130    130   CONTINUE
ISH 0131       IF (NUM.EQ.0)GO TO 12
ISH 0133       11   CONTINUE
ISH 0134       WRITE (3,1020)K
ISH 0135       12   RETURN
ISH 0136       1000  FORMAT(15)
ISH 0137       1010  FORMAT(5X,215,2D20.8)
ISH 0138       1020  FORMAT(15,25H REFINEMENT NOT ALLOWED)
ISH 0139       1030  FORMAT(16H SPECIAL CASE 215,2D25.16)
ISH 0140       C    END
SUBROUTINE VALUE(X,RO)

CALCULATE INTEGRANT F(NFUNO) WITH INPUT X (=RO).
ANSINT IS A SUBROUTINE DOING THE ANGULAR-INTEGRATION.
* F(X) =

ARGUMENTS ARE:
D(NFUNO), FLAG(NFUNO), INDEX0(NFUNO), POINT0(NFUNO),
REINT(NFUN), ESTEP(NFUN), INDEX(NFUN), POINT(NFUN),
NDFTEX(NFUN,2)

SUBROUTINE USED:
* ANSINT

REAL*8 EDA1, EDA2, EDA3, RO, X, Y

COMMON /CONST1/EDA1, EDA2, EDA3
COMMON /INDEX1/INDEX0(NFUNO), LIMIT
COMMON /INDEX2/FLAG, NDFTEX, NANG
COMMON /VARIABLE/RO
COMMON /INTARY/INDEX, POINT
COMMON /INTARY/INDEX, INDEX0, POINT0

RO=X

IF (RO) 9,4,9
L I=1,NFUNO
D(I)=0.0

GO TO 100

INITIALIZE ROEXP(5,2)
ROEXP(5,2) IS AN ARRAY WHICH HAS VALUES OF
EXP(-EDA1*RO) *RO**N AND EXP(-EDA1*2*RO) *RO**N
N IS THE INDEX OF ROW.

Y=ROEXP(-EDA1*RO)
ROEXP(1,1)=Y*RO
ROEXP(1,2)=ROEXP(1,1)*Y

DC 10 K=2,5
ROEXP(K,1)=ROEXP(K-1,1)*RO
ROEXP(K,2)=ROEXP(K,1)*Y
CONTINUE
C * INITIALIZE INDEX(NFUN)
C * INDEX(NFUN) IS AN INDICATOR AND EXPLAINED IN SUBROUTINE
C * ANGIST

C

ISN 0025
ISN 0026

C

NFUN2=NFUN/2
NFUN02=NFUN0/2

C

ISN 0027
ISN 0028

DC 20 J=1,NFUN,2
K=KK
K=NDIFRO(J,1)+NDIFRO(J+1,1)
J2=(J+1)/2

C * REAL PART OF INTEGRAL F(I).
C
INDEX(J2)=1

C * IMAGINARY PART OF INTEGRAL F(I).
C
J2=J2+NFUN2
INDEX(JTEMP)=1
DO 14 JJ=1,K
K=K1+1
IF (INDEX(K1).EQ.1) GO TO 12
INDEX(J2)=0
12 KTEMP=K1+NFUN02
IF (INDEX(KTEMP).EQ.1) GO TO 14
INDEX(JTEMP)=0
14 CONTINUE

ISN 0045

KK=KK+K

ISN 0046

CONTINUE

C
INDEX1 IS A TEMPORARY INDICATOR ARRAY

C

ISN 0047
ISN 0048

DO 22 I=1,NFUN
INDEX1(I)=INDEX(I)

ISN 0049

C * CALL ANGIST (-1.0D0,1.DO,NFUN,LIMIT)
C
C
ISN 0050
ISN 0051
KK=NFUN02

ISN 0052

DO 90 J=1,NFUN2

ISN 0053

IF (INDEX1(J).EQ.1) GO TO 50
DO 50 JJ=1,2
L1=NDIFRO(2*J+II-2,1)
IF (L1) 25,40,25
L2=NDIFRO(2*J+II-2,2)-1
DO 30 JJ=1,L1
L1=L1+1
30 JJ=1,L1
IF (INDEX1(LL).EQ.1) GO TO 30
D(LL)=BEINT(J)*BOEIP(L2+JJ,II)

ISN 0064

CONTINUE

ISN 0065

CONTINUE

ISN 0066

GO TO 55

ISN 0067

GO TO 55

ISN 0068

L1=L1+NDIFRO(2*J-1,1)+NDIFRO(2*J+1,1)

ISN 0069

IF (INDEX1(JK).EQ.1) GO TO 80

ISN 0071

DC 70 II=1,2
L1=NDIFR0 (2*J+II-2, 1)
IF (L1) 57, 70, 57
L2=NDIFR0 (2*J+II-2, 2)-1
DO 60 JJ=1, L1
KK=KK+1
IF (INDEXO(KK), EQ. 1) GO TO 60
D(KK)=NEWINT(JK)*ROZIP(L2+JJ, II)
CONTINUE
60 CONTINUE
GO TO 90
80 CONTINUE
C
100 RETURN
END
**CLENSHAW-CURTIS QUADRATURE**

- **EVALUATE THE ANGULAR INTEGRATION BY CHEBYSHEV SERIES**
- **EXPANSION, CLENSHAW-CURTIS QUADRATURE.**
- **BY FAST FOURIER TRANSFORM ALGORITHM, THE COSINE**
- **TRANSFORM COEFFICIENTS CAN BE CALCULATED.**

**REFERENCE:**


**THE ORIGINAL FORTRAN PROGRAM IN THE ABOVE REFERENCE**

**DOES NOT WORK IN IBM 360/370 COMPUTER SYSTEM.**

**SO THIS SUBROUTINE HAS BEEN MODIFIED TO BE USED IN**

**IBM 360/370 SYSTEM AND ALSO TO BE ABLE TO INTEGRATE**

**NFUN AT THE SAME TIME FROM A TO B TO THE REQUESTED**

**RELATIVE ACCURACY WHILE USING NO MORE THAN LIMIT**

**FUNCTION EVALUATIONS.**

**ALL THE INTEGRALS MUST HAVE THE SAME UPPER AND LOWER**

**BOUNDS.**

**IN ORDER TO SAVE SOME STORAGE, PUT NEWINT, ESTERR**

**AND INDEX INTO COMMON BLOCKS WITH THE OUTSIDE PROGRAM.**

**AN ALTERNATIVE METHOD IS PASSING IN NEWINT, ESTERR**

**AS ARGUMENTS, I.E.**

**SUBROUTINE ANGINT(A,B,NFUN,LIMIT,NEWINT,ESTERR)**

**THIS SUBROUTINE WILL RETURN NEWINT(NFUN)**

**ESTERR(NFUN), POINTS(NFUN) AND INDEX(NFUN)**

**THROUGH THE COMMON BLOCKS.**

**SUBROUTINES USED:**

**EVALL, R3PASS**

**A,B:** LOWER AND UPPER BOUNDS OF THE INTEGRAL.

**LIMIT:** MAXIMUM NUMBER OF INTERVALS CAN BE USED TO

**EVALUATE THE INTEGRAL.** (29)

**NFUN:** TOTAL NUMBER OF FUNCTIONS TO BE INTEGRATED.

**CSXFM:** (NFUN*LIMIT) ARRAY OF COSINE TRANSFORM COEFFICIENTS.

**NEWINT:** A (NFUN) ARRAY OF INTEGRATED RESULTS FOR NFUN.

**ESTERR:** ARRAY OF ESTIMATED RELATIVE ERRORS OF NFUN FUNCTIONS.

**INDEX:** A (NFUN) ARRAY IS INITIALLY PASSED IN BY COMMON BLOCK.

**IF EACH ELEMENT IS 0 (DO THE INTEGRAL) OR**

**1 (DO NOT CARRY OUT THE INTEGRATION).**

**IF THE INTEGRAL REACHES THE TOLERANCE, SET TO 1.**

**THEN NO MORE CALCULATIONS FOR THIS INTEGRAL.**

**POINTS:** EACH ELEMENT A (NFUN) ARRAY IS THE TOTAL NUMBER

**OF INTERVALS HAVE BEEN USED.**

**NUM:** INITIALLY EQUALS THE NUMBER OF FUNCTIONS NEED

**INTEGRATIONS.**

**DECREASES BY 1 WHEN ONE INTEGRAL HAS REACHED**

**TOLERANCE.
C * IF NUM EQUALS ZERO, ALL INTEGRALS HAVE REACHED
C * THE TOLERANCE.
C * FVALUE: FVALUE(X,CT1) IS A SUBROUTINE TO EVALUATE NFUN
C * DIFFERENT FUNCTIONS WITH ARGUMENT X AND RETURNS
C * A NFUN-DIM VECTOR CT1 AS ALL FUNCTION VALUES.
C
C
ISN 0003
REAL*8 A,B,TOLEBR
REAL*8 SCINT
ISN 0004
REAL*8 PI,RT3,CENTRE,WIDTH,SHIFT,FUN,D,ANGLE,C,S,TMP
ISN 0005
REAL*8 CSXRNM(12,163),ESTIMBR(42),OLDINT(42),
1 NEWINT(42),T1(42),T2(42),T3(42),T4(42),
2 T5(42),T6(42),T7(42),T8(42),
3 T9(42),T10(42),T11(42),T12(42)
ISN 0006
REAL*8 C(T1(42),CT2(42),CT3(42),CT4(42),
1 CT5(42),CT6(42),CT7(42)
ISN 0007
INTEGER L(8),INDEX(42),POINTS(42)
ISN 0008
EQUIVALENCE (CSXRNM(1,1),CT1(1)), (CSXRNM(1,2),CT2(1)),
1 (CSXRNM(1,3),CT3(1)), (CSXRNM(1,4),CT4(1)),
2 (CSXRNM(1,5),CT5(1)), (CSXRNM(1,6),CT6(1)),
3 (CSXRNM(1,7),CT7(1))
ISN 0009
EQUIVALENCE (L(1),L1), (L(2),L2), (L(3),L3), (L(4),L4),
1 (L(5),L5), (L(6),L6), (L(7),L7), (L(8),L8),
2 (J8,JREV)
ISN 0010
COMMON /DATOL/TOLEBR
ISN 0011
COMMON /INTARY,INDEX,POINTS
ISN 0012
COMMON /BEARY,CSXRNM
ISN 0013
COMMON /ESTIMBR/NEWINT,ESTIMBR
ISN 0014
DATA PI,RT3/ 3.141592653589793D0, 1.7320508075680D0 /
C * MAXIMUM INTERVALS ONE CAN DO IN THIS PROGRAM ARE
C * 2*3**(MMAX+1).
C * FOR MMAX=6, THE LIMIT IS 39366.
ISN 0015
DATA MMAX/ 6 /
C * IF TEST THE PROGRAM AND WANT TO CHECK THE INPUT VALUES,
C * SET ITEST=1
C
ISN 0016
ISN 0017
ITEST=0
ISN 0018
IF (ITEST.EQ.0) GO TO 0
ISN 0019
WRITE(3,?I10)A,B,NFUN,LIMIT
ISN 0020
1110 YCMAT(10*F5.1,5X,F5.1,5X,2X5)
ISN 0021
8 CONTINUE
C
C ** INITIALIZATION
C
ISN 0022
CENTRE=(A+B)*.5D0
ISN 0023  C WIDTH=(B-A)*.5D0
ISN 0024  C NFN=0
ISN 0025  DC 5 I=1,WFUN
ISN 0026  IF (INDEX(I).EQ.1) GO TO 5
ISN 0028  NUM=NUM+1
ISN 0029  5 CONTINUE
ISN 0030  C DC 10 J=1,MAX
ISN 0031  L(J)=1
ISN 0032  10 CONTINUE
C ** CCSINE TRANSFORM WITH N=6
C N=6
C * CT1 TO CT7 ARE TEMPORARY VECTORS WITH DIMENSION NFUN
C
ISN 0034  CALL FVALUE(A,CT1)
ISN 0035  CALL FVALUE(B,CT7)
ISN 0036  SHIFT=WIDTH*ET3*.5D0
ISN 0037  TEMP=CENTRE-SHIFT
ISN 0038  CALL FVALUE(TEMP,CT2)
ISN 0039  TEMP=CENTRE-SHIFT
ISN 0040  CALL FVALUE(TEMP,CT6)
ISN 0041  SHIFT= WIDTH*.5D0
ISN 0042  TEMP=CENTRE-SHIFT
ISN 0043  CALL FVALUE(TEMP,CT3)
ISN 0044  TEMP=CENTRE-SHIFT
ISN 0045  CALL FVALUE(TEMP,CT5)
ISN 0046  CALL FVALUE(CENTRE,CT4)
C
ISN 0047  DC 20 I=1,NFUN
ISN 0048  IF (INDEX(I).EQ.1) GC TO 18
ISN 0050  POINTS(I)=W
ISN 0051  GO TO 20
ISN 0052  18 POINTS(I)=0
ISN 0053  20 CONTINUE
C ** EVALUATE THE FACTORED N=6 COSINE TRANSFORM
C ** FOR NFUN FUNCTIONS
C
ISN 0054  DO 30 I=1,NFUN
ISN 0055  IF (INDEX(I).EQ.1) GO TO 30
ISN 0057  T1(I)=CSXFER(I,1)+CSXFER(I,7)
ISN 0058  T2(I)=CSXFER(I,1)-CSXFER(I,7)
ISN 0059  T3(I)=2.D0*CSXFER(I,4)
ISN 0060  T4(I)=CSXFER(I,2)+CSXFER(I,6)
ISN 0061  T5(I)=(CSXFER(I,2)-CSXFER(I,6))*ET3
ISN 0062  T6(I)=CSXFER(I,3)+CSXFER(I,5)
ISN 0063  T7(I)=CSXFER(I,3)-CSXFER(I,5)
ISN 0064  T8(I)=T1(I)+2.D0*T6(I)
ISN 0065  T9(I)=2.D0*T4(I)+T3(I)
ISN 0066  T10(I)=T2(I)+T7(I)
ISN 0067  T11(I)=T1(I)-T6(I)
ISN 0068  T12(I)=T4(I)-T3(I)
CSXFRM(I,1) = T6(I) + T9(I)
CSXFRM(I,2) = T10(I) + T5(I)
CSXFRM(I,3) = T11(I) + T12(I)
CSXFRM(I,4) = T2(I) - 2*D0*T17(I)
CSXFRM(I,5) = T11(I) - T12(I)
CSXFRM(I,6) = T10(I) - T5(I)
CSXFRM(I,7) = T6(I) - T9(I)

30 CONTINUE

C * NUMBER OF POINTS HAS BEEN USED.

