110
12/4/79

AD. 361

MLM-2661

MASTER

**Algorithms and FORTRAN Programs
to Calculate Classical Collision
Integrals for Realistic Intermolecular
Potentials**

William L. Taylor

November 29, 1979

Monsanto

MOUND FACILITY

Miamisburg, Ohio 45342

operated by

MONSANTO RESEARCH CORPORATION

a subsidiary of Monsanto Company

for the

U. S. DEPARTMENT OF ENERGY

Contract No. DE-AC04-76-DP00053

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

NOTICE

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for any third party's use or the results of such use of any information, apparatus, product or process disclosed in this report, or represents that its use by such third party would not infringe privately owned rights.

Printed in the United States of America
Available from
National Technical Information Service
U. S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

NTIS price codes
Printed copy: A03
Microfiche copy: A01

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Algorithms and FORTRAN Programs to Calculate Classical Collision Integrals for Realistic Intermolecular Potentials

William L. Taylor

Issued: November 29, 1979

DISCLAIMER

This book was prepared as an official work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service, or to any trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

MOUND FACILITY

Miamisburg, Ohio 45342

operated by

MONSANTO RESEARCH CORPORATION

a subsidiary of Monsanto Company

for the

U. S. DEPARTMENT OF ENERGY

Contract No. DE-AC04-76-DP00053

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Contents

	<u>Page</u>
ABSTRACT.	3
I. INTRODUCTION.	3
II. GENERAL FORMULAS.	4
III. NUMERICAL PROCEDURE	5
A. Boundary Curve.	5
B. Angle of Deflection	8
C. Cross Sections.	10
D. Collision Integrals	11
IV. DESCRIPTION OF PROGRAMS	13
V. INTERMOLECULAR POTENTIALS FOR WHICH PROGRAMS ARE APPLICABLE	16
VI. INPUT DATA.	17
VII. ACKNOWLEDGEMENT	19
REFERENCES.	19
APPENDIX A: ORBITING CONDITIONS.	21
APPENDIX B: LISTING OF PROGRAMS.	23
DISTRIBUTION.	45

Abstract

Numerical methods and computer programs are given to evaluate, for an arbitrary intermolecular potential, the classical transport collision integrals which appear in the kinetic theory of dilute gases. The method of Gaussian quadrature was employed to integrate the triple integral. A detailed discussion is given of the mathematics necessary to determine the boundaries of the individual integrations as well as a detailed analysis of errors introduced by the numerical procedures. Results for a recently published helium potential, the HFDHE2, are given in Appendix B.

I. Introduction

In the Chapman Enskog theory of a dilute gas, the transport properties of a gas are expressed in terms of a set of collision integrals [Hirschfelder, 1954]. These collision integrals are functions of the interaction potential between the molecules of gas and, except for very simple potentials, they must be evaluated numerically on a computer. With the large scientific computers available today, these integrals can be evaluated in a few seconds for a potential function of the Lennard-Jones type. The calculations, however, are rather complicated and programming them for the computer is still a major undertaking. To eliminate this obstacle a computer program has been written that will calculate the collision integrals for any reasonable and realistic intermolecular potential.

The program is designed to utilize an intermolecular potential with the general characteristics shown in Figure 1; namely, it is repulsive at small separation distances, is zero at $r = r_1$, has an attractive minimum at $r = r_2$, and has an inflection point at $r = r_3$. The following additional assumptions are also made: the potential has only one minimum and one inflection point in the attractive

region; it goes to zero as $-r^{-6}$ or faster as $r \rightarrow \infty$; its first and second derivatives are defined for all separation distances; and the potential and its first derivative are continuous.

Although these assumptions seem fairly reasonable, there are many potentials which do not satisfy one or more of them. For example, the modified Buckingham (Exp.-6) potential is undefined for small separation distances; the Morse (Exp.-Exp.) potential does not approach zero asymptotically as $-r^{-6}$; or, a potential that is

FIGURE 1 - General characteristics of a realistic intermolecular potential r_1 is the zero, r_2 is the minimum, and r_3 is the point of inflection

defined piecemeal may not have a continuous first derivative. Actually, the program will frequently work with little or no modification for a potential that does not satisfy some or all of these assumptions.

To explain how the program can be used for potentials not satisfying some of the above assumptions, a description of the general numerical methods and of the specific techniques of implementing these methods in the program is given. The main consideration in the selection of numerical methods was to use methods that give the greatest accuracy for a given amount of calculation [Smith, 1964]. Calculations for the Lennard-Jones potential were performed to determine the accuracy of the methods, and the error analysis was made for this model.

The program is written in FORTRAN IV language for the IBM 360 model 30. This is a very basic version of FORTRAN which lacks many of the features introduced into FORTRAN in recent years. Hence, it should be possible to run the program on any FORTRAN system currently in use with only minor modifications. The program, exclusive of the operating system, requires 31,000 bytes of storage. It could be broken into two parts to reduce this requirement.

II. General formulas

The collision integrals are defined by the equations [Hirschfelder, 1954]

$$\Omega^{(\ell, s)}(T^*) = \frac{1}{(s+1)!} \int_0^\infty e^{-x} x^{s+1} Q^{(\ell)}(xT^*) dx, \quad (1)$$

$$Q^{(\ell)}(xT^*) = \frac{2}{[1 - \frac{1+(-1)^\ell}{2(1+\ell)}]} \int_0^\infty (1 - \cos \ell \chi) b db, \quad (2)$$

and

$$\chi(E, b) = \pi - 2b \int_{r_m}^\infty \frac{dr}{r^2 F(r)} \quad (3)$$

where

$$F(r) = [1 - \frac{\phi(r)}{E} - b^2/r^2]^{\frac{1}{2}}$$

and χ is the angle of deflection in the center-of-mass system of two colliding particles where $\phi(r)$ is the interaction potential between them. The collision is characterized by the relative initial kinetic energy E and the impact parameter b ; the distance of closest approach, r_m , is related to b and E by the equation $F(r_m) = 0$. The cross section $Q^{(\ell)}$, is an integral over all possible angles of deflection at a given energy, $E = xT^*$. The collision integrals, $\Omega^{(\ell, s)}$, are a weighted average of all the cross sections where the weighting factor corresponds to a Maxwellian velocity distribution. The equations for the collision integrals are given in the reduced form which is commonly used in the expressions for the transport properties; that is, $\Omega^{(\ell, s)}$ and $Q^{(\ell)}$ are expressed in units of their corresponding values for a rigid sphere potential. $T^* = T/(\epsilon/k)$, where T is the temperature in degrees Kelvin, k is Boltzman's constant, and ϵ is a parameter having the dimensions of energy. E and $\phi(r)$ are also expressed in units of ϵ . The distances b , r_m , and r are expressed in units of a distance parameter σ . The choice of the parameters ϵ and σ is arbitrary; however, they are usually chosen as some characteristic energy and length of the potential. For a fixed energy b and r_m are related by the equation

$$Er_m^2 - r_m^2 \phi(r_m) - b^2 E = 0. \quad (4)$$

If b is the variable of integration in the integral for the cross sections, this transcendental equation must be solved numerically for r_m . This relation, however, can be used to change the variable of integration from b to r_m , and then b is given as a function of r_m by

$$b = r_m \sqrt{\frac{E - \phi(r_m)}{E}} \quad (5)$$

and the integral for the cross section is

$$I_q = \frac{1}{2E} \int [2(E - \phi) - r_m^2 \dot{\phi}] (1 - \cos \chi) r_m dr_m \quad (6)$$

Thus, by changing the variable of integration to r_m , the numerical solution of an equation is replaced by the calculation of a square root. Not only is this simpler, but it requires fewer evaluations of the potential function.

The limits of integration of the cross section integral with respect to b are 0 to ∞ . For $b=0$ the relation between b and r_m reduces to $E = \phi(r_m)$, which is the equation for the repulsive portion of the potential. For a given energy the solution of this equation is the lower limit of integration of I_q and is denoted by $r_E(E)$. It is shown in Appendix A that for energies less than a certain critical energy, E_c , r_m is discontinuous at a certain value of b , b_0 . The value of r_m as $b \rightarrow b_0$ from the right (i.e., $b > b_0$) is given for a fixed energy, by the equations

$$E - \phi_{\text{eff}} = 0 \text{ and } \dot{\phi}_{\text{eff}} = 0 \quad (7)$$

where ϕ_{eff} is the effective potential defined by

$$\phi_{\text{eff}} = \phi + b^2 E / r^2 \quad (8)$$

By eliminating b from Equations 7, they can be written as

$$2(E - \phi) - r \dot{\phi} = 0. \quad (9)$$

The solution of this equation is denoted by $r_O(E)$, and b_0 is obtained by substituting r_O into Eq. 5. The value of r_m as $b \rightarrow b_0$ from the left is given by

$$E - \phi_{\text{eff}}(r, b_0) = 0 \quad (10)$$

and is denoted by $r'_O(E)$. Thus, for $E \leq E_c$ the integral must be broken into two parts: (a) an integral over the interval $r_E(E)$ to $r'_O(E)$; and (b) an integral over $r_O(E)$ to ∞ . For $E > E_c$ the integral is over the interval $r_E(E)$ to ∞ .

The region of integration for the collision integrals can be represented as an area in the E - r_m plane bounded by the curve composed of $r_E(E)$, $r'_O(E)$ as shown in Figure 2. Each point within this region represents a collision characterized by (E, r_m) . Points on the boundary to the right of r_1 represent orbiting collisions (i.e., $\chi = -\infty$). The angle of deflection is defined at each point within this region by Eq. 3. The cross sections are integrations at constant energies over all values of r_m within the region, and the collision integrals are integrals of the cross sections over all energies.

III. Numerical procedure

A. boundary curve

The first step in the calculation of the boundary curve is the evaluation of the critical energy and the corresponding distance of closest approach, E_c and r_c . This distance of closest approach is the

FIGURE 2 - Region of integration for the collision integrals in the E - Y_m plane.

solution of the equation (see Appendix A)

$$3\phi' + r\phi'' = 0. \quad (11)$$

By considering the signs of ϕ' and ϕ'' , it can be shown that $r_c > r_3$ and that the left hand side of Eq. 11 is greater than zero for $r > r_c$. This information is used in the program to calculate r_c by a stepping procedure. After r_c has been found, the critical energy is given by

$$E_c = \frac{1}{2} r_c \phi'(r_c) + \phi(r_c).$$

The portion of the boundary curve denoted by $r_0(E)$ is the solution of Eq. 9; i.e.,

$$y(r) = 2(E - \phi) - r\phi' = 0. \quad (12)$$

A typical curve for $y(r)$ is shown in Figure 3. The Newton-Raphson formula [Hildebrand, 1956]

$$(r_0)_{i+1} = (r_0)_i - \frac{y(r_{0,i})}{y'(r_{0,i})} \quad (13)$$

is used to calculate the solution. The initial approximation to the solution, i.e., $(r_0)_0$, is obtained by a stepping procedure. To ensure that the correct solution is found, the iteration is checked at each step, and if $r_{0,i} < r_c$ the initial approximation is moved towards the solution and the iterative calculation is started over.

The portion of the boundary curve denoted by $r'_0(E)$ is the solution of Eq. 10 which, using Eq. 5 and 8, can be written as

$$u(r) = r^2(E - \phi) - b_0^2 E = 0 \quad (14)$$

where

$$b_0 = r_0 \sqrt{\frac{E - \phi(r_0)}{E}}.$$

For a given energy, r_0 is obtained from Eq. 9 and used to calculate b_0 , which is then substituted into Eq. 14. A typical curve for $u(r)$ is shown in Figure 4. The Newton-Raphson formula is also used to

FIGURE 3 - The function $y(r)$ which defines r_0 (see Eq. 12). The minimum in $y(r)$ occurs at $r = r_c$ for all energies.

FIGURE 4 - The function $u(r)$ which defines r'_0 (see Eq. 14).

calculate r'_0 . It has been found that r_1 is a good starting point for the calculation of the exact value of r'_0 for almost all energies. Each iteration is checked and if $r'_{0,i}$ lies outside the interval (r_1, r_c) , the initial approximation is modified and the calculation is started over.

The portion of the boundary curve denoted by $r_E(E)$ is the solution of the equation $E - \phi = 0$. Without making further assumptions about the repulsive portion of the potential, there is no simple approximation to this equation which will serve as a good starting point for all energies in the Newton-Raphson formula. A starting point for which the Newton-Raphson formula will always converge to the solution is a value of r such that $\phi(r) > E$. This starting point is found by decreasing r , starting with r_1 , until $\phi(r) > E$. The exact value is then calculated by the Newton-Raphson formula.

The Newton-Raphson method has been used to calculate r_0 , even though it results in the appearance of the second derivative of the potential, because of the rapid convergence of the method. If a potential for which the second derivative does not exist is encountered, a method such as the regula falsi, which does not give rise to a second derivative, can be used to calculate r_0 . The second derivative also appears in the defining equation for r_c . By the analysis presented in Appendix A, it is evident that points of discontinuity in the second derivative, for which the left hand side of Eq. 11 has the same sign, have no effect on the value of r_c . A point of discontinuity for which the left hand side of Eq. 11 has different signs represents the maximum in the boundary curve. Depending upon how close the points of discontinuity in the second derivative are to r_c , the regula falsi method may be able to calculate r_c . Discontinuities in the second derivative present no problem in the remainder of the calculation of the collision integrals.

B. angle of deflection

The integrand in the expression for the angle of deflection (Eq. 3) has a singularity at the lower limit (i.e., r_m). By changing the variable of integration to $v = r_m/r$ it can be shown that this singularity is of the order of $1/(1-v^2)^{1/2}$. The expression for χ in terms of v is

$$\chi = \pi - \frac{2b}{r_m} \int_0^1 \frac{f(v)}{(1-v^2)^{1/2}} dv, \quad (15)$$

where

$$f(v) = \left[\frac{(1-v^2)}{1 - \frac{bv^2}{r_m^2} - \frac{\phi(r_m/v)}{E}} \right]^{1/2}.$$

Applying L'Hospital's rule gives

$$\lim_{v \rightarrow 1} f(v) = \lim_{v \rightarrow 1} \left[\frac{2v}{\phi'_{\text{eff}}/E} \right]^{1/2} = \left[\frac{2E}{\phi'_{\text{eff}}(1)} \right]^{1/2} < \infty$$

if $\phi'_{\text{eff}}(1) \neq 0$. This condition is satisfied everywhere in the region of integration, except on the boundary curve for $r_m > r_1$ (see Figure 2).

The form of the integrand in Eq. 15 suggests the use of a Gaussian quadrature formula with a weight function of $(1-v^2)^{-1/2}$ to evaluate the integral for χ . The Gaussian quadrature formula with this weight function has an especially simple form [Hildebrand, 1956], namely

$$\int_0^1 \frac{f(v)}{(1-v^2)^{1/2}} dv = \frac{\pi}{2n} \sum_{j=1}^n f\left(\cos \frac{2j-1}{4n} \pi\right).$$

Letting $a_j = \cos \frac{2j-1}{4n} \pi$ and $f(v) = (1-v^2)^{1/2} g(v)$, where

$$g(v) = \frac{1}{\left[1 - b^2 v^2 / r_m^2 - \phi(r_m/v)\right]^{1/2}},$$

then

$$f(a_j) = (1-a_j^2)^{1/2} g(a_j) = a_{n+1-j} g(a_j),$$

and

$$\int_0^1 \frac{f(v)}{(1-v^2)^{1/2}} dv = \frac{\pi}{2n} \sum_{j=1}^n a_{n+1-j} g(a_j).$$

Noting that the coefficients in this sum form the sequence $a_n, a_{n-1}, \dots, a_2, a_1$ and the arguments of g form the sequence $a_1, a_2, \dots, a_{n-1}, a_n$, the sum can be written in the form, if n is even,

$$\sum_{j=1}^m \left[a_{k-j} g(a_j) + a_j g(a_{k-j}) \right]$$

FIGURE 5 - Quadrature points necessary to calculate χ to 0.05% for (E, r_m) in the "well" portion of the angle of deflection region of integration for the Lennard-Jones potential.

where $k = n+1$ and $m = n/2$. If n is odd then $m = (n+1)/2$ and the second term is omitted for $j=m$. Thus, the quadrature formula for χ is

$$\chi = \pi - \frac{b}{nr_m} \sum_{j=1}^m \left[a_{k-j} g(a_j) + a_j g(a_{k-j}) \right] \quad (16)$$

where n is the number of points in the quadrature formula, $k = n+1$, $a_j = \cos \frac{2j-1}{4n} \pi$, and $g(a_j)$ is defined above. If n is even, $m = n/2$, and if n is odd, $m = (n+1)/2$ and the second term is omitted for $j = m$.

For energies greater than $1.25E_0$ ten points are sufficient in the quadrature formula (16) to calculate the angle of deflection to better than 0.05% for all values of r_m . Considerably more points are necessary to calculate the angle of deflection accurately to 0.05% for orbiting energies. In Figure 5, the number of points required is shown for the "well" portion of the region of integration

(i.e., $E < E_c$ and $r_E < r_m < r'_0$) for the Lennard-Jones potential. The number of points increases both as the energy decreases and as r_m approaches r'_0 . In Figure 6 the number of points required for orbiting energies and $r_m > r_0$ is shown as a function of the angle of deflection. For $r_m = r_0$ the angle of deflection is $-\infty$ and it increases to zero as r_m increases. For this region, the number of points is not very sensitive to the energy, but it does increase as r_m approaches r_0 . For energies slightly larger than E_c , the angle of deflection as a function of the distance of closest approach has a large negative minimum, and more than ten points are required in the region of this minimum. This minimum occurs for a value of r_m slightly less than r_c , and the number of points given in Figure 5 for $E = 0.99E_c$ is adequate for these energies and $r_E < r_m < r_c$.

FIGURE 6 - Quadrature points necessary to calculate χ to 0.05% for (E, r_m) in the region $E < E_c$ and $r_m > r_0$.

For energies near E_c there is a loss of numerical precision when the angle of deflection is large in magnitude. There are differences of approximately 0.1% between values calculated with ten significant digits and ones calculated with seven significant digits. Because of the large number of angles of deflection which contribute to a given collision integral, this loss of precision has a negligible effect on the final results.

C. cross sections

Equation 6 is used for the numerical evaluation of the cross sections. For orbiting energies the integral from r_E to r'_0 is evaluated by the Gaussian

$$\int_{r_E}^{r'_0} F(r) dr_m = \sum_{j=1}^n A_j^n F(x_j^n) \quad (17)$$

where $F(r_m) = r_m[2(E-\phi)-r_m\phi'](1-\cos\chi)$, x_j^n and A_j^n are the Gaussian abscissas and weights for the interval $(-1, 1)$,

and $r_m = \frac{1}{2}(r'_0 - r_E)(x_j^n + 1) + r_E$. The integral from r_0 to ∞ is evaluated by the Gauss-Laguerre quadrature formula [Hildebrand, 1956]

$$\int_{r_0}^{\infty} F(r_m) dr_m = \alpha \sum_{j=1}^n B_j^n e y_j^n F(y_j^n) \quad (18)$$

where y_j^n and B_j^n are the Gauss-Laguerre abscissas and weights and $r_m = r_0 + \alpha y_j^n$. The scale factor α is selected such that the contribution to the integral for r_m greater than $r_0 + \alpha y_n^n$ is negligible. This condition will be satisfied if

$$\delta = b^2/r_m^2 - 1$$

is sufficiently small. From Eq. 5

$$\delta = -\phi/E,$$

and since r_m is large, ϕ can be approximated by $\phi = -a/r^6$, so that

$$\delta = a/r_m^6 E.$$

Substituting r_m from this equation into $r_m = r_0 + \alpha y_n^n$, the following equation is obtained for α

$$\alpha = \frac{(a/\delta E)^{1/6} - r_0}{y_n^n} \quad (19)$$

The optimum value of δ is that value which requires the least number of points in the quadrature formula 18. For the Lennard-Jones potential $\delta = 10^{-4}$ is very close to the optimum value for all energies, and this is the value used in the program.