C
MUSED = 7

C ** GO TO INTEGRAL COMPUTATION

C
GO TO 210

C

C * COMPUTE REFINED APPROXIMATION
C * EVALUATE FUNCTIONS FOR N=36,54,...,2*3**9.
C * SAMPLE FUNCTION AT INTERMEDIATE POINTS IN DIGIT
C * REVERSED ORDER.
C * AS THE SEQUENCE IS GENERATED, COMPUTE THE FIRST
C * (RADIX FOUR TRANSFORM) PASS OF THE FAST FOURIER
C * TRANSFORM (FFT).
C

100 DO 110 J = 2, MMAX
L(J) = L(J)
110 CONTINUE

C

L(MMAX) = 3*L(MMAX-1)
J = MUSED
FUND = PI/DFLOAT(3*M)
DC 128 J1 = 1, L1, 1
DO 127 J2 = J1, L2, L1
DO 126 J3 = J2, L3, L2
DO 125 J4 = J3, L4, L3
DO 124 J5 = J4, L5, L4

DO 123 J6 = J5, 16, L5
DO 122 J7 = J6, L7, L6
DO 121 J8 = J7, L8, L7

DO 120 J9 = J8, 11, L8

DO 120 I = 1, PFUN
IF (INDEX(I).EQ.1) GO TO 120
T5(I) = T1(I) + T3(I)
T6(I) = T2(I) + T4(I)
CSXFRM(I, J+1) = T5(I) + T5(I)
```plaintext
ISN 0110 120  
ISN 0111  
ISN 0112  
ISN 0113  
CONTINUE  
J=J+4  
ISN 0114  
CONTINUE  
CONTINUE  
ISN 0115  
CONTINUE  
ISN 0116  
CONTINUE  
ISN 0117  
CONTINUE  
ISN 0118  
CONTINUE  
ISN 0119  
CONTINUE  
CONTINUE  
ISN 0120  
CONTINUE  
CONTINUE  
ISN 0121  
CONTINUE  
CONTINUE  
ISN 0122  
CONTINUE  
C  
** DO BAXIDX3 PASSES OF FAST FOURIER TRANSFORM  
C  
ISN 0123  
W2=2*W  
ISN 0124  
#STEP=4  
ISN 0125  
J1=NUSED+NSTEP  
ISN 0126  
J2=J1+NSTEP  
ISN 0127  
CALL B3PASS (N2,#STEP,NFUN,NUSED,J1,J2)  
ISN 0128  
NSTEP=3*NSTEP  
ISN 0129  
IF (#STEP .LT. W) GO TO 150  
C  
** COMBINE RESULTS  
C  
** FIRST DO J=0 AND J=W  
C  
ISN 0130  
DO 40 I=1,NFUN  
IF (INDEX(I) .EQ. 1) GO TO 40  
ISN 0131  
T1(I)=CSXFRM(I,J1)  
ISN 0132  
T2(I)=CSXFRM(I,J1)  
ISN 0133  
CSXFRM(I,J1)=T1(I)+T2(I)  
ISN 0134  
CSXFRM(I,J1)=T1(I)+T2(I)  
ISN 0135  
T1(I)=CSXFRM(I,J1)  
ISN 0136  
T2(I)=CSXFRM(I,J1)  
ISN 0137  
CSXFRM(I,J2)+T1(I)+T2(I)  
ISN 0138  
CSXFRM(I,J2)+T1(I)+T2(I)  
ISN 0139  
CSXFRM(I,J1)+T1(I)+T2(I)  
ISN 0140  
CSXFRM(I,J2)+T1(I)+T2(I)  
ISN 0141  
CSXFRM(I,J1)+T1(I)+T2(I)  
ISN 0142  
40 CONTINUE  
C  
** NOW DO REMAINING VALUES OF J  
C  
ISN 0143  
N3=3*N  
ISN 0144  
NLESS1=N-1  
ISN 0145  
DO 180 J=1,NLESS1  
ISN 0146  
J3=J+N  
ISN 0147  
J2=N3-J  
ISN 0148  
ANGLE=FUND+FLOAT(J)  
ISN 0149  
C=DCOS(ANGLE)  
ISN 0150  
S=DSIN(ANGLE)  
ISN 0151  
DO 185 I=1,NFUN  
ISN 0152  
IF (INDEX(I) .EQ. 1) GO TO 185  
ISN 0153  
T1(I)=CSXFRM(I,J1+2)+C*CSXFRM(I,J1+2)  
ISN 0154  
T2(I)=S*CSXFRM(I,J1+2)+C*CSXFRM(I,J1+2)  
ISN 0155  
CSXFRM(I,J1+2)=T1(I)+T2(I)  
ISN 0156  
CSXFRM(I,J1+2)=T1(I)+T2(I)  
ISN 0157  
CSXFRM(I,J1+2)=CSXFRM(I,J1+2)+T1(I)+T2(I)  
```

CSXFRM(I,J+1) = CSXFRM(I,J+1) + 2.0D0*TI(I)
CONTINUE

C ** NOW UNSCRAMBLE THE DIGIT REVERSED ORDER TO GET
C ** THE CORRECT DIGIT ORDER.
C ** SPECIAL FEATURE OF FFT.

DO 50 I = 1, N
IF (INDEX(I) .EQ. 1) GO TO 50
50 T1(I) = CSXFRM(I, N2+1)
T2(I) = CSXFRM(I, N2+2)
CONTINUE

DO 190 J = 1, NLESS1
190 J = J + 1
CONTINUE

IF (INDEX(I) .EQ. 1) GO TO 195
CSXFRM(I, J2) = CSXFRM(I, J1)
CSXFRM(I, J1) = CSXFRM(I, J2+2)
CONTINUE

CONTINUE

DC 60 I = 1, N
IF (INDEX(I) .EQ. 1) GO TO 60
CSXFRM(I, N3) = T1(I)
CSXFRM(I, N3+1) = T2(I)
CONTINUE

N = N3
NUSED = N + 1

C ** INTEGRAL EVALUATION
C ** EVALUATE NEW ESTIMATE OF INTEGRAL

NEWINT(I) = NEWINT(I) + 5.0D0*CSXFRM(I, 1)
CONTINUE

DC 220 J = 1, NLESS3
JREV = J
DO 225 I = 1, N
IF (INDEX(I) .EQ. 1) GC TO 225
225 NEWINT(I) = NEWINT(I) + CSXFRM(I, JREV) / DFLOAT(JREV*(2-JREV))
CONTINUE

CONTINUE

IF (N .EQ. 6) GO TO 230
C ** T Esti mated E rror A dequate for n >= 16. C

DC 80 I=1,NFUN
ISN 0203 IF (INDEX(I) = EQ. 1) GO TO 80
ISN 0205 ESTERR(I) = DABS(GOLDINT(I) + 3.0 - NEWINT(I))
ISN 0206 POINTS(I) = n

C * IF TESTING, PUT THE FOLLOWING STATEMENTS IN. C
C SCLINT = WIDTH * NEWINT(I) / DFLOAT(N/2)
C WRITE (3,930) I, POINTS(I), SCLINT, ESTERR(I)
C 930 FORMAT (3H 1=15,3H N=15,2H INTEGRAL ESTIMATED AS ,
C 1 D15.8,7H ERROR ,D15.8)

ISN 0207 IF (DABS(NEWINT(I) * TOLEB) .LT. ESTERR(I)) GC TO 75
ISN 0209 NUM = NUM - 1
ISN 0210 INDEX(I) = 1
ISN 0211 IF (NEWINT(I)) 77, 80, 77
ISN 0212 77 ESTERR(I) = ESTERR(I) / DABS(NEWINT(I))
ISN 0213 80 CONTINUE

C * All integrations reach the required tolerance.

ISN 0214 IF (NUM = EQ. 0) GC TO 400
ISN 0216 DO 230 I = 1, NFUN
ISN 0217 IF (INDEX(I) = EQ. 1) GO TO 90
ISN 0219 OLDINT(I) = NEWINT(I)
ISN 0220 90 CONTINUE

C ** IF estimated error too large, refine sampling if permitted C
C * I.E. M (NUMBER OF INTERVALS USED) < LIMIT .

ISN 0221 IF (3 * M + 1 .LE. LIMIT) GC TO 100
ISN 0223 400 DC 300 I=1,NFUN
ISN 0224 IF (POINTS(I) = EQ. 0) GO TO 300
ISN 0226 NEWINT(I) = WIDTH * NEWINT(I) / DFLOAT(POINTS(I)/2)
C WRITE (3,920) POINTS(I), NEWINT(I), ESTERR(I)
C 920 FORMAT (I15,5X,D15.8,5X,D15.8)
ISN 0227 300 CONTINUE

C C 910 F C M AT (*4.,50X,25 B REFINEMENT NOT PERMITTED)
C 920 F C M AT (/15,5X,D15.8,5X,D15.8)

ISN 0228 RETURN
ISN 0229 END
SUBROUTINE H3PASS (N2,N,NFUN,L0,L1,L2)

* RADIX 3 PASS FOR FFT OF REAL SEQUENCE OF LENGTH N2 *

* REFERENCES:
* GENTLEMAN, W.M., COMM. ACM 15 (MAY 1972) 343-346
* GENTLEMAN, W.M. AND SANCHE, G., PROC. AFIPS 1966
* FJCC, VOL. 29, SPARTAN BOOKS, N.Y., 563-578

C REAL*8 C(42,163)
C REAL*8 TWOPI,HAPBT3,RSUM,EDIFF,RSUM2,ISUM,DIFF,DIFF2
C REAL*8 FUND,ANGLE,C1,S1,C2,S2,R0,M1,R2,10,11,12
C INTEGER HAPDM,H3,K,K0,K1,J,J0,J1,N2,N
C C CHANGE THE ARGUMENT OF IND ARRAY TO NFUN
C C INTEGER IND(42),POINIS(42)
C DATA TWOPI, HAPBT3 / 6.283185307179586D0, -866025403784D0 /
C COMMON /INTARY/IND,POINTS
C COMMON /HEARY/C
C C HALPM=(N-1)/2
C M3=M*3
C FUND=TWOPI/DFLOAT(M3)
C ** TWINND FACTOR UNITY

DO 10 I=1,N2,N3

10 CONTINUE

DO 20 I=1,NFUN

IF (IND(I).EQ.1) GO TO 10

RSUM=(C(I,1+K)+C(I,1+K)) * HAPBT3
C(I,1+K)=C(I,1+K)-RSUM
C(I,1+K)=C(I,1+K)+RSUM

DO 20 I=1,NFUN

IF (IND(I).EQ.1) GO TO 20

EDIFF=(C(I,1+K)-C(I,1+K))
C(I,1+K)=C(I,1+K)-EDIFF
C(I,1+K)=C(I,1+K)+EDIFF

20 CONTINUE
DO 40 J=1,HALFM
J=J+1
C ** COMPUTE THE TWIDDLE FACTOR
ANGLE=PUD*DEGAT(J)
C1=DCOS(ANGLE)
S1=DSIN(ANGLE)
C2=C1*C1*S1*S1
C ** CHOOSE THE REPLICATION
DO 30 KO=J0,K2,K3
K1=KO-J0+J
C ** OBTAIN TWIDELLED VALUES
DO 30 I=1,NPON
IP (IND(I),EQ.1) GO TO 30
B0=C(I,L0+KO)
I0=C(I,L0+KO)
B1=C1*C(I,L1+KO)-S1*C(I,L1+KO)
I1=S1*C(I,L1+KO)+C1*C(I,L1+KO)
B2=C2*C(I,L2+KO)-S2*C(I,L2+KO)
I2=S2*C(I,L2+KO)+C2*C(I,L2+KO)
C ** COMPUTE THE TRANSFORM AND RETURN IN PLACE
BSUM=B1+B2
BDIFF=(I2-I1)*HAPRT3
BSUM2=BSUM-5D0*BSUM
ISUM=I1+I2
IDIFF=(I1-I2)*HAPRT3
IDIFF2=I0-.5D0*TSUM
C(I,L0+KO)=BSUM
C(I,L0+KO)=BSUM2+IDIFF
C(I,L1+KO)=BSUM2-IDIFF
C(I,L1+KO)=IDIFF4
C(I,L2+KO)=IDIFF4
C(I,L2+KO)=IDIFF4
C(I,L2+KO)=I0+ISUM
CONTINUE
CONTINUE
RETURN
IND
**LEVEL 21.8 (JUN 74)**

**OS/360 FORTRAN H**

**COMPILER OPTIONS** - NAME= MAIN, OPT=00, LIMEN=50, SIZE=0000K,
SOURCE, EBCICL, MOLIST, DECK, LOAD, MAP, ROEDIT, ID, XREF

**ISN 0002**

SUBROUTINE FVALUE(C,F)

* This program evaluates the function for the angular integrals. C corresponds to the variable \( \xi \) (i.e., \( \cos \theta \))
* in the formula. \( B_0 \) is constant in this program.
* \( R \) is the internuclear distance, an input constant through the whole program.
* There are 27 complex angular integrands corresponding to the different argument combinations.