For energies slightly larger than E_C , the angle of deflection as a function of the distance of closest approach has a large negative minimum value for r_m near r_C . This minimum causes the integrand in Eq. 6 to oscillate rapidly in this region. To concentrate the quadrature points in this region, the integral is broken at $r_m = r_C$ and Eq. 17 and 18, with $r'_0 = r_0 = r_C$, are used to evaluate the two parts. For energies greater than $2E_C$ the minimum value of the angle of deflection is small enough in magnitude so that there are no rapid oscillations in the integrand, and Eq. 18 (with r_0 replaced by r_E) is used to evaluate the cross sections.

Equation 19 is used to evaluate the scale factor. For large energies the derivation of Eq. 19 is invalid, but in practice it gives satisfactory values except for very high energies. The energies for which it does not give satisfactory values will normally lie outside the range of interest.

In Figure 7, the percent error in the evaluation of $Q^{(2)}$ in the region r_E to r'_0 is shown as a function of the number of points in the quadrature formula

(Eq. 18). The "exact" values were calculated by using 64 points in the quadrature formula. The infinite oscillations in the integrand of Eq. 17 in the region of r'_0 limit the accuracy of the quadrature formula, and this is illustrated by the approximately 0.1% error that still exists with 48 points. Except for the lowest energy, 40 points are sufficient to calculate $Q^{(2)}$ to better than 0.2%. The convergence of Eq. 17 for $\ell = 1, 3$, and 4 is better than that shown in Figure 7 for $\ell = 2$. For these cross sections, the error with 40 points is less than 0.1%. In fact, for $\ell = 1$ and 3, the error is less than 0.1% with 32 points. The error with 32 points for $\ell = 4$ is approximately the same as that for $\ell = 2$.

The region around r_0 for which the integrand in Eq. 18 has an infinite number of oscillations makes a negligible contribution to the integral. The difference between the value of the integral evaluated with 40 and 60 points is less than 0.02% for all energies and all values of ℓ . For both orbiting and nonorbiting energies, 28 points are sufficient in the quadrature formula, Eq. 18, to evaluate the integral to better than 0.1% for all values of ℓ .

D. collision integrals

The integrals in Eq. 1 are evaluated by the Gauss-Laguerre quadrature formula

$$\begin{aligned} & \int_0^\infty e^{-x} x^{s+1} Q^{(\ell)}(xT^*) dx \\ &= \sum_{j=1}^n B_j^n (y_j^n)^{s+1} Q^{(\ell)}(y_j^n T^*). \end{aligned} \quad (21)$$

FIGURE 7 - Percent error in evaluation of $Q(2)$ from r_E to r_0' as a function of the number of Gaussian quadrature points. The "true" value of $Q(2)_0$ is that obtained with 64 Gaussian quadrature points.

For each reduced temperature T^* , this formula uses the cross sections evaluated at the energies $E_j = y_j^n T^*$. If the collision integrals are calculated at k temperatures, then this formula requires cross sections evaluated at kn energies. Instead of evaluating the cross sections directly at these energies, a table of cross sections as a function of energy is generated and the cross sections at the required energies are found by interpolation. The table is generated for the energy range

$$E_{\min} = T^*_{\min} y_1^n \quad (22)$$

$$E_{\max} = T^*_{\max} y_n^n$$

where $T^*_{\min}$ and $T^*_{\max}$ are the minimum and maximum reduced temperatures, respectively, for which the collision integrals are calculated. The cross sections oscillate slightly in the region of E_c

[Hirschfelder, 1954], so the energies are broken into the three ranges where y_j^n and B_j^n are the Gauss-Laguerre abscissas and weights.

$$E_{\min} < E < E_c$$

$$E_c < E < 2E_c$$

$$2E_c < E < E_{\max}$$

The cross sections are calculated at equal logarithmic intervals with eight points per decade for the first and third ranges and ten points for the second one. Because of the simplicity with which it can be programmed, Aitken's method is used to obtain the interpolated cross sections required for Eq. 21 [Kopal, 1961]. It has been found that interpolation of $\log Q^{(\ell)}$ as a function of $\log E$ gives more accurate results than interpolation of $Q^{(\ell)}$ as a function of E . Using four points in Aitken's formula, the interpolated values are accurate to better than one part in

10^3 in almost all cases. Since the number of energies for which the accuracy is less than this is very small, the total error resulting from the use of interpolated cross sections in Eq. 21 is much less than one part in 10^3 .

Twenty-eight points in the quadrature formula (21) are sufficient to calculate the integral to better than 0.1% for all values of ℓ and s .

IV. Description of programs

The calculation of the collision integrals is broken into two programs. The first program, SCAN, calculates some characteristic values of the potential. These values are: the zero of the potential, r_1 ; the minimum and its magnitude, r_2 and $E_{\min}$; the inflection point, r_3 ; and the maximum orbiting energy and corresponding distance of closest approach, E_C and r_C . The program also checks the potential for the presence of a spurious maximum in the repulsive region. If a spurious maximum is found it calculates the following: the separation distance, $r_{\max}$, at which the maximum occurs; the magnitude of the potential at the maximum, $E_{\max}$; the location of the inflection point, $r_{\inf}$, which occurs between r_1 and $r_{\max}$; and the magnitude of the potential, $E_{\inf}$, at the inflection point. If a spurious maximum is not found, the program indicates the maximum energy for which it checked. Normally, this energy will be $10^5 E_{\min}$. If it is less than this, the question of a spurious maximum should be investigated further.

The method of calculating E_C and r_C has been described above. The remaining separation distances are calculated by

the Newton-Raphson method, or, if the Newton-Raphson method would introduce a third derivative of the potential, by the regula-falsi method. The notation used in the program follows that used here rather closely; for example, the zero of the potential, r_1 , is represented by the variable R1.

The second program, COLL, calculates the 16 collision integrals required to evaluate all the transport properties through the third approximation; namely, the (1,1), (1,2), (1,3), (1,4), (1,5), (1,6), (1,7), (2,2), (2,3), (2,4), (2,5), (2,6), (3,3), (3,4), (3,5), and (4,4) integrals. The first section of the program reads the input data necessary for the calculation. The first items read are a table of Gaussian abscissas and weights and a table of Gauss-Laguerre abscissas and weights. Extensive tables of abscissas and weights for Gauss type quadrature formulas are given by Stroud [1966]. It is recommended that 40 Gaussian and 28 Gauss-Laguerre points be used, and tables for these values are given in the reference. The next items read are a title card, potential parameters, and the following characteristic values of the potential: r_1 , r_3 , r_C , and $E_{\inf}$, where $E_{\inf}$ is an energy above which the potential is invalid. $E_{\inf}$ is used for potentials which have a spurious maximum. If the potential has no spurious maximum, $E_{\inf}$ is set equal to "infinity". The last item read is a table of temperatures (a maximum of 100) for which the collision integrals are to be calculated. An explanation of how to set up the input data is given in the next section.

After evaluating the constant a/δ (see Eq. 19), the program calculates the table of

cross sections. This section of the program is primarily concerned with setting up the energy grid as described in Section III-D. The cross sections themselves are calculated by the subprogram CROSS. The cross sections for energies greater than E_{inf} are set equal to their values for the largest energy in the table which is less than E_{inf} . If one wishes to use some other procedure in this situation, the FORTRAN instructions can be inserted between statement numbers 258 and 260. If one wishes to print the cross sections, the print instruction should be placed immediately after statement 265. The four element array Q contains the cross sections for the energy contained in the variable E.

After calculating the table of cross sections, the program evaluates the reduced collision integrals (Eq. 1 and 21). The program could be broken into two parts at this point so that it would fit on a small computer. The second part would begin at statement 600. The input to it would consist of the input to the first part plus the table of the logarithm of the cross sections as a function of the logarithm of the energy calculated in the first part. This table is contained in the arrays EL and QL. In the last section of the program the reduced collision integrals are printed and punched on cards.

Subroutine CROSS calculates the cross sections for $\ell = 1, 2, 3$, and 4 as a function of the energy. The limits of integration (i.e., r_E , r'_O , and r_O) are calculated by subroutine ORBIT. If the energy is less than $2E_C$ the integral from r_E to r'_O (or r_O) is evaluated by Eq. 17. The number of points used to evaluate the angle of deflection, which is

calculated by the subroutine CHIGM, is specified by the array M_{ij} . CROSS determines in which section of the "well" portion of the region of integration (see Figure 2) the point (E, r_m) for which the angle of deflection is to be evaluated lies, and then selects the appropriate element of the array.

The second part of CROSS calculates the integral from r_L to ∞ by Eq. 18, where $r_L = r_O$ if $E < E_C$, $r_L = r_C$ if $E_C < E < 2E_C$, and $r_L = r_E$ if $E > 2E_C$. The scale factor (see Eq. 19) in this part of the program is evaluated in statement 400. The variable M2 is used to specify the number of points in the integration for the angle of deflection. If the energy is greater than $5/4 E_C$, M2 is set equal to 15 for all values of r_m . If the energy is less than $5/4 E_C$, M2 is initially set equal to 65 and is then decreased as the magnitude of the angle of deflection decreases.

Subroutine CHIGM calculates the angle of deflection. This subroutine, which is a straightforward programming of Eq. 16, is self-explanatory.

Subroutine ORBIT calculates the boundary curve parameters r_E , r'_O , and r_O as a function of the energy. Since the parameters are calculated for the sequence of energies specified by COLL, the results for the preceding energy are used as initial approximations. On entry to the subroutine, the variable EOLD contains the preceding energy, and the variables RE, ROP, and RO contain the values for this energy. On exit from the subroutine, the variables RE, ROP, and RO contain the results for the energy E. For the first energy, indicated by setting EOLD equal to -1.0, the initial approximations given

in Section III-A are used. The program also resorts to these approximations if the iterations go astray, using the preceding results as initial approximations.

The iterative calculations used in ORBIT are required to converge to one part in 10^6 , which is a rather severe requirement. If the convergence is only one part in 10^4 , a message indicating "poor convergence" is printed and the program continues. It is not unusual for a few of these messages to arise. If the convergence is less than one part in 10^4 a message indicating "no convergence" is printed and the program stops. The energy and the parameter for which the calculation failed are printed with this message. The program stopping with this message would normally be caused by an error in the input data or by a potential which does not satisfy the stated conditions.

The intermolecular potential is introduced into the program by the addition of two subprograms. The first subprogram calculates the potential as a function of the separation distance and has the general form `FUNCTION PHI(R)`, where R is the separation distance and PHI is the value of the potential. The second subprogram calculates the potential and its first and second derivatives as a function of the separation distance. It has the general form `SUBROUTINE POTEN(R,V,DV,DDV)`, where R is the separation distance, V is the value of the potential, DV is the value of the first derivative, and DDV is the value of the second derivative. A FORTRAN manual should be consulted for an explanation of how to write these subprograms. An example of the subprograms is given in Appendix B. If the potential is a function of some parameters, the values of

the parameters, rather than being explicitly given in the subprograms, may be given symbolic names which are placed in common storage. The values of the parameters are then read as input data and hence can be changed by reading new data.

An intermolecular potential has associated with it the dimensions of energy and length. The standard practice in defining the collision integrals is to express the potential energy in units of an energy parameter, ϵ , and the separation distance in units of a distance parameter, σ . Program COLL follows this practice, so these two subprograms must use the units of ϵ and σ . The choice of ϵ and σ is arbitrary, but some characteristic energy and length of the potential, such as E_{min} and r_1 , are normally used.

Program COLL actually calculates reduced collision integrals as a function of a reduced temperature. The reduced collision integral is the collision integral divided by its value for a rigid sphere of diameter σ , and the reduced temperature is the temperature divided by ϵ/k , where k is Boltzmann's constant. These reduced quantities are the starred quantities used by Hirschfelder [1954], and the definition of ϵ and σ given above is equivalent to his definition of these symbols.

To specify the accuracy of the collision integrals calculated by the program, let us define a temperature by $T_C = E_C$, where E_C is the maximum orbiting energy in units of ϵ and T_C is in units of ϵ/k . Then for temperatures less than $2 T_C$ the error is less than 0.2%; for temperatures greater than $2 T_C$, the error is less than 0.1%; and for temperatures greater than $5 T_C$, it is less than 0.05%. The main

limitation on the accuracy is the infinite oscillation of the integrand for the cross section near the boundary curve. Unless special techniques are used to evaluate the cross section integral in this region, it is impossible to improve the accuracy for low temperatures.

V. Intermolecular potentials for which programs are applicable

The program is specifically designed for potentials that satisfy the conditions stated in the Introduction. It can, however, be used in many cases where the potential does not satisfy some of these conditions. To understand how to use the program for these potentials, we will review how the various assumptions are used and how they effect the calculations.

The collision integrals at a given temperature are sensitive to only a portion of the potential, so it makes sense to calculate them for a potential that is not defined for all separation distances. The situation is simple for a potential that is not defined for large repulsive energies. Suppose the potential is defined for all energies less than E_{inf} , and for sufficiently low temperatures the contribution to the integral in Eq. 1 for cross sections greater than E_{inf} is negligible. Thus, the collision integrals at low temperatures can be calculated for such a potential by ignoring the high energy cross sections. In program COLL the energy E_{inf} is input data, and the program defines, rather than calculates, the cross sections for energies greater than E_{inf} .

The calculation of the collision integrals for a potential that is not defined in the attractive region is much more complicated mathematically. Since the cross sections at all energies depend upon the entire attractive portion, there is no simple method of ignoring the attractive portion in the calculations. To use the program the potential must be defined for the entire attractive region. Since the calculations for a repulsive potential are valid only for temperatures for which the influence of the attractive region is negligible, this requirement can be met by joining a realistic attractive portion to the repulsive potential. Spline functions have been used for this purpose and have been handled successfully by the described programs.

In defining a potential by joining portions of potentials together by spline functions, discontinuities may occur in either the first or second derivative, or both, at the joining points. These derivatives are used to calculate the characteristic values of the potential (i.e., r_1 , r_2 , and r_3) and the boundary curve parameters (i.e., r_E , r_C , r_O , and r'_O) by the Newton-Raphson method. They also appear in the defining equations for some of these quantities. If a discontinuity does not lie too close to these quantities, the programs will be able to calculate them. If the program fails to calculate r_E , r_O , or r'_O for some energy, the value of the parameter can be moved away from the discontinuity by slightly changing the highest (for r_E) or lowest (for r_O and r'_O) temperature. Accurate values of r_1 , r_2 , r_3 , and r_C are not required so graphical methods could be used to evaluate them if necessary. The programs have been used for several potentials

which have discontinuities in the derivatives. Except for a few cases where it was necessary to change the lowest temperature, no problems were encountered in calculating any of these quantities.

The first derivative also occurs in the integrand of the cross section integral used in the program (see Eq. 6). Discontinuities in the integrand may affect the accuracy of the quadrature formulas. It will be necessary to check this by doing the calculations for two sets of Gauss and Gauss-Laguerre quadrature points. It will probably be sufficient to compare the collision integrals, but a more sensitive test would be to compare the cross sections.

The assumption that the potential goes as $-a/r^6$ or faster as $r \rightarrow \infty$ is used in the derivation of the scale factor in the Gauss-Laguerre quadrature formula for the cross sections (see Eq. 19). The effect of an error in the evaluation of the scale factor is that the Gauss-Laguerre quadrature formula is inaccurate. It can be determined whether the assumption (that the potential goes as $-a/r^6$ or faster) is affecting the results by comparing the cross sections, or collision integrals, evaluated with two different sets of Gauss-Laguerre points. If necessary, the program can be changed by removing statement number 400 from subroutine CROSS and inserting instructions to evaluate the scale factor using the correct asymptotic form. The example given in the next section illustrates how to make this change.

VI. Input data

The subprograms to calculate the potential function, although they are FORTRAN

programs, are conceptually a portion of the required input. The subprogram POTEN is used with program SCAN and both subprogram POTEN and subprogram PHI are used with program COLL. The general form of these subprograms is discussed in the preceding section and an example is given in Appendix B. If the values of the parameters in the potential function are explicitly given in the subprograms, then new subprograms must be written to change the values. To avoid this, the parameters may be given symbolic names and these symbolic names listed in a COMMON statement. The actual values of the parameters are read as input data.

The units of the input data are determined by the units used in the potential subprograms. There are no restrictions on the units used in subprogram POTEN when it is used with program SCAN. For program COLL energy and distance in the potential subprograms must be expressed in units of ϵ and σ , respectively. The input data for COLL is expressed in these units and the temperature unit is ϵ/k . Although the choice of ϵ and σ is arbitrary, in order that the magnitude of the results does not exceed the output specifications used in the program, their magnitude is restricted; σ must be of the order of 1 to $10^8 \text{\AA}$, and ϵ must be chosen so that the reduced temperatures lie between 10^{-2} and 10^4 .

Unless stated otherwise, all input data are specified by the programs to be punched as E type numbers. This means the number is punched in the form $x.xxxE+yy$, where the value of the number is $x.xxxx10^{yy}$. For example, 1.5×10^{-3} is punched as 1.5E-03. The units digit of the exponent must be punched in the last column of the field containing the number. For example, if the number is punched in the 10 column

field consisting of columns 11 through 20, then the units digit must be punched in column 20. Most FORTRAN systems will accept a number punched without an exponent even though the specification calls for one. Thus, in systems which allow this, 1.5×10^{-3} could be punched as .0015.

A. Input data for program SCAN

Each data set consists of two cards. The first card is a title card in which the user may punch any information he desires. The second card is divided into eight fields of ten columns each. An approximation to the zero of the potential is punched in the first field. The other seven fields are for the values of the potential parameters, if there are any. The order of the parameters is the same as their order in the COMMON statement in POTEN.

More than one set of data can be fed to the program. Two blank cards should be placed after the last set of data.

B. Input data for program COLL

A table of abscissas and weights for the Gauss quadrature formula and a table of abscissas and weights for the Gauss-Laguerre quadrature formula are the first items read. The first card in each of these tables contains the number of points in the quadrature formula; the units digit of the number must be in column 12. Subsequent cards in the tables contain one abscissa and weight punched in the first 20 columns and the second 20 columns, respectively. Because of the symmetry, the number of cards in the Gauss table is half the number of points in the quadrature formula. Also, the number of points

used in the Gauss quadrature formula must be even. The number of cards in the Gauss-Laguerre table is equal to the number of points in the quadrature formula.

The Gauss table is read first, and then the Gauss-Laguerre table. The set of data, which is read next, consists of the following cards:

1st card - title

2nd card - 1 in column 10 if a table of temperatures is to be read. The remaining 70 columns are divided into 7 fields of 10 columns each. The potential parameters, if any, are punched in these fields. The order of the parameters is the same as their order in the COMMON statement in the potential subprograms.

3rd card - The first 50 columns are divided into 5 fields of 10 columns each.

The following data, which are calculated by program SCAN, are punched on this card.

1st field - r_1 , zero of potential

2nd field - r_3 , inflection point of potential

3rd field - r_c , orbiting parameter

4th field - E_c , orbiting parameter

5th field - E_{inf} , energy above which potential is undefined. If the potential has no spurious maximum, then E_{inf} is set equal to "infinity". (Except for extremely high temperatures, $10^5 E_{min}$ is effectively infinity.)

If there is a 1 in column 10 of card 2 the following cards are also required.

4th card - number of temperatures in table is punched in this card with units digit in column 5. The maximum allowed value is 100.

5th and subsequent cards - These cards contain the table of temperatures.

The first 50 columns of each card are divided into 5 fields of 10 columns each, and one temperature is punched in each field. The number of cards is determined by the number of temperatures in the table. Temperatures in the table must be in increasing order.

More than one set of data can be fed into the program. The first set of data must include a table of temperatures; thus, there must be a 1 in column 10 of the second card of the first set of data. Subsequent sets of data need not contain a table of temperatures.