* \( \xi = \xi' \)

**ISN 0003**

REAL*8 Y0,Y1,Y1C,Y1S,Y
1 EDA1,EDA2,EDA3,R,LAMDA,RO,ROVE2,ROVB,
2 C,S,S2,F(42),FF(16),TEXP(2),
3 BESJ0,BESJ1,BESJ2,BESEL0,BESEL1

**ISN 0004**

INTEGER WAVG(16,2),FLAG(114),NDIFBO(42,2),
1 INDEX(42),POINTS(42)

**ISN 0005**

COMMON /CONS1/EDA1,EDA2,EDA3
**ISN 0006**

COMMON /CONS2/R,LAMDA
**ISN 0007**

COMMON /INTER/INDEX,POINTS
**ISN 0008**

COMMON /INNT2/FLAG,NDIFBO,NGANG
**ISN 0009**

COMMON /VARSBL/BO

**ISN 0011**

NFUN2=NFUN/2
**ISN 0012**

ROVE2=RO*ROVB*RO
**ISN 0013**

ROVB=RO*RO*2.0
**ISN 0014**

Y=DSCRT((ROVB2-ROVB*C)
**ISN 0015**

IF (EDA1.LT.EDA2) Y=DSCRT((ROVB2+ROVB*C)
**ISN 0017**

S2=Y-C*C
**ISN 0018**

S=DSCRT(S2)
**ISN 0019**

Y0=BO*Y*LAMDA
**ISN 0020**

Y1=EDA3*Y0*C
**ISN 0021**

Y1C=DCOS(Y1)
**ISN 0022**

Y1S=DSIN(Y1)

* CHECK TO SEE IF BESSEL FUNCTION NEEDS TO BE CALLED.

**ISN 0023**

DC 5 I=18,21
**ISN 0024**

IF (INDEX(I)-EQ.1)GO TO 3
**ISN 0026**

BESJ1=BESZ1(Y0)
**ISN 0027**

BESJ0=BESZ0(Y0)
**ISN 0028**

GO TO 7

**ISN 0029**

3 II=I+NFUN2
**ISN 0030**

IF (INDEX(I)-EQ.1)GO TO 5
**ISN 0032**

BESJ1=BESZ1(Y0)
**ISN 0033**

BESJ0=BESZ0(Y0)
**ISN 0034**

GO TO 7
ISN 0035    5    CONTINUE
ISN 0036    GC TO 30
ISN 0037    7    IF (IO) 10, 20, 10
ISN 0038   10    BESJ2=2.DO*BESJ1/YO-BESJ0
ISN 0039    C
ISN 0040   20    BESJ2=0.DO
ISN 0041    GO TO 90
ISN 0042   30    BESJ0=0.DO
ISN 0043    BESJ2=0.DO
ISN 0044    DO 50 I=1,10
ISN 0045    IF (INDEX(I)-EQ.1)GO TO 40
ISN 0047    BESJ0=BESJ0(Y0)
ISN 0048    GO TO 60
ISN 0049   40    II=I+NPUN2
ISN 0050    IF (INDEX(II)-EQ.1)GO TO 50
ISN 0052    BESJ0=BESJ0(Y0)
ISN 0053    GO TO 60
ISN 0054   50    CONTINUE
ISN 0055    C
ISN 0056   60    BESJ1=0.DO
ISN 0058    DC 80 I=11,17
ISN 0057    IF (INDEX(I)-EQ.1)GO TO 70
ISN 0058    BESJ1=BESJ1(Y0)
ISN 0060    GO TO 90
ISN 0061   70    II=I+NPUN2
ISN 0062    IF (INDEX(II)-EQ.1)GO TO 80
ISN 0064    BESJ1=BESJ1(Y0)
ISN 0065    GO TO 90
ISN 0066   80    CCCONTINUE
ISN 0067   90    CONTINUE
ISN 0068    C
ISN 0069    YEXP(2)=YEXP(-EDA2*Y)
C * YEXP(1)=YEXP(-EDA2*Y)
ISN 0070    C
ISN 0071    YEXP(2)=YEXP(-EDA2*Y)
ISN 0072    YEXP(1)=YEXP(2)**2
ISN 0073    C
ISN 0074    INITIAL CALCULATION FOR NECESSARY FUNCTION EVALUATIONS
C
ISN 0075    FF(1)=BESJ0
ISN 0076    FF(2)=Y*BESJ0
ISN 0077    FF(3)=C*BESJ0
ISN 0078    FF(4)=FF(2)*C
ISN 0079    FF(5)=FF(3)*C
ISN 0080    FF(6)=FF(5)*Y
ISN 0081    FF(7)=FF(5)*C
C
ISN 0084    FF(8)=S*BESJ1
ISN 0085    FF(9)=FF(8)*C
ISN 0086    FF(10)=Y*FF(8)
ISN 0087    FF(11)=FF(10)*C
ISN 0081  C  FF(12) = FF(9) * C
ISN 0082  C  FF(13) = S2*(BESJ0 - BESJ2)
ISN 0083  C  FF(14) = FF(13) * C
ISN 0084  C  FF(15) = S2*(BESJ0 + BESJ2)
ISN 0085  C  FF(16) = C*FF(15)

C REAL AND IMAGINARY PARTS OF FUNCTIONS LISTED IN TABLE 9-1

ISN 0086  C  K=0
ISN 0087  EC 200  I=1,16
ISN 0088  DO 100  J=1,2
ISN 0089  L1=NAVG(I,J)
ISN 0090  IF (L1.EQ.0) GO TO 100
ISN 0092  K=K+1
ISN 0093  F(K)=FF(I) * YEXP(J)
ISN 0094  KK=K+MPUN2
ISN 0095  F(KK)=F(K) * Y1S
ISN 0096  F(K)=F(K) * Y1C
ISN 0097  100 CONTINUE
ISN 0098  200 CONTINUE
ISN 0099  C  RETURN
ISN 0100  C  END
COMPILER OPTIONS - NAME= MAIN,OPT=02,LINECNT=58,SIZE=0000X.
SOURCES,EDIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,XXEP
ISN 0002
C
C * EVALUATE BESSEL FUNCTION J1 TO A HIGHER ACCURACY.
C * USE CHEBYSHEV SERIES EXPANSION.
C * REFERENCES:
C * NATIONAL PHYSICAL LABORATORY MATHEMATICAL TABLES,
C * VOL. 5, CHEBYSHEV SERIES FOR MATHEMATICAL FUNCTIONS, BY C. W. CLENSHAW
C * (1962)

ISN 0003
C
C INTEGER SIGN

ISN 0004
C REAL*8 AS(15), AP(10), AQ(10)

ISN 0005
C REAL*8, B, E, Q, P, PI2, PI4, B1, P1, Q1, Q, Q1, XI

ISN 0006
C DATA PI4/2.356154490152345D0/

ISN 0007
C DATA AS(15), AS(14), AS(13), AS(12), AS(11), AS(10), AS(9),
1 AS(8), AS(7), AS(6), AS(5), AS(4), AS(3), AS(2), AS(1),
2 1/1.296717541210530000D0, 1.918011605412717000D0,
3 1.287994098857678000D0, 0.661443934134543000D0,
4 1.777091172397280000D0, 0.029175524606154000D0,
5 0.003240270182684000D0, 0.000260443893486000D0,
6 1.45870214240000D0, 0.376175870000D0, 0.294970700000D0,
7 0.942421D-9, 0.25281D-10, 0.576D-12, 0.11D-13/

ISN 0008
C DATA AP(10), AP(9), AP(8), AP(7), AP(6), AP(5), AP(4), AP(3),
1 AP(2), AP(1)/2.001906081720027000D0, 0.0009899883308600D0,
2 0.398728430000D0, 0.617763400000D0, 0.187189100000D0,
3 0.88160D-10, 0.5705D-11, 0.470D-12, 0.5D-14/

ISN 0009
C DATA AQ(10), AQ(9), AQ(8), AQ(7), AQ(6), AQ(5), AQ(4), AQ(3),
1 AQ(2), AQ(1)/0.0355555439071000D0, 0.362772354932D-4,
2 0.913861526D-6, 0.20957818D-7, 0.822919D-9, 0.46864D-10,
3 0.3515D-11, 0.326D-12, 0.36D-13, 0.4D-14/

ISN 0010
C X=XX

ISN 0011
C SIGN=0

ISN 0012
C IF (X) 20, 30, 40

ISN 0013
C * FLAG IS SET FOR NEGATIVE X

ISN 0014
C 20 SIGN=1

ISN 0015
C X=DABS(X)

ISN 0016
C 40 IF (X=8.0D0) 50, 60, 70

ISN 0017
C IF (X<0) 60

ISN 0018
C 50 Y=X*X/16.-D0.-2.-D0

ISN 0019
C B1=0.0D0

ISN 0020
C DC 80 I=1,15

ISN 0021
C B2=B1

ISN 0022
C B=8*B1+B2+AS(I)

ISN 0023
C 80 CONTINUE

ISN 0024
C BESSEL=I*X*(B-B2)/16.-D0

ISN 0025
C GO TO 100
C FCB X>8

0026 XPI2=.79788456080286580

0027 XPI2=XPI2/DSQRT(I)

0028 E=X-P14

0029 X=6.D0/X

0030 Y=4.00*X*X-2.00

0031 P=0.00

0032 P=0.00

0033 CI=0.00

0034 Q=0.00

0035 DC 90 I=1,10

0036 P2=P1

0037 P1=P

0038 C2=Q1

0039 Q1=Q

0040 E=Y*P1-P2+AP (I)

0041 Q=Y*Q1-Q2+AQ (I)

0042 50 CONTINUE

0043 BESEL1=XPI2* ((E-P2) *DCCS (E) -I* (Q-Q2) *DSIN (E)) /2.00

0044 GO TO 100

C X=0

0045 30 BESEL1=0.D0

0046 GC TO 100

C X=8

0047 60 BESEL1=.234636346853915.D0

C OUTPUT

0048 IF (SIGN.EQ.1) BESEL1=-BESEL1

C RETURN

0050 END
IWEL 21.8 (Jun '74) CS/360 FORTRAN H

COMPILE OPTIONS - NAME=MAIN,OPT=02,LINSEQ=58,SIZE=00000F,
SOURCE,ECDC,MOLIST,DECK,LOAD,MAP,MODIT,IBM

C

ISW 0002
C

DC Double Precision Function BESLO(X)

C

* EVALUATE BESSEL FUNCTION J0 TO A HIGHER ACCURACY.
C

* USE CHEBYSHEV SERIES EXPANSION.
C

* REFERENCE: NATIONAL PHYSICAL LABORATORY MATHEMATICAL
C

* TABLES. VOL. 5, CHEBYSHEV SERIES FOR MATHEMATICAL
C

* FUNCTIONS, BY C. W. CLEMSHAW. (1962)
C

ISW 0003
REAL*8 AS(15),AQ(10)
ISW 0004
REAL*8 X,1,B2,E,C2,P2,PI2,PI4,B1,P1,Q1,XX
ISW 0005
C

DATA PI4/-78539615333746800/

ISW 0006
C

DATA AS(15),AS(14),AS(13),AS(12),AS(11),AS(10),AS(9),
1 AS(8),AS(7),AS(6),AS(5),AS(4),AS(3),AS(2),AS(1),
2 .315455942949780D0,-.0087334423528525D0,
3 .370094953672650D0,
4 .156067102332097D0,-.034693769411409D0,
5 .0048191800696486D0,-.000460626166206D0,
6 .000032460328821D0,-.1761946908D0,-
7 .76081636D7,-.2679254D-6,.78487D-10,
8 -.1944D-11,.41D-13/

ISW 0007
C

DATA AP(10),AP(9),AP(8),AP(7),AP(6),AP(5),AP(4),AP(3),
1 AP(2),AP(1)/1.99892698695037D0,-.536522046813D-3,
2 .3075184786D-5,-.51705945D-7,-.1630646D-8,
3 -.78641D-10,.5168D-11,.430D-12,.43D-13,.5D-14/

ISW 0008
C

DATA AQ(10),AQ(9),AQ(8),AQ(7),AQ(6),AQ(5),AQ(4),AQ(3),
1 AQ(2),AQ(1)/-.031111709210674D0,.683451994264D-4,
2 -.741449841D-6,.17972457D-7,.727192D-9,-.42201D-10,
3 -.3207D-11,.301D-12,.33D-13,.5D-14/

ISW 0009
C

X=XX

ISW 0010
IF (X) 10,20,30
ISW 0011
10 X=DABS(X)
ISW 0012
30 IF (X-61D0) 40,50,60
C

* FOR I<8

ISW 0013
40 X=X*16.D0-2.D0
ISW 0014
B=0.D0
ISW 0015
B=0.D0
ISW 0016
DC 70 J=1,15
ISW 0017
B2=B1
ISW 0018
B1=B
ISW 0019
B=Y*B1-B2*A5(I)
ISW 0020
70 CONTINUE
ISW 0021
BESLO=(B-B2)/2.D0
ISW 0022
GC TO 90
C

* FOR I>8
ISN 0023  60  XPI2=-7.9788456802865D0
ISN 0024  XPI2=XPI2/DSQRT(1)
ISN 0025  I=X-PX4
ISN 0026  X=8.D0/X
ISN 0027  Y=4.D0*X*X-2.D0
ISN 0028  P1=0.D0
ISN 0029  P=0.D0
ISN 0030  C1=0.D0
ISN 0031  C=0.D0
ISN 0032  DO 80 I=1,10
ISN 0033    P2=P1
ISN 0034    P1=P
ISN 0035    Q2=Q1
ISN 0036    Q1=Q
ISN 0037    R=X*P1-P2+AP(I)
ISN 0038    C=Y*Q1-Q2+AQ(I)
ISN 0039  80  CONTINUE
ISN 0040  BESEL0=XPI2*{(P-P2)*DCOS(E)-X*(Q-Q2)*DSIN(E)}/2.D0
ISN 0041  GC TO 90
   C  C  *  I=0
ISN 0042  20  BESEL0=1.D0
ISN 0043  GO TO 90
   C  C  *  I=8
ISN 0044  50  BESEL0=-17165087137554D0
   C
ISN 0045  90  RETURN
ISN 0046  END
COMPILE OPTIONS - NAME= MAIN,OPT=02,LINKCNT=58,SIZE=8000K,
SOURCE,EXTEND,NOLIST,DECK,LOAD,MAP,NOEDIT,IL,IREP

PROGRAM NAME: TRANRAY

PURPOSE---
COMBINE ALL THE DOUBLE INTEGRATIONS INTO FOUR 5x5 MATRICES.
FROM THESE MATRICES, THE INTERCHANGE TRANSITION MATRIX CAN BE
CONSTRUCTED.