VII. Acknowledgement

I would like to acknowledge the considerable effort made by James M. Keller on this project prior to his illness and untimely death. Jim was in large part responsible for the numerical procedures employed in this method and he also did much of the programming. He unfortunately passed away before this manuscript could be completed.

References

Hildebrand, F. B., Introduction to Numerical Analysis, McGraw-Hill Book Company, Inc., 1956.

Hirschfelder, J. O., C. F. Curtiss and R. B. Bird, Molecular Theory of Gases and Liquids, John Wiley & Sons, 1954.

Kopal, Z., Numerical Analysis, John Wiley & Sons, 1961.

Smith, F. J., and R. J. Munn, J. Chem. Phys., 41, 3560 (1964).

Stroud, A. H., and D. Secrest, Gaussian Quadrature Formulas, Prentice-Hall, 1966.

Appendix A: orbiting conditions

In Eq. 3 of the body of the report χ is the angle of deflection in the center of a mass system of two particles interacting under the potential $\phi(r)$. The dynamics of this interaction can be studied by considering the one-dimensional problem of a particle moving under the effective potential

$$\phi_{\text{eff}} = \phi + E(b^2/r^2).$$

In Figure A-1, ϕ_{eff} is plotted for three values of b for the fixed energy E . Since the radial velocity between the two particles is proportional to $E - \phi_{\text{eff}}$, the closest the two particles can come to each other is the largest root of the equation $E - \phi_{\text{eff}} = 0$. This root is shown in the figure for $b = b_1$ and b_2 . For $b = b_0$, the system can reach r'_0 without passing through a region of negative relative radial velocity; however, at $r = r_0$ the relative radial velocity and radial acceleration are zero. Hence,

for this value of b , the system "sticks" at r_0 ; i.e., the particles orbit each other at a separation r_0 . Clearly, the angle of deflection in this case is infinite.

It is obvious from the figure that b_0 and r_0 are given by the two simultaneous equations

$$E - \phi_{\text{eff}} = 0 \text{ and } \phi'_{\text{eff}} = 0,$$

and that r'_0 is the other root of the equation

$$E - \phi_{\text{eff}}(r, b_0) = 0.$$

For $b > b_0$ the distance of closest approach is greater than r_0 , while for $b < b_0$ the distance of closest approach is less than r'_0 . Hence, the distance of closest approach as a function of the impact parameter is discontinuous at $b = b_0$, and distances of closest approach lying between r'_0 and r_0 are physically impossible.

FIGURE A-1 - Illustration of the three types of collisions that occur at the energy E , depending upon the value of the impact parameter b .

The question now arises whether the situation depicted in Figure A-1 is true for all energies. By eliminating b_0 from the defining equations, it is found that

$$r_0 \phi'(r_0) = 2[E - \phi(r_0)] > 0,$$

which implies that r_0 is greater than the minimum of the potential for all energies. This equation can be considered as giving the energy at which orbiting occurs as a function of r_0 ; then

$$\frac{dE}{dr_0} = \frac{1}{2} (3\phi' + r_0 \phi'').$$

Since ϕ goes to zero faster than $1/r^2$, there is a value of r_0 , say r_c , for which $dE/dr_0 = 0$. By considering the signs of ϕ' and ϕ'' , it is found that $dE/dr_0 > 0$ for $r_0 < r_c$ and $dE/dr_0 < 0$ for $r_0 > r_c$, so $r_0 = r_c$ corresponds to a maximum energy which is given by

$$E_c = \frac{1}{2} r_c \phi'(r_c) + \phi(r_c).$$

The orbiting situation depicted in Figure A-1 occurs only for energies less than this critical energy, E_c .

Appendix B:

listing of programs

The potential chosen for the example (HFDHE2) consists of a Hartree-Fock repulsion between closed shell systems, an empirical estimate of the correlation energy, and semiempirically determined dispersion coefficients, C_6 , C_8 , and C_{10} . The form of the potential is

$$V(r) = \epsilon V^*(x)$$

$$V^*(x) = A \exp(-\alpha x) - \frac{C_6}{x^6} + \frac{C_8}{x^8} + \frac{C_{10}}{x^{10}} F(x),$$

where

$$F(x) = \begin{cases} \exp\left(-\frac{D}{x} - 1\right)^2 & \text{for } x < D \\ 1 & \text{for } x \geq D \end{cases}$$

Reduced parameters for the HFDHE2 potential are: $A = 0.5448504(6)$; $\alpha = 13.353384$; $C_6 = 1.3732412$; $C_8 = 0.4253785$; $C_{10} = 0.178100$; and $D = 1.241314$. Not all figures displayed are significant but are shown to avoid round-off errors. The potential is one recently developed for helium and further details are discussed in the reference which is given on a comment card in the listing of SUBROUTINE POTEN. Listings of the programs and sample output follow.

```

1      SUBROUTINE POTEN(R,V,DV,DDV)
2      C      HFDHE2 POTENTIAL FOR HELIUM
3      C      REFERENCE: R. A. AZIZ, V. P. S. NAIN, J. S. CARLEY, W. L. TAYLOR,
4      C      AND G. T. MCCONVILLE, J. CHEM. PHYS. 70, 4330 (1979).
5      C      POTENTIAL PARAMETERS ARE: RM=2.9673, E/K=10.8
6      A=0.544850E+06
7      ALPHA =13.353384
8      C6=1.3732412
9      C8=.4253785
10     C10=.178100
11     D=1.241314
12     X=R
13     IF(X-7.6) 10,10,11
14     10 VR=A*EXP(-ALPHA*X)
15     GO TO 12
16     11 VR=0.
17     12 DVR=-ALPHA*VR
18     DDVR=ALPHA*ALPHA*VR
19     X2=X*X
20     X4=X2*X2
21     X6=X4*X2
22     VA=-(C6/X6+C8/(X6*X2)+C10/(X6*X4))
23     DVA=(6.*C6/(X6*X)+8.*C8/(X6*X2*X)+10.*C10/(X6*X4*X))
24     DDVA=-(42.*C6/(X6*X2)+72.*C8/(X6*X4)+110.*C10/(X6*X6))
25     IF(X-D) 1,2,2
26     1 FX=EXP(-(D/X-1)**2)
27     DFX=2.*FX*(D*D/(X2*X)-D/X2)
28     DDFX=4.*FX*(D**4/X6-2.*D**3/(X4*X)-D*D/(2.*X4)+D/(X2*X))
29     V=VR+VA*FX
30     DV=DVR+DVA*FX+VA*DFX
31     DDV=DDVR+FX*DDVA+2.*DFX*DVA+VA*DDFX
32     GO TO 99
33     2 V=VR+VA
34     DV=DVR+DVA
35     DDV=DDVR+DDVA
36     99 RETURN
37     END

```

```

1  C      PROGRAM SCAN
2  C      CALCULATES FOLLOWING CHARACTERISTIC PARAMETERS OF POTENTIAL
3  C      ZERO (R1), MINIMUM (R2 AND EMIN), INFLECTION POINT (R3).
4  C      MAXIMUM ORBITING ENERGY AND ASSOCIATED DISTANCE OF CLOSEST
5  C      APPROACH (EC AND RC)
6  C      IF POTENTIAL HAS A SPURIOUS MAXIMUM, CALCULATES MAXIMUM (EMAX
7  C      AND RMAX) AND INFLECTION POINT (EINF AND RINF)
8  C      USES SUBROUTINE POTEN
9  C      INPUT IS AN APPROXIMATION TO R1
10 C      OUTPUT CONSISTS OF ABOVE QUANTITIES
11      COMMON PARAM(7)
12      NR=1
13      NW=3
14      105 READ(NR,911)
15          READ(NR,900) R,PARAM
16          READ (NR,915) KWIT
17          R1=R
18          IF(R) 108,108,109
19      108 CALL EXIT
20      109 CONTINUE
21          WRITE(NW,912)
22          R=.1
23      703 CALL POTEN(R,V,DV,DDV)
24          WRITE(NW,913)R,V,DV,DDV
25      913 FORMAT(4(7X,E10.3))
26          IF(10.-R)701,701,702
27      702 IF(R-.0) 139,140,140
28      139 R=R+0.05
29          GO TO 703
30      140 IF (R-1.195) 137,141,141
31      137 R=R+0.01
32          GO TO 703
33      141 R=R+0.1
34          GO TO 703
35      701 CONTINUE
36          R=R1
37          WRITE(NW,901)
38          WRITE(NW,911)
39          WRITE(NW,905) PARAM
40      120 CALL POTEN(R,PSI,DPSI,DDPSI)
41          IF(PSI)130,135,135
42      130 IF(DPSI)155,180,180
43      135 IF(DDPSI)138,138,145
44      138 R=R+0.1*R

```

```

45 C      CALCULATE R1
46      145 CORR=-PSI/DPSI
47      IF (ABS (CORR/R)-.00001)200,200,150
48      150 X=R+CORR
49      IF (X)155,155,160
50      155 X=R-0.06*R
51      160 CALL POTEN(X,PSI,DPSI,DDPSI)
52      IF (DDPSI)170,170,165
53      165 CORR=0.5*CORR
54      GO TO 150
55      170 R=X
56      GO TO 145
57      180 IF (DDPSI)185,190,190
58      185 R=0.5*R
59      GO TO 120
60      190 R=0.75*R
61      GO TO 120
62      200 R1=R+CORR
63 C      CALCULATE R2
64      R=R1
65      X=R
66      CORR=0.5*R
67      GO TO 230
68      210 CORR=-DDPSI/DDPSI
69      IF (ABS (CORR/R)-.00001)275,275,215
70      215 X=R+CORR
71      IF (X-R1)220,220,230
72      220 X=R-0.06*(ABS (DPSI)/DPSI)*R
73      230 CALL POTEN(X,PSI,DPSI,DDPSI)
74      IF (DDPSI)235,235,240
75      235 CORR=0.5*CORR
76      GO TO 215
77      240 R=X
78      GO TO 210
79      275 R2=R
80      EMIN=PSI
81 C      CALCULATE R3
82      U=1.
83      X=R2
84      DX=R2-R1
85      312 X=X+DX
86      CALL POTEN(X,V,DV,DDV)
87      IF (U*DDV) 322,370,312
88      322 U=-U

```

```

89      DX=-.5*DX
90      IF (ABS(DX/X)-1.E-04) 370,370,312
91 370   R3=X
92   C   CALCULATE RC AND EC
93      U=1.
94      X=R3
95      DX=R3-R2
96 422   X=X+DX
97      CALL POTEN(X,V,DV,DDV)
98      Y=X*DDV+3.*DV
99      IF (U*Y) 432,470,422
100 432  U=-U
101      DX=-.5*DX
102      IF (ABS(DX/X)-1.E-05) 470,470,422
103 470   RC=X
104      EC=V+.5*RC*D.V
105      BC=RC*SQRT((EC-V)/EC)
106      WRITE(NW,906)
107      WRITE(NW,902) R1,R2,R3,EMIN,RC,EC
108   C   CHECK FOR A SPURIOUS MAXIMUM
109      X=-1.E+05*EMIN
110      DR=.01*R1
111      R=R1
112      DO 525 I=1,90
113      CALL POTEN(R,V,DV,DDV)
114      IF (DDV) 540,520,520
115 520   IF (V-X) 525,530,530
116 525   R=R-DR
117 530   EINF=0.0
118      RINF=0.0
119      EMAX=V
120      RMAX=R+DR
121      WRITE(NW,910) EMAX,RMAX
122      GO TO 600
123   C   POTENTIAL HAS A SPURIOUS MAXIMUM
124   C   CALCULATE RINF,EINF,RMAX,AND EMAX
125 540   Y=DDV
126      X=R
127      R=R+DR
128 545   CALL POTEN(R,V,DV,DDV)
129      CORR=DDV*(P-X)/(DDV-Y)
130      IF (ABS (CORR/R)-.0001) 560,550,550
131      Y=DDV
132      X=R

```

```

133      R=R-CORR
134      GO TO 545
135 560 RINF=R
136      EINF=V
137      DR=.1*DR
138 570 R=R-DR
139      CALL POTEN(R,V,DV,DDV)
140      IF(DV)570,580,580
141 580 CORR=DV/DDV
142      IF(ABS(CORR/R)-.0001)590,590,585
143 585 R=R-CORR
144      CALL POTEN(R,V,DV,DDV)
145      GO TO 580
146 590 RMAX=R
147      EMAX=V
148      WRITE(NW,903) RINF,EINF,RMAX,EMAX
149 600 R2=R2/R1
150      R3=R3/R1
151      RC=RC/R1
152      RINF=RINF/R1
153      RMAX=RMAX/R1
154      R1=1.0
155      EMIN=-EMIN
156 EC=EC/EMIN
157      EINF=EINF/EMIN
158      EMAX=EMAX/EMIN
159      EMIN=-1.0
160      WRITE(NW,904)
161      WRITE(NW,902) R1,R2,R3,EMIN,RC,EC
162      IF(RINF)650,655,650
163 650 WRITE(NW,903) RINF,EINF,RMAX,EMAX
164      GO TO 680
165 655 WRITE(NW,910) EMAX,RMAX
166 680 CONTINUE
167      IF(KWIT) 704,105,105
168 900 FORMAT(8E10.4)
169 901 FORMAT(1H1,25X15H PROGRAM SCAN )
170 902 FORMAT(4X,3HR1=E14.5,7H , R2=E12.5,7H R3=E12.5/
171      1 4X,5HEMIN=E12.5,7H RC=E12.5,7H EC=E12.5/)
172 903 FORMAT(3X,34H POTENTIAL HAS A SPURIOUS MAXIMUM /
173      1 4X,5HRINF=E12.5,7H EINF=E12.5/
174      2 4X,5HRMAX=E12.5,7H EMAX=E12.5)
175 904 FORMAT(////17X,32H RESULTS IN UNITS OF EMIN AND R1 )
176 905 FORMAT(25H POTENTIAL PARAMETERS /4E15.6/3E15.6)
177 906 FORMAT(////17X31H RESULTS IN UNITS USED IN POTEN
178 910 FORMAT(3X,35H POTENTIAL HAS NO SPURIOUS MAXIMUM
179      1 3X,16H FOR E LESS THAN E15.5/
180      2 3X,19H AND R GREATER THAN E12.5)
181 911 FORMAT(80H
182      1 )
183 912 FORMAT(1H1,////20X,33HINTERMOLECULAR POTENTIAL FUNCTION//,
184      125X,6HV(R) =//,14X,1HR,19X,4HV(R),15X,5HDV(R),14X,6HDDV(R)//)
185 915 FORMAT (I2)
186 704 STOP
187      END

```


PROGRAM SCAN
 450HEZ POTENTIAL FOR HELIUM
 POTENTIAL PARAMETERS
 0.000000E+00 0.000000E+00 0.000000E+00 0.000000E+00
 0.000000E+00 0.000000E+00 0.000000E+00

RESULTS IN UNITS USED IN POTEN
 P1= 0.98919E+00 R2= 0.10000E+01 R3= 0.11087E+01
 SIN=-0.99997E+00 RC= 0.11590E+01 EC= 0.92629E+00

POTENTIAL HAS NO SPURIOUS MAXIMUM
 FOR E LESS THAN 0.10336E+06
 AND R GREATER THAN 0.13337E+00

RESULTS IN UNITS OF EMIN AND R1
 R1= 0.10000E+01 R2= 0.11247E+01 R3= 0.12469E+01
 EMIN=-0.10000E+01 RC= 0.13034E+01 EC= 0.92631E+00

POTENTIAL HAS NO SPURIOUS MAXIMUM
 FOR E LESS THAN 0.10337E+06
 AND R GREATER THAN 0.14999E+00

```

1      PROGRAM COLL
2      C      CALCULATION OF CLASSICAL COLLISION INTEGRALS
3      C      USES SUBPROGRAMS CROSS,ORBIT,CHIGM,POTEN,PHI
4      DIMENSION Q(4),T(100),QL(100,4),EL(100),A(8),OMEGA(16),OMEGAS(100,
5      116),DIF(4)
6      COMMON PARAM(7)
7      COMMON U(50),W(50),UL(50),WL(50)
8      NX=1
9      NPUN=2
10     NZ=3
11     NF=1
12     READ (NF,901) N1
13     N=N1/2
14     IF(2*N-N1) 104,105,104
15     104 WRITE(NZ,907)
16     STOP
17     105 READ (NF,902) (U(I),W(I),I=1,N)
18     READ (NF,901) N2
19     READ (NF,902) (UL(I),WL(I),I=1,N2)
20     110 READ(NX,970,END=112)
21     C 110 READ(NX,970)
22     C READ(NX,971) J,PARAM
23     READ(NX,971,END=112) J,PARAM
24     READ (NX,904,END=112) R1,R3,RC,EC,EINF
25     C READ (NX,904) R1,R3,RC,EC,EINF
26     IF(R1) 112,112,113
27     112 CALL EXIT
28     113 IF(J) 115,120,115
29     115 READ (NX,903,END=112) NT
30     C 115 READ (NX,903) NT
31     READ (NX,904,END=112) (T(I),I=1,NT)
32     C READ (NX,904) (T(I),I=1,NT)
33     120 WRITE (NZ,953) R1,R3,RC,EC,EINF
34     KN=40
35     KX=49
36     C CONST. USED TO DETERMINE SCALE FACTOR FOR CROSS SECTION INTEGRAL
37     R34=4.*R3
38     CALL POTEN(R34,PISS,X,X)
39     CONST=PISS*((4.*R3)**6)*1.E+04
40     CONST=-CONST
41     C BEGIN CALCULATION OF TABLE OF CROSS SECTIONS
42     C MIN. AND MAX. ENERGIES
43     EMIN=T(1)*UL(1)/1.01
44     EMAX=T(NT)*UL(N2)*1.01

```

```

45 C      DL AND DU SPECIFY NUMBER OF CROSS SECTIONS/DECADE FOR ENERGY
46 C      RANGES EMIN TO EC AND 2*EC TO EMAX, RESPECTIVELY
47      DL=8.0/2.3026
48      DU=8.0/2.3026
49      EMINL=ALOG(EMIN)
50      EMAXL=ALOG(EMAX)
51      ECL=ALOG(EC)
52      EOLD=-1.0
53      K=1
54      K1=0
55      J=0
56      DE=DL*(ECL-EMINL)
57      IF(DE)205,205,210
58 205    ECL=EMINL-.69315
59      GO TO 355
60 210    K1=DE+.5
61      IF(K1-KN)220,220,215
62 215    K1=KN
63 220    DE=K1
64      DE=(ECL-EMINL)/DE
65      DE=EXP (DE)
66      E=EMIN
67 250    DO 275 I=K,K1
68      IF(E-EINF)255,255,250
69 255    CALL CROSS(E,Q,EC,RC,R1,EOLD,CONST,N1,N2)
70      GO TO 260
71 250    CONTINUE
72 260    DO 265 L=1,4
73      IF(Q(L))75,75,76
74      75    QL(I,L)=0.
75      GO TO 265
76      76    QL(I,L)=ALOG(Q(L))
77 265    CONTINUE
78      EL(I)=ALOG(E)
79      EOLD=E
80 275    E=DE*E
91      IF(E-1.5*EC)305,305,350
82 305    DE=((ECL+.69315)-EL(K1))/12.
83      E=EXP (EL(K1)+2.*DE)
84      DE=EXP (DE)
85      K=K1+1
86      K1=K1+10
87      GO TO 250
88 350    IF(J)355,355,400