FOR SOME OTHER MATRICES WITH DIFFERENT DIMENSIONS,
ONE NEEDS A DIFFERENT PROGRAM.

INPUT--
FILE: UNIT NUMBER WHERE THE INPUT CAN BE READ IN.
NCUT: UNIT NUMBER WHERE THE OUTPUT CAN BE WRITTEN ON.
NFUNO: TOTAL NUMBER OF DIFFERENT DOUBLE INTEGRALS.
ITEST: =0, IF TESTING THE PROGRAM, PRINT OUT THE RESULTS.
       =1, OTHERWISE DO NOT PRINT OUT THE RESULTS.
       2(1): Z(1)=Z1 THE PROJECTILE NUCLEAR CHARGE.
       2(2): Z(2)=Z2 THE TARGET NUCLEAR CHARGE.
       M2(1): M2(1)=MASS OF THE PROJECTILE.
       M2(2): M2(2)=MASS OF THE TARGET.
       MRATIO: RATIO OF PROTON MASS TO ELECTRON MASS.
       EL: INCIDENT ENERGY IN KEV PER AVG IN LAB FRAME.
       V1: INCIDENT RELATIVE VELOCITY IN A.U. OF THE
       PROJECTILE.
       EA: IMPACT PARAMETER IN A.U.
       SCANGL: SCATTERING ANGLE IN DEGREE.
       R: INTERNUCLEAR DISTANCE OF THE PROJECTILE AND
       TARGET ATOM IN A.U.
       VDOT: RADIAL VELOCITY OF THE RELATIVE MOTION OF
       TWO NUCLEI.
       LAMDA: =V1*EA/R.
       SCALE: THE TANGENTIAL VELOCITY AT THE
       CLOSEST APPROACH POINT.
       RNL: A 3-DIMENSIONAL (2,2,NFUNO) ARRAY.
       X=1, AND 2 POINT TO REAL AND IMAGINARY PARTS.
       J=1, AND 2 POINT TO (A,B) AND (B,A) COMBINATIONS.
       R=NFUNO, DIFFERENT FUNCTION VALUES.

OUTPUT--
ECOUP,DCOUP,OVLP,ACOUP.
FOUR 3-DIMENSIONAL (2,2,25) ARRAY FOR RADIAL COUPLING,
DIPOLE COUPLING, OVERLAPPING AND ANGULAR (ROTATIONAL)
COUPLING.

ETN1
ONE 3-DIMENSIONAL (2,2,25) ARRAY FOR INTERCHANGE
TRANSITION MATRIX.

SUGGESTIONS AND HINTS --
THE GENERAL ROW AND COLUMN DEFINITION OF (2,2,25) ARRAYS IS

1 2
THE 25 ELEMENTS WHICH CORRESPOND TO (5X5) MATRICES ARE
ARRANGED AS IN THE FOLLOWING

<table>
<thead>
<tr>
<th>REAL</th>
<th>IMAGINARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A, B)</td>
<td>(B, A)</td>
</tr>
</tbody>
</table>

1S  2S  2PZ  2PX  2FY

ISW 0002

C C
C
C REAL*6 MRATIO,EL,V1,BA,SCANGL,E,RDOT,LAMEA,B2,
1 Z12,Z23,Z202,C1,C2,C3,C4,C5,C6,C7,C8,
2 DBL(2,2,57),ACOUP(2,2,25),DCOUP(2,2,25),
3 ER,OVLP(2,2,25),ACOUP(2,2,25),
4 FASREL,FASING,FINK(2,2,25),WT(2)
C
ISW 0003

C
C INTEGER Z(2),NZ(2),ST,S2
C
ISW 0004

C BIAD(1,1000) NFIL,DNDT,UFUNO,ITEST
ISW 0005

C NFUNO2=NFUNO/2
ISW 0006

C NDATA=0
C
ISW 0007

S C CONTINUE
ISW 0008

C DC 10 J=1,2
ISW 0009

C READ (NFIL,DNDT,END=2000) Z(1),Z(2),NZ(1),NZ(2)
ISW 0010

C READ (NFIL,1020) MRATIO,EL,V1,BA,SCANGL,E,RDOT,LAMEA,
1 ((DBL(I,J,K),K=1,NFUNO2),I=1,2)
ISW 0011

10 C CONTINUE
ISW 0012

C NDATA=NDATA+1
C
ISW 0013

C Z12=DFLOAT(Z(1)*Z(2))
ISW 0014

C Z32=Z12*DQRT(Z(2))
ISW 0015

C Z42=2*DQRT(Z12)
ISW 0016

C C1=2*C3
ISW 0017

C C2=3*C2/TWO
ISW 0018

C C3=Z32/16.DO
ISW 0019

C C5=Z12*C3
ISW 0020

C C4=C5/2.DO
C
ISW 0021

C INITIALIZATION
ISW 0022

C DO 16 J=1,2
ISW 0023

C DO 14 K=1,25
ISW 0024

C BCOUP(I,J,K)=0.DO
ISW 0025

C DCOUP(I,J,K)=0.DO
ISW 0026

C OVLAP(I,J,K)=0.DO
ISN 0027
ACOUP(I,J,K) = 0.00
ISN 0028 14 CONTINUE
ISN 0029 16 CONTINUE
ISN 0030 16 CONTINUE

THE COMBINATIONS ARE:
I=1, II=2 OR I=2, II=1
J=1, JJK=2 OR J=2, JJK=1
SO WHEN I INDICATES THE REAL PARTS (I.E. I=1), II POINTS TO THE IMAGINARY PARTS (I.E. II=2), AND VICE VERSA.
SIMILARLY WHEN J POINTS TO A OR (A,B), JJ POINTS TO B OR (B,A).

THESE INTERCHANGE PROPERTIES ARE DUE TO THE SYMMETRY OF EXCHANGE A AND B.

THE FOLLOWING EQUATIONS ARE LISTED IN TABLE B-1.

II=2
DO 50 I=1,2
   DO 40 J=1,2
      IF (J-1)22,20,22
      JJ=2
      GO TO 24
   ISN 0035 20
   ISN 0036
   ISN 0037 22
   ISN 0038 24
   CONTINUE
   ISN 0039
   ISN 0040
   ISN 0041
   S=1
   S1=-1
   IF (I.EQ.1.AND.J.EQ.1) S1=-1
   IF (I.EQ.2.AND.J.EQ.2) S2=-1
   ISN 0042
   ISN 0043
   ISN 0044
   C RADIAL COUPLING
   ISN 0045
   ISN 0046
   BCOUP(I,J,1) = 2*C1*DBL(I,J,4) -
   BCOUP(I,J,2) = 2*C2*(2*DBL(I,J,9) - Z(JJ)*DBL(I,J,14))
   BCOUP(I,J,3) = 2*C2*(JJ)*DBL(I,J,23) - 8*DBL(I,J,9))
   BCOUP(I,J,4) = S1*C2*Z(JJ)*DBL(I,J,41)
   ISN 0047
   ISN 0048
   ISN 0049
   C BCOUP(I,J,6) = 2*C2*(2*DBL(I,J,1) - Z(J)*DBL(I,J,2))
   BCOUP(I,J,7) = S2*C3*(2*DBL(I,J,6) - Z(J)*DBL(I,J,7) -
   2*(JJ)*DBL(I,J,11)) + 2*DBL(I,J,12))
   BCOUP(I,J,8) = 2*C3*Z(JJ)*DBL(I,J,20) - 8*DBL(I,J,6))
   BCOUP(I,J,9) = C3*Z(JJ) - DBL(I,J,21))
   ISN 0050
   ISN 0051
   ISN 0052
   ISN 0053
   C BCOUP(I,J,11) = 2*C2*Z(J) - DBL(I,J,15)
   BCOUP(I,J,12) = 2*C3*Z(J) + 2*DBL(I,J,20) - Z(J)*DBL(I,J,15))
   BCOUP(I,J,13) = S2*C5*DBL(I,J,30) - 8*DBL(I,J,20))
   BCOUP(I,J,14) = S1*C5*DBL(I,J,44)
   ISN 0054
   ISN 0055
   ISN 0056
   ISN 0057
   C BCOUP(I,J,16) = S1*C2*Z(J) - DBL(I,J,36)
   BCOUP(I,J,17) = S1*C3*Z(J) - 2*DBL(I,J,38))
   BCOUP(I,J,18) = S1*C5*DBL(I,J,48))
   ISN 0058
   ISN 0059
   ISN 0060
   ISN 0061
   ISN 0062
   C DIPOL coupling
ISN 0063
DCOUP(I,J,1)=S2*C1*DBL(I,JJ,19)
ISN 0064
DCOUP(I,J,2)=S2*C2*(2*DBL(I,JJ,17)-Z(JJ)*DBL(I,JJ,18))
ISN 0065
DCOUP(I,J,3)=S2*C2*Z(JJ)*DBL(I,JJ,29)
ISN 0066
BCOUP(I,J,4)=S1*C2*Z(JJ)*DBL(I,JJ,43)

ISN 0067
DCOUP(I,J,6)=S2*C2*(2*DBL(I,JJ,24)-Z(JJ)*DBL(I,JJ,28))
ISN 0068
DCOUP(I,J,7)=C3*(2*DBL(I,JJ,21)-Z(JJ)*DBL(I,JJ,26))
ISN 0069
-2*(J)*DBL(I,JJ,22)+Z(JJ)*DBL(I,JJ,27)]*S2
ISN 0070
DCOUP(I,J,8)=C3*Z(JJ)*(2*DBL(I,JJ,31)-Z(JJ)*DBL(I,JJ,34))
ISN 0071
DCOUP(I,J,9)=C3*Z(JJ)*(2*DBL(I,JJ,45)-Z(JJ)*DBL(I,JJ,50))

ISN 0072
DCOUP(I,J,11)=S2*C2*Z(JJ)*(2*DBL(I,JJ,24)+DBL(I,JJ,33))
ISN 0073
DCOUP(I,J,12)=C3*Z(JJ)*(2*DBL(I,JJ,24)+DBL(I,JJ,33))
ISN 0074
-2*(J)*DBL(I,JJ,22)+DBL(I,JJ,32)]*S2
ISN 0075
DCOUP(I,J,13)=S2*C5*(DBL(I,JJ,35)+DBL(I,JJ,37))
ISN 0076
DCOUP(I,J,14)=C5*(DBL(I,JJ,45)+DBL(I,JJ,51))

ISN 0077
DCOUP(I,J,16)=S1*C2*Z(JJ)*DBL(I,JJ,47)
ISN 0078
DCOUP(I,J,17)=C3*Z(JJ)*(2*DBL(I,JJ,45)-Z(JJ)*DBL(I,JJ,46))
ISN 0079
DCOUP(I,J,18)=S2*C4*DBL(I,JJ,51)

ISN 0080
 OVLAP(I,J,1)=S2*C1*DBL(I,JJ,5)
ISN 0081
 OVLAP(I,J,2)=S2*C2*(2*DBL(I,JJ,2)-Z(JJ)*DBL(I,JJ,3))
ISN 0082
 OVLAP(I,J,3)=S2*C2*Z(JJ)*DBL(I,JJ,17)
ISN 0083
 OVLAP(I,J,4)=S1*C2*Z(JJ)*DBL(I,JJ,47)
ISN 0084
 OVLAP(I,J,6)=C2*(2*DBL(I,JJ,10)-Z(JJ)*DBL(I,JJ,15))
ISN 0085
 OVLAP(I,J,7)=C3*(2*DBL(I,JJ,7)-Z(JJ)*DBL(I,JJ,12))
ISN 0086
 OVLAP(I,J,9)=S2*C2*Z(JJ)*DBL(I,JJ,26)
ISN 0087
 OVLAP(I,J,10)=C3*Z(JJ)*(2*DBL(I,JJ,39)-Z(JJ)*DBL(I,JJ,49))
ISN 0088
 OVLAP(I,J,11)=C2*Z(JJ)*(2*DBL(I,JJ,10)+DBL(I,JJ,24))
ISN 0089
 OVLAP(I,J,12)=C3*Z(JJ)*(2*DBL(I,JJ,21)+DBL(I,JJ,31))
ISN 0090
 OVLAP(I,J,13)=S2*C5*(2*DBL(I,JJ,39)+DBL(I,JJ,45))
ISN 0091
 OVLAP(I,J,14)=C5*(2*DBL(I,JJ,39)+DBL(I,JJ,45))

ISN 0092
 OVLAP(I,J,16)=S1*C2*Z(JJ)*DBL(I,JJ,42)
ISN 0093
 OVLAP(I,J,17)=S1*C3*Z(JJ)*(2*DBL(I,JJ,49))
ISN 0094
 OVLAP(I,J,18)=S1*C5*DBL(I,JJ,45)
ISN 0095
 OVLAP(I,J,19)=S2*Z(JJ)*DBL(I,JJ,53)
ISN 0096
 OVLAP(I,J,20)=S2*Z(JJ)*DBL(I,JJ,56)

ISN 0097
ACOUP(I,J,3)=S2*(-C2)*Z(JJ)*DBL(I,JJ,37)
ISN 0098
ACOUP(I,J,6)=S2*(-C3)*Z(JJ)*(2*DBL(I,JJ,39)-Z(JJ)*DBL(I,JJ,49))
ACOUP(I,J,13)=S2*(-C5)*(RE*DBL(I,J,J,39)+DBL(I,J,J,45))
ACOUP(I,J,18)=S1*CN*DBL(I,J,J,53)
ACOUP(I,J,4)=S1*(-C2)*2(JJ)*DBL(I,J,J,17)
ACOUP(I,J,9)=S1*(-C3)*2(JJ)*(2*DBL(I,J,J,21)-
Z(J)*DBL(I,J,J,26))
ACOUP(I,J,16)=(-C5)*(RE*DBL(I,J,J,21)+DBL(I,J,J,31))
ACOUP(I,J,19)=S2*CS*DBL(I,J,J,45)

CONTINUE

I=1

CONTINUE

THESE DOUBLE INTEGRATIONS (I.Y, RADIAL,
LIPOLE, OVERLAPPING AND ANGULAR COUPLING)
HAVE AN ADDITIONAL PHASE, EXP(I*R*RDOT/2).
FASREL AND FASING ARE THE REAL AND THE
IMAGINARY PARTS OF THE PHASE FACTOR.