```

```

99      355 DE=DU*(EMAXL-(ECL+.69315))
90      IF(DE)400,360,360
91      360 J=DE+.5
92      IF(J-KX)370,370,365
93      365 J=KX
94      370 DE=J
95      DE=(EMAXL-(ECL+.69315))/DE
96      DE=EXP (DE)
97      E=EXP (ECL+.69315)
98      K=K1+1
99      K1=K1+J+1
100     GO TO 250
101 C     END CALCULATION OF TABLE OF CROSS SECTIONS
102 C     BEGIN CALCULATION OF COLLISION INTEGRALS
103     400 CONTINUE
104     600 DO 750 I=1,NT
105         TS=T(I)
106         DO 610 L=1,16
107     610 OMEGA(L)=0.0
108         DO 700 K=1,N2
109             E=(TS*UL(K))
110 C         BEGIN INTERPOLATION
111 C         NUM IS NUMBER OF AITKEN INTERPOLATION POINTS
112             NUM=5
113             ISN=NUM/2
114             ITN=(NUM-1)/2
115             EXL=ALOG(E)
116             DO 625 JNT=1,K1
117                 IF(EXL-EL(JNT))630,625,625
118             625 CONTINUE
119             630 IS=JNT-ISN
120                 IT=JNT+ITN
121                 JNT=NUM
122             631 IF(IS)632,632,635
123             632 IS=IS+1
124                 JNT=JNT-1
125                 GO TO 631
126             635 IF(IT-K1)640,640,637
127             637 IT=IT-1
128                 JNT=JNT-1
129                 GO TO 635
130             640 LNT=JNT
131                 JNT=JNT-1
132                 DO 660 L=1,4

```

```

133      DO 642 J=1,LNT
134      KKK=IS+J-1
135      642 A(J)=QL(KKK,L)
136      DO 650 N=1,JNT
137      DO 650 J=N,JNT
138      JIS=J+IS
139      NIS=N+IS-1
140      650 A(J+1)=(A(N)*(EL(JIS)-EXL)-A(J+1)*(EL(NIS)-EXL))/(EL(JIS)-EL(NIS))
141      660 Q(L)=EXP (A(JNT+1))
142  C      END INTERPOLATION
143  C      CALCULATION OF THE (1.S) INTEGRALS
144      675 JS=2
145      DO 680 L=1,7
146      OMEGA(L)=OMEGA(L)+WL(K)*(UL(K)**JS)*Q(1)
147      680 JS=JS+1
148  C      CALCULATION OF THE (2.S) INTEGRALS
149      JS=3
150      DO 685 L=8,12
151      OMEGA(L)=OMEGA(L)+WL(K)*(UL(K)**JS)*Q(2)
152      685 JS=JS+1
153  C      CALCULATION OF THE (3.S) INTEGRALS
154      JS=4
155      DO 690 L=13,15
156      OMEGA(L)=OMEGA(L)+WL(K)*(UL(K)**JS)*Q(3)
157      690 JS=JS+1
158  C      CALCULATION OF THE (4,4) INTEGRAL
159      OMEGA(16)=OMEGA(16)+WL(K)*(UL(K)**5)*Q(4)
160      700 CONTINUE
161  C      DIVISION OF INTEGALS BY (S+1) FACTORIAL
162      JS=2
163      DO 725 J=1,15
164      N=1
165      DO 715 L=2,JS
166      715 N=L*N
167      E=N
168      OMEGAS(1,J)=OMEGA(J)/E
169      IF (I-7)725,718,720
170      718 JS=2
171      GO TO 725
172      720 IF (J-12)725,722,725
173      722 JS=3
174      725 JS=JS+1
175      OMEGAS(1,16)=OMEGA(16)/120.0
176      750 CONTINUE

```

```

177      WRITE(NZ,910)
178      WRITE(NZ,970)
179      WRITE(NZ,972) PARAM
180      WRITE(NZ,911) (J,J=1,7)
181      WRITE (NZ,912) (T(I),(OMEGAS(I,L),L=1,8),I=1,NT)
182      WRITE (NZ,930) (J,J=3,6),(I,I=3,5)
183      WRITE (NZ,912) (T(I),(OMEGAS(I,L),L=9,16),I=1,NT)
184      GO TO 110
185      901 FORMAT (10X12)
186      902 FORMAT (2E20.6)
187      903 FORMAT (3I5)
188      904 FORMAT (5E10.5)
189      907 FORMAT(//52H THE NUMBER OF GAUSS QUADRATURE POINTS MUST BE EVEN.)
190      910 FORMAT(1H1,20X20H COLLISION INTEGRALS )
191      911 FORMAT(////4X5HTSTAR,7(4X2H1,I1,3X),4X3H2,2/)
192      912 FORMAT(F9.2,8F10.5)
193      930 FORMAT(////4X5HTSTAR,4(4X2H2,I1,3X),
194      1      3(4X2H3,I1,3X),4X3H4,4/)
195      953 FORMAT(43H1      CALCULATION OF COLLISION INTEGRALS/5H R1=F6.3
196      1      ,5H R3=F6.3,5H RC=F6.3,5H EC=F6.3,7H EINF=E12.4//)
197      960 FORMAT(2X5HTSTAR,7(3X2H1,I1,3X),3X3H2,2)
198      961 FORMAT(F8.2,8F9.5)
199      962 FORMAT(2X5HTSTAR,4(3X2H2,I1,3X),3(3X2H3,I1,3X),3X3H4,4)
200      970 FORMAT(80H
201      1
202      971 FORMAT(I10,7E10.4)
203      972 FORMAT(25H      POTENTIAL PARAMETERS /4E15.6/3E15.6)
204      END

```

```

1      SUBROUTINE CROSS(E,Q,EC,RC,R1,EOLD,AL,N1,N2)
2  C    EVALUATION OF CLASSICAL CROSS SECTIONS FOR COLLISION INTEGRALS
3  C    TABLES OF GAUSSIAN ABSCISSAS AND WEIGHTS AND GAUSS-LAGUERRE
4  C    USES SUBPROGRAMS ORBIT,CHIGM,POTEN,PHI
5  C    ABSCISSAS AND WEIGHTS MUST BE IN COMMON STORAGE
6      DIMENSION Q(4),M(3,4)
7      COMMON PARAM(7)
8      COMMON U(50),W(50),UL(50),WL(50)
9      BF(R)=R*SQRT ((E-V)/E)
10     BDBF(R)=R*(2.0*(E-V)-R*DV)
11     NW=3
12     DO 66 I=1,4
13         66 M(1,I)=10
14         M(3,4)=100
15         M(2,1)=20
16         M(3,1)=25
17         M(2,2)=25
18         M(3,2)=35
19         M(2,3)=40
20         M(3,3)=45
21         M(2,4)=70
22     DO 150 L=1,4
23         150 Q(L)=0.0
24         K=1
25         IF(E-1.25*EC) 160,200,200
26         160 K=2
27         IF(E-.1*EC) 165,200,200
28         165 K=3
29     200 CALL ORBIT(E,RE,ROP,PO,EC,RC,R1,EOLD)
30     803 IF(E-2.0*EC) 210,210,375
31  C    EVALUATE INTEGRAL FROM RE TO ROP (OR RC IF E G.T. EC) BY
32  C    GAUSSIAN QUADRATURE
33     210 A=ROP-RE
34     RL=RO
35     N=N1/2
36     DO 303 J=1,2
37         DO 300 I=1,N
38         U(I)=-U(I)
39         X=0.5*(U(I)+1.0)
40         L=1
41         GO TO (225,215),J
42     215 L=2
43         IF(X-.9) 225,225,218
44     218 L=3

```

```

45      IF(X-.95)225,225,220
46      220 L=4
47      225 M1=M(K,L)
48      227 R=A*X+RE
49      CALL POTEN(R,V,DV,DDV)
50      B=BF(R)
51      CALL CHIGM(CHI,E,B,R,M1)
52      Y=COS (CHI)
53      Z=Y
54      BDB=BDBF(R)
55      DO 275 L=1,4
56      Q(L)=Q(L)+0.5*W(I)*BDB*(1.0-Y)
57      275 Y=Y*Z
58      300 CONTINUE
59      303 CONTINUE
60      DO 305 L=1,4
61      305 Q(L)=A*Q(L)
62      GO TO 400
63      375 RL=RE
64      C EVALUATE INTEGRAL FROM R0 (RC IF E G.T. EC, RE IF E G.T. 2*EC)
65      C TO INFINITY BY GAUSS-LAGUERRE QUADRATURE
66      400 SCALE=((AL/E)**.16667)-RL/UL(N2)
67      IF(SCALE) 403,403,404
68      403 WRITE(NW,900) E
69      STOP
70      404 K=1
71      M2=10
72      IF(E-1.25*EC)405,405,410
73      405 K=2
74      M2=40
75      410 DO 500 I=1,N2
76      X=UL(I)
77      R=X*SCALE+RL
78      CALL POTEN(R,V,DV,DDV)
79      B=BF(R)
80      CALL CHIGM(CHI,E,B,R,M2)
81      Y=COS (CHI)
82      Z=Y
83      BDB=SCALE*BDBF(R)*EXP (X)
84      DO 475 L=1,4
85      Q(L)=Q(L)+W(I)*BDB*(1.0-Y)
86      475 Y=Y*Z
87      GO TO (500,480),K
88      480 IF(CHI+3.142)500,500,482
89      482 M2=15
90      IF(CHI+1.571)500,500,484
91      484 M2=10
92      500 CONTINUE
93      DO 560 L=1,4
94      X=L
95      X=1.0-0.5*((1.0+(-1.0)**L)/(1.0+X))
96      560 Q(L)=Q(L)/(X*E)
97      580 RETURN
98      900 FORMAT(//52H THE APPROXIMATION USED TO EVALUATE THE SCALE FACTOR
99      1 /49H IN THE GAUSS-LAGUERRE QUADRATURE FORMULA FOR THE/
100      1 42H CROSS SECTIONS IS INVALID FOR THE ENERGY=E13.4)
101      END