CONTINUE

FASING=R*RDOT/2.0
FASREL=DCOS(FASING)
FASING=DSIN(FASING)

WRITE(3,1020)FASREL,FASING

DO 130 I=1,2
DO 120 J=1,2
DO 110 K=1,25
RCOUP(I,J,K)=RCOUP(I,J,K)*FASREL
DCOUP(I,J,K)=DCOUP(I,J,K)*FASREL

CONTINUE

CONTINUE

CONTINUE

CHANGE SIGN FOR SOME MATRIX ELEMENTS

CONTINUE

CONTINUE

CONTINUE

CONTINUE

CONTINUE

DO 130 I=1,2
DO 120 J=1,2
DO 110 K=1,25
RCOUP(I,J,2)=RCOUP(I,J,2)
RCOUP(I,2,9)=RCOUP(I,2,9)
RCOUP(I,2,14)=RCOUP(I,2,14)
RCOUP(I,2,16)=RCOUP(I,2,16)
RCOUP(I,2,17)=RCOUP(I,2,17)
RCOUP(I,2,18)=RCOUP(I,2,18)
RCOUP(I,2,4)=RCOUP(I,2,4)
RCOUP(I,2,9)=RCOUP(I,2,9)
RCOUP(I,2,14)=RCOUP(I,2,14)
RCOUP(I,2,16)=RCOUP(I,2,16)
ISN 0136
DCOUP (1, 2, 17) = -DCOUP (I, 2, 17)
DCOUP (1, 2, 18) = -DCOUP (I, 2, 18)

ISN 0137
C

ISN 0138
C

ISN 0139
CVLAP (I, 2, 4) = OVLAP (I, 2, 4)

ISN 0140
C

ISN 0141
CVLAP (I, 2, 9) = OVLAP (I, 2, 9)

ISN 0142
CVLAP (I, 2, 14) = OVLAP (I, 2, 14)

ISN 0143
C

ISN 0144
CVLAP (I, 2, 16) = OVLAP (I, 2, 16)

ISN 0145
C

ISN 0146
CVLAP (I, 2, 17) = OVLAP (I, 2, 17)

ISN 0147
CVLAP (I, 2, 18) = OVLAP (I, 2, 18)

ISN 0148
131
CONTINUE

C

E2 = E*R
C6 = 212/E2

ISN 0149
ISN 0150
WT (1) = E2 (1) * BBATIO
WT (2) = E2 (2) * BBATIO

ISN 0151
C7 = 212 ((E2*E2) DO (WT (1) + WT (2))

ISN 0152
C8 = V1*BA/E2

ISN 0153
C

ISN 0154
IF (ITEST, EQ. 1) GO TO 136

ISN 0155
ISN 0156
WRITE (3, 1130) C6, C7, C8, WT (1), WT (2), E2

ISN 0157
DC 156 I = 2

ISN 0158
DO 134 J = 1, 2

ISN 0159
DC 132 K = 1, 25

ISN 0160
WRITE (3, 1030) BCOUP (I, J, K), DCOUP (I, J, K)

1

ISN 0161
OVLP (I, J, K), ACoup (I, J, K)

ISN 0162
CONTINUE

ISN 0163
CONTINUE

ISN 0164
CONTINUE

ISN 0165
CONTINUE

ISN 0166
JJ = 2

ISN 0167
DC 160 J = 1, 2

ISN 0168
DO 150 I = 1, 2

ISN 0169
DO 140 K = 1, 25

ISN 0170
ETMK (I, J, K) = -2 (J) * BCOUP (I, J, K) + C6 * DCOUP (I, J, K) / WT (JJ)
+ C7 * WT (J) * OVLP (I, J, K) / WT (JJ) - C8 * ACoup (I, J, K)

1

ISN 0171
CONTINUE

ISN 0172
CONTINUE

ISN 0173
JJ = 1

ISN 0174
CONTINUE

C

ISN 0175
PRINT OUT THE INTERCHANGE TRANSITION MATRIX ELEMENT

C

ISN 0176
AND OVERLAPPING MATRIX ELEMENTS.

C

ISN 0177
WRITE (3, 1040)

ISN 0178
WRITE (3, 1040) NDATA, NOUP

ISN 0179
IF (NDATA.EQ.1) WRITE (NOUT, 1020) EL, V1, BA, SCANGI

ISN 0180
WRITE (NOUT, 1020) B, EDET, ((ETMK (I, J, K), OVLP (I, J, K),

1

K = 1, 25, I = 1, 2, J = 1, 2)
GO TO 5
C
1000 FORMAT(4I5)
1010 FORMAT(2I5)
1020 FORMAT(2D25.16)
1030 FORMAT(4D20.8)
1040 FORMAT(/33H INTERCHANGE TRANSITION MATRIX ,/)
1130 FORMAT(/' COEFFICIENTS:'//1X,6D15.4)
C
2000 SIGP
C
END
COMPILER OPTIONS - NAME= MAIN, OPT=07, LNEC=58, SIZE=0000K,
SOURCE, FEDIC, NOLIST, DECK, LOAD, MAP, NOEDIT, ID, keep

PROGRAM: DCOPUL

PURPOSE:
Use the precalculated matrix elements to solve the 16-coupled
differential equations as in equation (4.11)

METHOD/STRATEGY:
This program calls DGEAR, an IMSL routine to solve the coupled-
differential equations. The method used is the variable order
ADAMS predictor-corrector method or GEAES method.

INPUT:
E: Incident energy in KEV/AMU in the lab frame
21,22: Projectile and target charge
E21,E22: Projectile and target mass in proton (approximately
AMU)
NDATA: number of data points of the matrix elements
NINPUT: unit number of the file where the matrix elements are stored
NREAD: Unit number for the backup file which stores the
intermediate integrated values.
V1: Incident velocity in A.U.
EA: Impact parameter in A.U.
SCGL: Scattering angle in degrees
B(22): Array of internuclear distance
MET=(22): Array of the time derivatives of the internuclear distance
EXCEL, EVL: (16,2,22) arrays of the real and imaginary parts of the
exchange matrix elements
GYL: (16,2,22) arrays of the real and imaginary parts of the
overlap integrals

METHOD:
parameters necessary for DGEAR subroutine, refer to
IMSL package for detail information
ND: number of coupled differential equations
INDT: indicator for the positive or negative time
=1, positive time
=0, negative time
KMAX: maximum steps allowed in one run
INIT: an indicator to determine where to read the initial data
=0, read from cards
=1, read from backup disk file (unit number = NREAD)
BMAX: maximum B(2) where the time integration will stop
DELT: increment of PHI angle in radians for each integration step
DEL: required tolerance
STEP: step size required by DGEAR subroutine, refer to IMSL
package for detail information
EDIV: the internuclear distance at which the independent
(integration) parameter R or PHI are exchanged
RI(22): array contains the logarithmic of B(22)
IST: an indicator
=0, R, the internuclear distance, is the integration parameter
=1, PHI angle is the integration parameter
XSTR, XEND: input parameter for DGEAR subroutine

OUTPUT:
AY: (16) array contains the values of the a vector, equation (9-12), this is the result for the coupled differential equations
AYSQR: (8) array contains the squares of AY elements
SUM: the sum of the squares of AY elements
PRBEYL: real parts of the probability
PRBSING: imaginary parts of the probability

SUBROUTINE CALLED:
DGEAR (IMSL package)
PROGB

C
REAL*8 EL, V1, B8, SCANGL, B (22), SDCI(22), ETKREL(16, 2, 22),
1 ETKING(16, 2, 22), DML, OYKEL(16, 2, 22), E1(22),
2 OYLING(16, 2, 22), P1KREL(2, 22), P1KING(2, 22), REMATI,
3 ESTR(2, 22), P10ING(2, 22) R

C
REAL SDUMMY(4)

C
INTEGER N1, N2, N3, N4, N5, N6, N7, N8, N9, N10, N11, N12

C
INTEGER Z1, Z2, M1, M2

C
REAL*8 Z12, Z22, Z124, Z224, EAB1, EBA1, EAB2, EBA2, FAB, EBB

C
COMMON /BLOG, EL
1 /CONST1, Z1, Z2, M1, M2
2 /CONST2, REMATI, V1, B8, B0, E, C, BA2, MG
3 /CONST3, NDATA
4 /ETKREL, ETKING, OYKEL, OYLING
5 /ENUC/E
6 /FLAG/END, FR
7 /GEAR/NDUMMY, SDUMMY, IDUMMY
8 /ENERGY/EAB, EBB, EAB1, EBA2, EBA1, EBA2

C
EXTERNAL FCD, FCMD

C
DATA E0, 0.272116085D+2, 1
1 PI/3.14159265358979D0/

C
MRATIO = 1.83615152D+3

C
READ (1, 1030) Z1, Z2, M1, M2

C
READ (1, 1000) NDATA, NINPUT, NREAD

C
Read the exchange and overlap transition matrix elements from disk file
C
READ(NINPUT,1020)EL,Y1,BA,SCANGL
DO 50 I=1,NDATA
   READ(NINPUT,1020)E(J),EDOT(J)
50   CONTINUE
C
ISN 0018    KKK=0
ISN 0019    DO 20 K=1,4
ISN 0020    DO 10 KK=1,4
ISN 0021    KKK=KKK+1
ISN 0022    READ(NINPUT,1020)ETKREL(KKK,J,I),CYREL(KKK,J,I)
ISN 0023    10    CONTINUE
ISN 0024    READ(NINPUT,1010)DUN
ISN 0025    20    CONTINUE
ISN 0026    DO 25 N=1,4
ISN 0027    READ(NINPUT,1010)DUN
ISN 0028    25    CONTINUE
ISN 0029    READ(NINPUT,1020)PIKREL(J,I),P1OREL(J,I)
ISN 0030    KKK=0
ISN 0031    DO 35 N=1,4
ISN 0032    DO 30 K=1,4
ISN 0033    KKK=KKK+1
ISN 0034    READ(NINPUT,1020)ETKING(KKK,J,I),CYLING(KKK,J,I)
ISN 0035    30    CONTINUE
ISN 0036    READ(NINPUT,1010)DUN
ISN 0037    35    CONTINUE
ISN 0038    DO 38 N=1,4
ISN 0039    READ(NINPUT,1010)DUN
ISN 0040    38    CONTINUE
ISN 0041    READ(NINPUT,1020)PIKING(J,I),P1OING(J,I)
ISN 0042    40    CONTINUE
ISN 0043    50    CONTINUE
C
CHANGE THE INTERNUCLEAB DISTANCE INTO LOG SCALE.
DO 100 I=1,NDATA
   B(I)=DLOG(B(I))
100   CONTINUE
C
ISN 0046    Z12=DFLOAT(Z1*Z1)
ISN 0047    Z22=DFLOAT(Z2*Z2)
ISN 0048    Z12=Z12/4.D0
ISN 0049    Z222=Z22/4.D0
ISN 0050    EAB1=(Z12-222)*C.5D00
ISN 0051    EAB2=(Z12-222)*C.5D00
ISN 0052    EBA1=(222-Z12)*C.5D00
ISN 0053    EBA2=(222-Z12)*C.5D00
ISN 0054    EAB=Z12*3.D0/8.D0
ISN 0055    EBA=222*3.D0/6.D0
ISN 0056    E=(1.0D+3)*E1/E0
ISN 0057    MU=DFLOAT(M21*M22)/DFLOAT(M21+M22)
ISN 0058    E=MU+E
C
ISN 0059    MU=MU/MRATIO
ISN 0060    E0=DFLOAT(Z1*Z2)/(2*E)
ISN 0061    EA2=BA*BA
ISN 0062    C=DSQRT(EA2+B0*B0)
ISN 0063  ESTB=(1.0D-2.0D*SCAKGL/180.0D)*0.5D0*PI
ISN 0064  WRITE(3, 1040) ESTB, C, FA2, DO

C READ INPUT TO SOLVE DIFFERENTIAL EQUATIONS
C
ISN 0065  READ(1, 1030) METH, MITER, MD, IND, KMAX, INITIL
ISN 0066  WRITE(3, 1030) METH, MITER, MD, IND, KMAX, INITIL
ISN 0067  READ(1, 1040) BMAX, DELT, TOL, H, RDIV
ISN 0068  WRITE(3, 1040) BMAX, DELT, TOL, H, RDIV

C CHECK IF THE INITIAL VALUES ARE READ FROM DISK
C
ISN 0069  IF (INITIL.EQ.0) GO TO 110
ISN 0071  READ(NREAD) IBT, XSTB, AX
ISN 0072  GO TO 120
ISN 0073  110 READ (1, 1045) IBT
ISN 0074  READ (2, 1050) XSTB, (AY(I), II=1, 16)
ISN 0075  WRITE (3, 1050) IBT, XSTB, (AY(I), II=1, 16)

C FIRST CALL TO DGEAR
C
ISN 0076  INDEX=1
ISN 0077  XND=XSTB+DELT
ISN 0078  IF (IBT .NE. 0) XEND=XSTB+DELT

C  200 K=1, KMAX
C
ISN 0080  CALL DGEAR (ND, FCH, FCNJ, XSTB, H, AY, XEND, TOL,
C            METH, MITER, INDEX, INK, WK, IER)
C            1
C
ISN 0081  Evaluate (a star)*a+(b star)*b

ISN 0082  SUM=0.0D
ISN 0083  DO 135 I=1, 4
ISN 0084     J=I+4
ISN 0085     II=I+4
ISN 0086     JJ=II+4
ISN 0087     AYSQR(I)=AY(I)**24AY(J)**2
ISN 0088     AYSQR(J)=AY(I)**24AY(J)**2
ISN 0089     SUM=SUM+AYSQR(I)+AYSQR(J)
ISN 0090  CONTINUE
ISN 0091  WRITE(1, 1055) IBT, XSTB, (AY(I), II=1, 8), AYSQR(I), II=1, 8
ISN 0092  WRITE(3, 1060) H, XEND, SUM
ISN 0093  INDT=1
ISN 0094  IF (XSTBLE, GT, 0.0D) INDT=1

C Evaluate the overlapping probability

ISN 0096  CALL PROBAB (XSTB, AX, PBBREL, PRBING)
ISN 0097  PBBREL=SUM*PBBREL
ISN 0098  WRITE (3, 1065) PBBREL, PRBING

C CHECK IF THE INTEGRATION REACHES THE TRANSACTION POINT.
C
ISN 0099  IF (IBT .NE. 0) GO TO 730
C IS THE INDEPENDENT VARIABLE OF THE COUPLED DIFFERENTIAL EQUATIONS

C

ISN 0101

ESTB = DABS(ISTR)

ISN 0102

IF (ESTB .GT. BDIV) GO TO 150

ISN 0104

IRT = 1

ISN 0105

WRITE (3, 1110)

ISN 0106

XSTR = INDT * DABCCS((BA2 / ESTB + B0) / C)

ISN 0107

H = DELB

ISN 0108

INDEX = 1

ISN 0109

GO TO 150

C IS THE INDEPENDENT VARIABLE OF THE COUPLED DIFFERENTIAL EQUATIONS

C

ISN 0110

130 ESTB = BA2/(-BO4+C*DCOS(XSTR))

ISN 0111

IF (BA2 .LE. BDIV) GO TO 150

ISN 0113

IRT = 0

ISN 0114

WRITE (3, 1110)

ISN 0115

XSTR = ESTB

ISN 0116

H = DELB

ISN 0117

INDEX = 1

C

ISN 0118

REMIND NREAD

ISN 0119

WRITE (NREAD) IRT, XSTR, AT

ISN 0120

ENDFILE NREAD

ISN 0121

WRITE (3, 1070) IRT, INDT, XSTR, ESTB

ISN 0122

WRITE (3, 1120) DUMMY(8), DUMMY(6), DUMMY(7), DUMMY(8)

ISN 0123

XEND = XSTR + H

C

ISN 0124

IF (BA2 .GT. BMAX .AND. INDT .GT. 0) GO TO 2000

ISN 0126

IF (IRT .GE. 128) GO TO 2000

C

ISN 0128

200 CONTINUE

C

ISN 0129

1000 FCERMAT (315)

ISN 0130

1010 FCERMAT (D25.16)

ISN 0131

1020 FCERMAT (2D25.16)

ISN 0132

1030 FCERMAT (213)

ISN 0133

1040 FCERMAT (D10.3)

ISN 0134

1045 FCERMAT (X5)

ISN 0135

1050 FCERMAT (I2, D15.6 / 16 (2x, D15.6/))

ISN 0136

1055 FCERMAT (I2, D15.6/)

1 6x, * EXCHANGE*, 13x, * DIRECT*, 14x, * SQUARE*/

2 8(2x, 3(D15.6, 5X)/)

ISN 0137

1060 FCERMAT (/ I RT = *, D15.8, * XEND = *, D15.8, * SUM = *, D15.8)

ISN 0138

1065 FCERMAT (* PROBABILITY * REAL = *, D15.6, 5X, * IMAGINARY = *, D15.6)

ISN 0139

1070 FCERMAT (/ I RT = *, I3, * INDT = *, I3, * XSTR = *, D15.8,

1 * BSTR = *, D15.8)

ISN 0140

1100 FCERMAT (/ * The independent variable has changed’,

1 ‘ from R to PHI ’)

ISN 0141

1110 FCERMAT (‘ * The independent variable has changed’,

1 ‘ from PHI to R ’)
** The step size β last used successfully = ',D25.8/
1   ** The order last used successfully = ',I10/
2   ** The cumulative number of steps taken = ',I10/
3   ** The cumulative number of FCN evaluations = ',I10/

** C

** ISN 0143 2000 STOP
** ISN 0144 END
LEVEL 21.8 ( JUN 74 )

OS/360 FORTRAN H

COMPILER OPTIONS = NAME= MAIN,OPT=02,LINECNT=58,SIZE=0000K,
SOURCE,EDCIC,NOLIST,DECK,LOAD,MAP,NOEDIT,ID,XREF

C
C SUBROUTINE PROBATE(X,Y,PBREL,PBING)

C
C REAL*8 X,Y(16),PBREL,PBING
C REAL*8 R,BSTRA1(16),BSTRA2(16),ASTRB1(16),ASTRB2(16),
C 1 YR(22),YI(22),PBFAC

C
C COMMON /RLOG/RL
C REAL*8 RL(22)

C
C COMMON /CONST3/NDATA
C INTEGER NDATA

C
C COMMON /ETMTRK/ETKREL,ETKING,OVLREL,OVLING
C REAL*8 ETKREL(16,2,22),ETKING(16,2,22),OVLREL(16,2,22),
C 1 OVLING(16,2,22)

C
C COMMON /FLAG/INDT,IRT
C INTEGER INDT,IRT

C
C COMMON /ENERGY/EAA,EEA,EEA1,EAA1,EAB,EBAL,EBAL1,EBA2,EBA2
C REAL*8 EAA,EEA1,EAA1,EAB,EBAL,EBAL1,EBA2,EBA2

C
C COMMON /PHASE/RELAB1,IMGAB1,RELBA1,IMGBA1,
C 1 RELAB2,IMGAB2,RELBA2,IMGBA2,
C 2 SI,SR
C REAL*8 RELAB1,IMGAB1,RELBA1,IMGBA1,RELAB2,IMGAB2,RELBA2,
C 1 IMGBA2,SI(2,16),SR(2,16)

C
C

C
C INITIALIZATION

C
C CALL TIMENG(X,Y)

C
C Interpolate Swm(A,B) and Smn(B,A), both imaginary and real

C
C
C DO 30 J=1,2
C
C DO 20 II=1,16
C
C DO 10 I=1,NDATA
C
C YR(I)=OVLREL(II,I,J)
C
C YI(I)=OVLING(II,I,J)

C
C 10 CONTINUE

C
C
C 20 CONTINUE

C
C 30 CONTINUE

C REDEFINE SR AND SI, CONSIDERING THE PHASE FACTORS.

C
C CALL EPHASE(INDT)

C
C Evaluate (b Star)*a
| LS   | 0029 | K=0   |
| LS   | 0030 | DC 50 J=9,12 |
| LS   | 0031 | DO 40 I=1,4 |
| LS   | 0032 | K=K+1 |
| LS   | 0033 | II=I+4 |
| LS   | 0034 | JJ=J+4 |
| LS   | 0035 | BSTRA1(K)=Y(I)*Y(J)+Y(II)*Y(JJ) |
| LS   | 0036 | BSTRA2(K)=Y(II)*Y(J)-Y(JJ)*Y(I) |
| LS   | 0037 | 40 CONTINUE |
| LS   | 0038 | 50 CONTINUE |
| LS   |       | Evaluate (a Star)*b |
| LS   |       | K=0   |
| LS   | 0039 | DC 70 I=1,4 |
| LS   | 0040 | DO 60 J=9,12 |
| LS   | 0041 | K=K+1 |
| LS   | 0042 | II=I+4 |
| LS   | 0043 | JJ=J+4 |
| LS   | 0044 | ASTRB1(K)=Y(I)*Y(J)+Y(II)*Y(JJ) |
| LS   | 0045 | ASTRB2(K)=Y(II)*Y(J)-Y(JJ)*Y(J) |
| LS   | 0046 | 60 CONTINUE |
| LS   | 0047 | 70 CONTINUE |
| LS   |       | Evaluate real parts and imaginary parts |
| LS   | 0048 | PRBREL=0. DO |
| LS   | 0049 | PRBIMG=0. DO |
| LS   | 0050 | DC 80 I=1,16 |
| LS   | 0051 | PRBREL=PRBREL+ASTR1(I)*SR(1,1)-ASTR2(I)*SI(1,1) |
| LS   | 0052 | +ASTR1(I)*SE(2,1)-ASTR2(I)*SI(2,1) |
| LS   | 0053 | PRBIMG=PRBIMG+ASTR1(I)*SI(1,1)+ASTR2(I)*SR(1,1) |
| LS   | 0054 | +ASTR2(I)*SE(2,1)+ASTR1(I)*SI(2,1) |
| LS   | 0055 | 80 CONTINUE |
| LS   | 0056 | RETURN |
| LS   |       | END |
SUBROUTINE FCN(N,X,Y,YPRIME)

C PURPOSE:
  this program provides the first derivatives (YPRIME(1),..., YPRIME(N)) of the Y(I) with respect to X. Please refer to Section 4.2 for details.

C CALLING SEQUENCE:

C INPUT:
  N:  # of coupled differential equations
  X:  independent variable, at which the first derivative should be evaluated.
  Y:  a N vector, a dependent variable

C OUTPUT:
  YPRIME: a N vector, which is the first derivative of Y with respect to X.

C SUGGESTION/HINT:
  The real and imaginary parts of the matrix elements are stored in two different arrays, SR and SI. Each array has dimension (2,16), the row number indicates (A,B) (I=1) and (E,A) (I=2) combinations.

C CALLED BY:  DGEAR

C SUBROUTINES AND FUNCTION(S) CALLED BY THIS PROGRAM:

TIBENG
TRMFAC (REAL*8 FUNCTION)
EPHASE
DISTH
LINV2P (IMSL ROUTINE)
VHULPF (IMSL ROUTINE)

C DATE:  MAR. 1980

C PROGRAMMER:  TEN-HSUIN K. WU

REAL*8 X,Y(16),YPRIME(16),UL(22),VR(22),YI(22),R,
1 ETKREL(16,2,22),ETKING(16,2,22),OVLDEL(16,2,22),
2 OVLING(16,2,22),OVLAP(16,16),INVOP(16,16),
3 WKAERA(304),SI(2,16),SE(2,16),UTKREL(2,16),
4 DTHING(2,16),EBMS(16,16),EBNAMP(16,16),FRMFAC

REAL*8 NEATIC,V1,EA,EO,E,C,EB2,MU,VEL
REAL*8 EAA,EBB,EAB1,EAB2,EB1,EB2
REAL*8 RELA1,IMG1A,RELBA1,IMGAB1,RELAB2,IMGAB2,
1 RELB2,IMGBA2,RELAA,IMGBA,RELBA,IMGEEB

INTEGER N,NDATA,INDT,IBT,21,22,EB1,EB2
COMMON /DIRFAS, ELEZAA, XINGAA, XEEBB, XINGBB

C C

ISN 0010

C C

INITIALIZATION

CALL TIMENG(K, R)

C C

Interpolate S_m(k, b) and S_m(b, a), both imaginary and real

C C

ISN 0012

DO 30 J=1, 2, 1, 16

30 CONTINUE

ISN 0013

DO 20 II=1, 16

20 CONTINUE

ISN 0014

KDATA

ISN 0015

YB(I)=OVLEL(II, J, I)

ISN 0016

YI(I)=OVLEL(II, J, I)

ISN 0017

CONTINUE

ISN 0018

SR(J, II)=FRLVAC(NDATA, BL, YB, E, 10, 10, DO)

ISN 0019

SI(J, II)=FRLVAC(NDATA, BL, YI, E, 10, 10, DO)

ISN 0020

CONTINUE

ISN 0021

CONTINUE

C C

REDEFINE SR AND SI, CONSIDERING THE PHASE FACTORS.

ISN 0022

CALL EPHASE(INDT)

C C

Initially set OVELAP zero.

ISN 0023

DO 50 I=1, 16

50 CONTINUE

ISN 0024

DO 40 J=1, 16

40 CONTINUE

ISN 0025

OVELAP(I, J)=0. DO

ISN 0026

50 CONTINUE

C C

Construct the nonzero upper-right part of OVELAP

ISN 0027

II=0

ISN 0028

DO 70 KK=1, 4

70 CONTINUE

ISN 0029

LI=KK4

ISN 0030

DO 60 K=9, 12

60 CONTINUE

ISN 0031

L=KK4

ISN 0032

II=II4

ISN 0033

OVELAP(KL, K)=SI(1, II)

ISN 0034

OVELAP(LL, I)=-SI(1, II)

ISN 0035

OVELAP(LL, K)=SI(1, II)

ISN 0036

OVELAP(KK, L)=SI(1, II)

ISN 0037

60 CONTINUE

ISN 0038

70 CONTINUE

C C

Construct the lower-left part of OVELAP
II=0
DC 90 KK=9,12
LL=KK+4
DO 80 K=1,4
L=K+4
II=II+1
OVERLAP(KK,K)=-SI(2,II)
OVERLAP(KK,L)=-SB(2,II)
OVERLAP(LL,K)=SB(2,II)
OVERLAP(LL,L)=-SI(2,II)
80 CONTINUE
DO 80 CONTINUE

C
DO 100 K=1,4
KK=K+4
OVERLAP(K,KK)=-1.D0
OVERLAP(K,K) = 1.D0
100 CONTINUE

C
DC 110 K=9,12
KK=K+4
OVERLAP(K,KK)=-1.D0
OVERLAP(K,K) = 1.D0
110 CONTINUE

C
Find the inversion of OVERLAP
C
CALL LINV2P(OVERLAP,16,16,INVOVR,6,WHAREA,1EE)
C
Interpolate Kmn(A,B) and Kmn(B,A)
C
DO 140 J=1,2
DO 130 II=1,16
DO 120 I=1,NDATA
YR(I)=ZEKEEL(II,J,I)
YI(I)=ZKING(II,J,I)
120 CONTINUE
130 CONTINUE
140 CONTINUE

C
Redefine SI and SB, considering the phase factors.
C
CALL EPHAS(EWDT)
C
Calculate the direct transition matrix elements, 
Kmn(A) and Kmn(B), these matrices will pass in 
through the common block DMATR
C
CALL DIETEN(B)
C
Redefine real and imaginary parts of the direct transition 
matrix elements.
C
DO 142 I=2,4
BEL=DTHREL(1, I)
IMG=DTHING(1, I)
DTHREL(1, I)=BEL*RELEAA-IMG*IMGEAA
DTHING(1, I)=BEL*IMGEAA+IMG*RELEAA
BEL=DTHREL(2, I)
IMG=DTHING(2, I)
DTHREL(2, I)=BEL*RELEBB-IMG*IMGEBB
DTHING(2, I)=BEL*IMGEBB+IMG*RELEBB

DO 144 I=5, 13, 4
C CONTINUE
BEL=DTHREL(1, I)
IMG=DTHING(1, I)
DTHREL(1, I)=BEL*RELEAA-IMG*IMGEAA
DTHING(1, I)=BEL*IMGEAA+IMG*RELEAA
BEL=DTHREL(2, I)
IMG=DTHING(2, I)
DTHREL(2, I)=BEL*RELEBB-IMG*IMGEBB
DTHING(2, I)=BEL*IMGEBB+IMG*RELEBB
C CONTINUE
C Rearrange Hm(A), Hm(B), KmA(A, B) and KmA(B, A)

MM=0
DO 160 K=1, 4
L=E44
J=L44
J=I44
DO 150 KK=1, 4
LL=KK+4
II=LL+4
JJ=I4+4
MM=MM+1
HAMP(K, KK)=DTHREL(1, MM)
HAMP(K, LL)=DTHING(1, MM)
HAMP(K, II)=SE(1, MM)
HAMP(K, JJ)=SI(1, MM)
HAMP(L, KK)=DTHREL(1, MM)
HAMP(L, LL)=DTHING(1, MM)
HAMP(L, II)=SI(1, MM)
HAMP(L, JJ)=SR(1, MM)
HAMP(I, KK)=SE(2, MM)
HAMP(I, LL)=SI(2, MM)
HAMP(I, II)=SI(2, MM)
HAMP(I, JJ)=DTHING(2, MM)
HAMP(J, KK)=SI(2, MM)
HAMP(J, LL)=SE(2, MM)
HAMP(J, II)=DTHING(2, MM)
HAMP(J, JJ)=DTHREL(2, MM)
C CONTINUE

C Multiply (INVOB*HAMP)
CALL VMULPP(INVOB, HAMP, 16, 16, 16, 16, TRHAMP, 16, IER)
VEL=V1*BA/B

IF (IST.EQ.0) GO TO 163

C C THETA is the independent variable
C
VEL=VEL/R
GO TO 165

C C R is the independent variable
C
VEL=DSQRT(2.DO*(2-DFLOAT(Z1*Z2))/8./HU-VEL**2)
VEL=VEL*IKD
CONTINUE
WRITE(3,1030)VEL
FORMAT('VEL = ',D25.16)
C C Calculate YPRIME
C
DC 180 I=1,N
DO 170 J=1,N
YPRIME(I)=0.DO
170 CONTINUE
YPRIME(I)=YPRIME(I)+TBWAMP(I,J)*Y(J)
CONTINUE
YPRIME(I)=YPRIME(I)/VEL
C C CONTINUE
C
RETURN
C C END
COMPILER OPTIONS - NAME= MAIN,OPT=02,LINENUM=58,SIZE=0000K,
SOURCE,EBCDIC,NOLIST,DECK,LOAD,MAP,NCEDIT,LD,XREF

ISH 0002   SUBROUTINE FCNJ (N,X,Y,PD)
ISH 0003   INTEGER N
ISH 0004   REAL*8 Y(N),PD(N,N),X
ISH 0005   RETURN
ISH 0006   END
SUBROUTINE PHASE(INDT)

C PURPOSE:
C when the exchange matrix elements were evaluated in TRIMAY, it
C did not consider time factor. Since t<0, DDT<0 and t>0, DDT>0,
C an adjustment of sign according to the time range is required.
C PHASE handles the above adjustment. Then the energy
C phase factor which is evaluated in TIMESD and passed in
C through Common block is multiplied to the exchange matrix
C elements.

RESIDUE RELAB1,IMGAB1,BLBAB1,IMGAB2,BLBAB2,IMGAB2,
1 RELAB2,IMGAB2,SR(2,16),SR(2,16)

INTEGER INDT

COMMON /PHASE/RELAB1,IMGAB1,BLBAB1,IMGAB2,
1 RELAB2,IMGAB2,SR(2,16),SR(2,16)
2 SI,SR

C Consider the translation factor EXP(1R4*DDT/2)
C where DDT has different sign for time greater
C than or less than zero.

DO 10 I=1,16
   SI(1,I)=SI(1,I)*INDT
   SI(2,I)=SI(2,I)*INDT
10   CONTINUE

C Consider the energy phase factor.

DO 20 I=2,4
   BDL=SR(1,I)
   IMG=SF(1,I)
   SR(1,I)=BDL*REL1-IMG*IMGAB1
   SI(1,I)=REL1*IMGAB1+IMG*RELAB1
   EEL=BR(2,I)
   IMG=SI(2,1)
   SR(2,1)=BDL*REL1-IMG*IMGAB1
   SI(2,1)=REL1*IMGAB1+IMG*RELAB1
20   CONTINUE

DO 30 I=5,13,4
   BDL=SR(1,I)
   IMG=SF(1,I)
   SR(1,I)=BDL*REL1-IMG*IMGAB2
   SI(1,I)=REL1*IMGAB2+IMG*RELAB2
   EEL=BR(2,I)
   IMG=SI(2,1)
   SR(2,1)=BDL*REL1-IMG*IMGAB2
   SI(2,1)=REL1*IMGAB2+IMG*RELAB2
30   CONTINUE

RETURN
END
This program evaluates the real and imaginary parts (\( \text{REL}, \text{IMG} \)) of the value \( (E^T) \) where \( E \) and \( T \) are energy level and time respectively.

\[
E = E^T
\]

\[
\text{REL} = \text{DCOS}(D)
\]

\[
\text{IMG} = \text{DSIN}(D)
\]

END
INPUT:

- **X** (XI): number of data points.
- **Y** (XI): the logarithmic of the R (X), internuclear distances.
- **X** (XI): the real scale of the matrix elements corresponds to the different R, (R (X) is input through the common block.)

OUTPUT:

- Cubic spline fit to (X,Y).
- **FMFAC** is the interpolated value at **X**.
- *There are several ways to fit the data (X,Y):*
  1. for **X**(XI/3) <= **X** <= **X**(XI), use cubic spline fit to (X,Y).
  2. for **X**(1) <= **X** <= **X**(XI/3), the data near the left end are very large compared with **Y** (input) and changing rapidly, use cubic spline fit to (X,DSQR(X)*LOG(Y)). Otherwise, just use the cubic spline fit to (X,Y).

```
REAL*8 X(XI), Y(XI), B(22), XE, YM, XL, YS(10), YL
INTEGER FLAG, XI3, IEV, INTHRD, JJ
COMMON /BNUC/N

NN number of points adjacent to **X** will be fitted.
find **X**'s position.

IF (IEV=XI3) 20, 10, 10
10 CALL ABAFIX (IEV, XI, XI, NN, XX, YY, JJ)
20 CALL CSFLNH (XX, YY, SB, XL, FMFAC)
RETURN

IF (Y(1))=0, 30, 40
30 IF (DABS(Y(2)).LE.YM) GO TO 10
40 IF (DABS(Y(2)/Y(3)).LE.2.0D0) GO TO 10
50 CALL ABAFIX (XI, XI, XI, XI, YY, JJ)
60 DO 1=1, NN
70 IF (YY(JJ))=0, 60, 70
80 CONTINUE
```
C All the YY(NN) are negative, change the sign and then take the logarithmic of them.

C EC 80 I = 1, NN
ISN 0029

ISN 0030

ISN 0031

80 CONTINUE

ISN 0032

IF (FLAG) 100, 90, 100

ISN 0033

100 DC 110 I = 1, NN

ISN 0034

JI = JJ + I

ISN 0035

YS(I) = DLOG(YY(I)) * R(JJ)/(1.0 + R(JJ))

ISN 0036

CONTINUE

ISN 0037

CALL CSFLN(XX, YY, NN, XL, XL)

ISN 0038

FEMPAC = - DEEP (YL * (1.0 + XE)/XE)

ISN 0039

RETURN

C

ISN 0040

90 YAM = MIN (YY(2), YY(3))

ISN 0041

DO 120 I = 4, NN

ISN 0042

YAM = MIN (YAM, YY(I))

ISN 0043

120 CONTINUE

C

ISN 0044

SHIFT = YAM/2.0 + 1.000

ISN 0045

DO 130 I = 1, NN

ISN 0046

JI = JJ + I

ISN 0047

YS(I) = DLOG(YY(I) + SHIFT) * R(JJ)/(1.0 + R(JJ))

ISN 0048

CONTINUE

ISN 0049

CALL CSFLN (XX, TS, NN, XL, XL)

ISN 0050

FEMPAC = (DEEP (YL * (1.0 + XE)/XE) - SHIFT)

ISN 0051

RETURN

C

ISN 0052

70 IF (FLAG. EQ. 0) YMIN = MIN (YY(2), YY(3))

ISN 0054

IF (FLAG. EQ. 1) YMIN = MIN (YY(I), YY(2), YY(3))

ISN 0056

YAM = MIN (YAM, DABS (YY(2)), DABS (YY(3)))

ISN 0057

DO 150 I = 4, NN

ISN 0058

YMIN = MIN (YMIN, YY(I), YMIN)

ISN 0059

YAM = MIN (YAM, DABS (YY(I)), YAM)

ISN 0060

CONTINUE

ISN 0061

SHIFT = DABS (YMIN) + YAM/2.0 + 1.000

ISN 0062

DO 160 I = 1, NN

ISN 0063

JI = JJ + I

ISN 0064

YS(I) = DLOG (YY(I) + SHIFT) * R(JJ)/(1.0 + R(JJ))

ISN 0065

CONTINUE

ISN 0066

CALL CSFLN (XX, TS, NN, XL, XL)

ISN 0067

FEMPAC = DEEP (YL * (1.0 + XE)/XE) - SHIFT

ISN 0068

RETURN

ISN 0069

END
SUBROUTINE ENDPBM(X1, X2, X3, Y1, Y2, Y3, A, B)

REAL*8 X1, X2, X3, Y1, Y2, Y3, A, B

B = (Y1 - Y2) / (X1 - X2)

A = (B - (Y1 - Y3) / (X1 - X3)) / (X2 - X3)

RETURN

END
SUBROUTINE CSPLN(X, Y, NX, XX, YY)
C
REAL*8 RX(NX), Y(NX), BPAP(4)*C(9, 3), XX, YY,
1 AL, BL, AB, BD
INTEGER IC, NX1, NX2
C
IC=NX-1
C
FIT THE END POINTS WITH QUADRATIC POLYNOMIALS.
CALL ENDP2M(X(1), X(2), X(3), Y(1), Y(2), Y(3), AL, BL)
ISN 0007 BPAP(1)=0. DO
ISN 0008 BPAP(2)=4. DO+AL
C
NX1=NX-1
ISN 0009 NX2=NX-2
ISN 0010 CALL ENDM(A(NX2), X(NX1), X(NX), Y(NX2), Y(NX1), Y(NX), AE, BR)
ISN 0011 BEAP(3)=0. DO
ISN 0012 BEAP(4)=4. DO+AB
C
CALL ICSCIU(X, Y, NX, BPAP, C, IC, IER)
ISN 0015 IERV=INTIRD(NX, 2)
ISN 0016 D=XX*X(IEVL)
ISN 0017 YY=(C(IEVL, 3)*E+C(IEVL, 2))*E+C(IEVL, 1),
1 1
ISN 0018 RETURN
ISN 0019 END
LEVEL 21.6 (JUN 74)

COMPILER OPTIONS - NAME= MAIN, OPT=02, LINECNT=58, SIZE=0000K 
SOURCE, EBCDIC, NOLIST, DECK, LOAD, MAP, NODIT, IL, XREF

ISN 0002 INTEGER FUNCTION INTRVD(D,D2,D2)
ISN 0003 REAL*8 DL,D(N)
ISN 0004 INTEGER LEFT,RIGHT
ISN 0005 INTRVD=0
ISN 0006 IF (DD.LT.D(1)) GO TO 100
ISN 0008 IF (DD.GT.D(N)) GO TO 100
ISN 0010 LEFT=1
ISN 0011 RIGHT=N
ISN 0012 10 MID=(LEFT+RIGHT)/2
ISN 0013 IF (DD-D(MID)) 20,30,40
ISN 0014 20 RIGHT=MID
ISN 0015 GC TO 50
ISN 0016 30 INTRVD=MID
ISN 0017 RETURN
ISN 0018 40 LEFT=MID
ISN 0019 50 LEFT=LEFT+1
ISN 0020 IF (LEFT.LT.RIGHT) GO TO 10
ISN 0022 INTRVD=LEFT
ISN 0023 IF (DD.EQ.D(RIGHT)) INTRVD=RIGHT
ISN 0025 RETURN
ISN 0026 100 WRITE(3,1000) DD
ISN 0027 1000 FORMAT(//" ** THE VALUE IS OUT OF THE 
" +" ARRAY RANGE **",E15.6)
ISN 0028 RETURN
ISN 0029 END
SUBROUTINE ARAPIX(IEV, NX, NY, NN, XX, YY, JJ)

Construct the array (XX(NN),YY(NN)) pairs to do the cubic spline fit.

REAL*8 XX(NN),YY(NN),X(22),Y(22)

IF IEV in near the right end.

IF (IEV-NR) 20, 10, 10

JJ=NX-NN

GO TO 50

If IEV is near the left end.

IF (IEV-NL) 30, 30, 40

JJ=0

GC TO 50

JJ=IEV-NL

CONTINUE

DO 60 I=1,NN

XX(I)=X(JJ+I)

YY(I)=Y(JJ+I)

CONTINUE

RETURN

END
SUBROUTINE DIREN(B)

C SUBROUTINE: DIREN(B)
C
C INPUT:
C B: Internuclear distance.
C
C INPUT THROUGH COMMON BLOCK:
C Z1, Z2, Z1, Z2:
C Charges and masses of the projectile and target nuclei.
C BEVIQ: Mass ratio of proton to electron (1 a.u.).
C V1: Incident velocity in a.u.
C E1: Impact parameter in a.u.
C
C OUTPUT THROUGH COMMON BLOCK:
DTHREL(2,16), DTHING(2,16):
C Real and imaginary parts of the direct transition matrix elements.
C
C FUNCTIONS CALLED BY THIS PROGRAM:
GAMA, BETA
C
C DATE: Aug. 8, 1979
C PROGRAMMER: T.K. NG
C
REAL*8 Z(2), W(2), Z12, BEVIQ, G(9), B(4),
1 V1, BA, R, DTHREL(2,16), DTHING(2,16),
2 GAMA, BETA, T1, A1, A2, A3, A4, A5, B2, DP(16),
3 TWOSQ, BB, BO, E, C, BA2, BU

INTEGER Z1, Z2, Z1, Z2
COMMON /CONST1/ Z1, Z2, Z1, Z2
COMMON /CONST2/ BEVIQ, V1, BA, BO, E, C, BA2, BU
COMMON /DTRAN/DTHREL, DTHING
COMMON /ICNUM/ BB

IF (E = BB) THEN
Z(1) = DFLOAT(Z1)
Z(2) = DFLOAT(Z2)
Z12 = Z(1) * Z(2)
W(1) = DFLOAT(Z1) * BEVIQ
W(2) = DFLOAT(Z2) * BEVIQ

II = 2
DC 100 I = 1, II

C Construct the basic formula for DP elements.
DO 10 J = 1, 5
G(J) = GAMA(J, Z(I))
10 CONTINUE

R = J
T1 = Z(1) * 1.5D0
DO 20 J=1,3
   JJ=J+1
   G(J)=GAMA(J,71)
   B(J)=BETA(3J,21)
   K=K+1
20    CONTINUE
B(4)=BETA(2,71)
C
T1=2.DO*Z(J)
G(9)=GAMA(1,71)
C
Initialize all the coefficients.

TWO5=GDSQRT(2.DO)*2.DO
A1=2(1)
A2=A1*A1
A3=A2*A1
A4=A3*A1
A5=A4*A1/240.DO
E2=2.0E-8

DO 30 J=1,16
   DP(J)=0.DO
30    CONTINUE

DO 40 J=1,16
   DTHREL(I,J)=0.DO
40    CONTINUE

IF (I.EQ.1) GO TO 40

DP(3)=-DP(3)
DP(7)=-DP(7)
A1=-2(1)
A2=-Z12*(W(1)*PB2)
A3=Z12*w(12)/(W(1)*(W(1)+W(2)))*E*2.DO
A4=V1*BA/E2

DTHREL(I,1)=A1*DP(1)+A3
DTHREL(I,2)=A1*DP(2)
DTHREL(I,3)=A1*DP(3)+A2*TWO5*64.DO/(2.43D0*Z(1))
DTHREL(I,4)=A1*DP(4)+A3
DTHREL(I,5)=A1*DP(7)+A2*(-3.DO/Z(1))
DTHREL(I,6)=A1*DP(11)+A3
DTHREL(I,7)=A1*DP(16)+A3
DTHREL(I,8)=DTHREL(I,2)
DTHREL(I,9)=DTHREL(I,3)

C
ISN 0068
DHING(I, 12) = A4
ISN 0069
DHING(I, 15) = A4
ISN 0070
C
II = 1
ISN 0071
C
100
CONTINUE
ISN 0072
C
RETURN
ISN 0073
END
DOUBLE PRECISION FUNCTION BETA(N,B)

Evaluate a series expansion.

INTEGER N1,N
COMMON /INTNUM/B

N1=N+1
B=B*B
IF (N) 20,10,20
EITA=F1.DO+DEXP(-BB)/BR
GO TO 40
S=DFLOAT(N1)
S1=1.DO
DC 30 N=1,N
S1=S1*BR/DFLOAT(N)
S=S+S1*DFLOAT(N1-B)
CONTINUE
BETA=F1.DO+DEXP(-BB)*S/BR
FAC=1.DO/B
DC 50 I=1,N
FAC=FAC*DFLOAT(I)/B
BETA=2.DO*BETA+FAC
RETURN
END
**DOUBLE PRECISION FUNCTION GAMA(N, G)**

**Evaluate a series expansion.**

```
REAL*8 R, GR, S, S1, S2, FAC
INTEGER N, N1, H, I
COMMON /INTNOC/R
N1 = N - 1
GR = G*R
S = DFPLOAT(N)
S1 = 1.0
IF (N.EQ.1) GO TO 20
DC 10 H=1,N1
S1 = S1*GR/DFLOAT(M)
S = S + S1*(N-H)/DFLOAT(N+1)
CCONTINUE
10 S2 = DFPLOAT(N+1)/GR
S = S2 - (S + S2)*DFLOAT(-GR)
FAC = 1.0
DO 30 I = 1, N
   FAC = FAC*I/G
30 GAMA = 2.0*FAC*S
RETURN
END
```
C
C PURPOSE:
C TIMING calculates time by given internuclear distance and
C evaluates the real and imaginary parts of the energy phase
C factor \exp(i(2\pi\theta_0)t).
C
C INPUT:
C X: an independent variable, it may be internuclear distance
C or the angle \phi.
C
C OUTPUT THROUGH COMMON BLOCKS:
C /EPRASE/
C /DISPAC/
C
C INPUT THROUGH COMMON BLOCKS:
C /FLAG/
C /CONST2/
C /ENERGY/
C
C PROGRAMMER: TZE-HSIN K. WU
C DATE: MAR. 1980
C
C
C
C REAL*8 X,E
C REAL*8 THETA,TANGEN,TIME
C
C COMMON /FLAG/INDT,IBT
C INTEGER INDT,IBT
C
C COMMON /CONST2/MRATIO,Y1,BA,B0,E,C,BA2,W0
C REAL*8 MEATIO,Y1,BA,B0,E,C,BA2,W0
C
C COMMON /ENERGY/EAA,EBB,ZAB1,ZEB2,EEA1,EEA2,EBAB1
C REAL*8 EAA,EBB,ZAB1,ZEB2,EEA1,EEA2,EBAB1
C
C COMMON /PHASE/RELAB1,IMGAB1,RELAB1,IMGAB1,
C 1 RELAB2,IMGAB2,RELAB2,IMGAB2,
C 2 SI,SB
C REAL*8 RELAB1,IMGAB1,RELAB1,IMGAB1,RELAB2,IMGAB2,RELAB2,
C 1 IMGAB2,SI(2,16),SB(2,16)
C
C COMMON /DISPAC/RELAA,IMGAA,RELEBB,IMGEBB
C REAL*8 RELAA,IMGAA,RELEBB,IMGEBB
C
C INDT=-1
C IF (X.GT.0. DO) INDT=1
C S=DAUX(X)
C IF (IBT.NE.0) R=BA2/(-BO+C*DCOS(X))
C CALCULATE TIME
C
C THETA=X
C IF (IBT.EQ.0) THETA=INDT*CARCOS((BA2/R+BO)/C)
C TANGEN=DIAN(THETA/2.0)+BA
C TIME=C-B0
TIME=(B*C*DSIN(THETA)/BA+)
1   BO=DLOG(-(TANGEN*TIME)/(TANGEN-TIME))

WRITE(3,1020)R,THETA,TANGEN,TIME
1020 FORMAT(*'R=',D20.10,'  THETA=',D20.10,'  TANGEN=',D20.10)

C       CALL REAL AND IMAGINARY PARTS OF THE ENERGY PHASE FACTORS
C       CALL RELING(EAB1,TIME,REAL1,IMGAB1)
C       CALL RELING(EAB2,TIME,REAL2,IMGAB2)
C       CALL RELING(EBA1,TIME,REAL1,IMGBA1)
C       CALL RELING(EBA2,TIME,REAL2,IMGBA2)
C       CALL RELING(EAA,TIME,REAL1,IMGAA1)
C       CALL RELING(EBB,TIME,REAL1,IMGBB1)

RETURN
1037   END
//ELECAP JOB UNC.P:Snnnn,WU.T-H,P=50,T=30,DEST=UNC,PRTY=9
//*PW=xxxxx
//EXEC FTHGD,R.G=190K
//G.DECK DD DSN=UNC.P:S2158.WUTH.HE2H.ELECAP,DISP=OLD
//G.FT05F001 DD DSN=UNC.P:S6502.WUTH.HE2H.BACKUP,DISP=OLD
//G.FT07F001 DD DSN=UNC.P:S6502.WUTH.HE2H.DBLINT,DISP=OLD
//G.SYSIN DD *
   1   5   7   6
   2   1
   4   1
42  114
163   1
  3   1
  2   1
  3   1
  2   1
  3   2
  1   3
  3   2
  2   2
  3   2
  1   3
  1   4
  0   0
  3   3
  1   4
  1   4
  0   0
  1   5
  0   0
  2   2
  0   0
  3   2
| 2  | 2  |
| 1  | 4  |
| 0  | 0  |
| 3  | 3  |
| 1  | 4  |
| 2  | 2  |
| 0  | 0  |
| 1  | 4  |
| 0  | 0  |
| 1  | 5  |
| 0  | 0  |
| 2  | 3  |
| 0  | 0  |
| 1  | 5  |
| 0  | 0  |
| 2  | 3  |
| 0  | 0  |
| 1  | 5  |
| 0  | 0  |
| 1  | 1  |
| 0  | 1  |
| 1  | 1  |
| 0  | 1  |
| 1  | 1  |
| 0  | 1  |
| 0  | 1  |
| 0  | 1  |
| 0  | 1  |
| 0  | 1  |
0 1
1.0D+2 5.0D+0
0.5D0 1.0D00
1.0D-5 5.2D00
1.0D+2 5.0D+0
1.0D00 0.5D00
1.0D-5 5.2D00

This JCL will produce the double integrals for R=5.2 a.u. The results will be stored on disk 'UNC.P.S6502.WUTH.HE2H.DBLINT'.
//TRANMAX JOB UNCP.Snnnn, WUT-H, T=3, P=100, DEST=UNCP,
// PRTY=0
//*PW=xxxxx
// EXEC FTHGD, R.G=200K
//G.DECK DD DSN=UNCP..S6502.WUTH.HE2H.TRANOBJ, DISP=OLD
//G.FT05F001 DD DSN=UNCP..S2158.WUTH.HE2H.TRANMAX, DISP=OLD
//G.SYSIN DD *
  1   5  114   1
(22 data sets of the double integral results from ELECAP)
//
//DCOUPL JOB UNC.P.Snnnn,WU.T-H,P=200,T=10,PRTY=0,DEST=UNC
//*PW=xxxxxx
//G EXEC FTHGD,R.G=300K
//G.DECK DD DSN=UNC.P.S2158.WUTH.HE2H.DGEAR,DISP=OLD
//G.FT05F001 DD DSN=UNC.P.S2158.WUTH.HE2H.TRANMAX,DISP=OLD
//G.FT07F001 DD DSN=UNC.P.S2158.WUTH.HE2H.DIFF,DISP=OLD
//G.SYSIN DD *
  2 1
  4 1
  22 5 7
  1 0
  16 -1
  500 1
    6.00
    .0050
    .0050
   1.0E-6
   1.0E-5
   1.0D-2
   0
   -6.D0
   0.
   0.
   0.
   0.
   0.
   0.
   1.0
   0.
   0.
   0.
   0.
   0.
   0.