```



```

1      SUBROUTINE ORBIT(E,RE,ROP,RO,EC,RC,R1,EOLD)
2  C    CALCULATION OF BOUNDARY CURVE PARAMETERS
3  C    USES SUBROUTINE POTEN
4      NW=3
5      ER1=1.0E-06
6      ER2=1.0E-06
7      ER3=1.0E-06
8      J1=0
9      J2=0
10     J3=0
11  C    BEGIN CALCULATION OF RE
12     BIAS=1.*ER1
13     102 R=R1
14     DR=R1/100.
15     IF (EOLD) 108, 108, 104
16     104 R=RE
17     DR=RE/100.
18     IF (E-EOLD) 115, 115, 108
19     108 DO 111 J4=1, 100
20     R=R-DR
21     CALL POTEN(R,V,DV,DDV)
22     IF (E-V) 112, 112, 109
23     109 IF (DDV) 115, 115, 111
24     111 CONTINUE
25     112 IF (DDV) 113, 113, 115
26     113 R=R+.5*DR
27     115 RE=R
28     117 CALL POTEN(RE,V,DV,DDV)
29     118 CORR=(E-V)/DV
30     J1=J1+1
31     120 X=RE+CORR
32     CALL POTEN(X,V,DV,DDV)
33     IF (DV) 123, 121, 121
34     121 CORR=.5*CORR
35     IF (EOLD) 120, 120, 122
36     122 EOLD=-1.0
37     GO TO 102
38     123 IF (ABS (CORR/RE)-ER1) 128, 128, 124
39     124 RE=X
40     IF (J1-30) 118, 125, 125
41     125 IF (ABS (CORR/RE)-100.0*ER1) 126, 126, 500
42     126 WRITE(NW,901) E
43     RE=X-CORR
44     BIAS=10.*ER1

```

```

45      128 RE=RE+CORR
46      IF(E-.5*EC) 129,135,135
47 C      BIAS RE TO INSURE THAT E IS GREATER THAN PHI(RE)
48      129 RE=RE+BIAS*RE
49      CALL POTEN(RE,V,DV,DDV)
50      IF(E-V) 129,135,135
51      135 IF(RE)500,500,136
52      136 IF(RE-(R1+.0001*R1))137,500,500
53 C      END CALCULATION OF RE
54      137 IF(E-EC)138,450,450
55 C      CALCULATE ORBITING PARAMETERS
56 C      BEGIN CALCULATION OF RO
57      138 IF(EOLD)140,140,200
58 C      INITIAL APPROXIMATION FOR RO
59      140 K=0
60      U=1.
61      X=RC
62      DX=RC
63      140 X=X+DX
64      CALL POTEN(X,V,DV,DDV)
65      Y=2.*(E-V)-X*D.V
66      IF(U*Y) 148,180,153
67      153 K=K+1
68      U=-U
69      DX=-.5*DX
70      IF(K-10) 148,180,180
71      180 RO=X
72      U=DX
73      200 R=RO
74 C      END INITIAL APPROXIMATION FOR RO
75      205 CALL POTEN(R,V,DV,DDV)
76      Y=2.0*(E-V)-R*D.V
77      DY=-3.0*D.V-R*DDV
78      CORR=-Y/DY
79      R=R+CORR
80      J2=J2+1
81      IF(R-RC)230,230,215
82      215 IF(ABS (CORR/R)-ER2)250,250,220
83      220 IF(J2-30)205,222,222
84      222 IF(ABS (CORR/R)-100.0*ER2)224,224,501
85      224 WRITE(NW,902) E
86      GO TO 250
87 C      MODIFY INITIAL APPROXIMATION
88      230 IF(EOLD)240,240,235

```

```

89 235 EOLD=-1.0
90 GO TO 140
91 240 RO=RO+U
92 J2=J2-1
93 GO TO 200
94 250 RO=R
95 C END CALCULATION OF RO, CALCULATE BO
96 CALL POTEN(RO,V,DV,DDV)
97 BO=RO*SQRT ((E-V)/E)
98 C BEGIN CALCULATION OF ROP
99 C INITIAL APPROXIMATION
100 IF (EOLD) 275, 275, 320
101 275 ROP=R1
102 IK=0
103 320 R=ROP
104 325 CALL POTEN(R,V,DV,DDV)
105 Y=R*R*(E-V)-BO*BO*E
106 DY=2.0*R*(E-V)-R*R*D.V
107 CORR=-Y/DY
108 J3=J3+1
109 X=R+CORR
110 IF (X-RC) 335, 340, 340
111 335 IF (X-R1) 340, 340, 375
112 C MODIFY INITIAL APPROXIMATION
113 340 IF (EOLD) 345, 345, 370
114 345 IF (IK) 365, 350, 365
115 350 IK=1
116 U=.05*(RC-R1)
117 ROP=RC
118 355 ROP=ROP-U
119 CALL POTEN(ROP,V,DV,DDV)
120 Y=ROP*ROP*(E-V)-BO*BO*E
121 IF (Y) 320, 320, 355
122 365 ROP=ROP+.05*U
123 GO TO 320
124 370 EOLD=-1.
125 GO TO 275
126 375 IF (ABS (CORR/R)-ER3) 400, 400, 380
127 380 R=X
128 IF (J3-30) 325, 385, 385
129 385 IF (ABS (CORR/R)-100.*ER3) 387, 387, 502
130 387 WRITE(NW,903) E
131 400 ROP=X
132 RETURN
133 450 RO=RC
134 ROP=RC
135 RETURN
136 500 WRITE(NW,904) E
137 GO TO 510
138 501 WRITE(NW,905) E
139 GO TO 510
140 502 WRITE(NW,906) E
141 RETURN
142 510 STOP
143 901 FORMAT(/38H POOR CONVERGENCE FOR RE, ENERGY= E14.5)
144 902 FORMAT(/38H POOR CONVERGENCE FOR RO, ENERGY= E14.5)
145 903 FORMAT(/38H POOR CONVERGENCE FOR ROP, ENERGY= E14.5)
146 904 FORMAT(/33H NO SOLUTION FOR RE, ENERGY= E14.5)
147 905 FORMAT(/33H NO SOLUTION FOR RO, ENERGY= E14.5)
148 906 FORMAT(/33H NO SOLUTION FOR ROP, ENERGY= E14.5)
149 END

```

```

1      SUBROUTINE CHIGM (CHI,E,B,RM,N)
2  C    EVALUATION OF ANGLE OF DEFLECTION BY GAUSS-MEHLER QUADRATURE
3  C    SUBROUTINE USES FUNCTION PHI(R)
4      K=N+1
5      PI=3.1415927
6      M=K/2
7      XL=4*N
8      JC=1
9      SUM=0.0
10     DO 200 J=1,M
11         AJ=(2*J-1)
12         AJ=COS(AJ*PI/XL)
13         AJN=(2*(K-J)-1)
14         AJN=COS(AJN*PI/XL)
15     158 U=1.0-(PHI(RM/AJ)/E)-(B*B*AJ*AJ/(RM*RM))
16         IF(U) 1,1,2
17         1 CONTINUE
18         U=0.
19         GO TO 3
20         2 U=AJN/SQRT(U)
21         3 SUM=SUM+U
22         IF(JC) 173,200,173
23     173 JC=0
24         IF(J-M) 176,174,176
25     174 IF(K-2*M) 176,200,176
26     176 U=AJN
27         AJN=AJ
28         AJ=U
29         GO TO 158
30     200 JC=1
31         U=N
32         CHI=PI*(1.0-B*SUM/(U*RM))
33         RETURN
34         END

```

```

1      FUNCTION PHI(R)
2  C    HFDHE2 POTENTIAL FOR HELIUM
3  C    REFERENCE: R. A. AZIZ, V. P. S. NAIN, J. S. CARLEY, W. L. TAYLOR,
4  C    AND G. T. MCCONVILLE, J. CHEM. PHYS. 70, 4330 (1979).
5  C    POTENTIAL PARAMETERS ARE: RM=2.9673, E/K=10.8
6      A=0.544850E+06
7      ALPHA =13.353384
8      C6=1.3732412
9      C8=.4253785
10     C10=.178100
11     D=1.241314
12     X=R
13     IF(X-7.6) 10,10,11
14 10   VR=A*EXP(-ALPHA*X)
15     GO TO 12
16 11   VR=0.
17 12   CONTINUE
18     X2=X*X
19     X4=X2*X2
20     X6=X4*X2
21     VA=-(C6/X6+C8/(X6*X2)+C10/(X6*X4))
22     IF(X-D) 1,2,2
23 1    FX=EXP(-(D/X-1)**2)
24     V=VR+VA*FX
25     PHI=V
26     GO TO 99
27 2    V=VR+VA
28     PHI=V
29 99   RETURN
30     END

```

COLLISION INTEGRALS

HFDHE2 POTENTIAL FOR HELIUM

POTENTIAL PARAMETERS

0.000000E+00 0.000000E+00 0.000000E+00 0.000000E+00
 0.000000E+00 0.000000E+00 0.000000E+00

TSTAR	1.1	1.2	1.3	1.4	1.5	1.6	1.7	2.2
0.10	2.87977	2.56881	2.35908	2.20239	2.07729	1.97229	1.88039	2.99264
0.20	2.28845	2.02255	1.82941	1.67225	1.53727	1.41961	1.31775	2.40878
0.30	1.96577	1.70236	1.50493	1.34812	1.22292	1.12403	1.04665	2.11933
0.40	1.74093	1.48043	1.29097	1.15036	1.04668	0.97040	0.91368	1.90597
0.50	1.56607	1.31531	1.14340	1.02405	0.94065	0.88129	0.83777	1.74243
0.60	1.43631	1.19898	1.04394	0.94148	0.87240	0.82409	0.78874	1.59618
0.70	1.32587	1.10650	0.96991	0.88257	0.82453	0.78395	0.75400	1.48589
0.80	1.24118	1.03714	0.91520	0.83923	0.78917	0.75400	0.72774	1.38922
0.90	1.17626	0.98446	0.87364	0.80611	0.76185	0.73055	0.70693	1.30627
1.00	1.11721	0.94005	0.84004	0.77962	0.73988	0.71149	0.68982	1.24066
1.10	1.06663	0.90318	0.81255	0.75792	0.72170	0.69555	0.67536	1.18625
1.20	1.02406	0.87253	0.78970	0.73974	0.70632	0.68191	0.66289	1.13926
1.30	0.98820	0.84678	0.77040	0.72422	0.69304	0.67003	0.65193	1.09832
1.40	0.95743	0.82480	0.75384	0.71076	0.68140	0.65953	0.64218	1.06259
1.50	0.93183	0.80604	0.73945	0.69893	0.67107	0.65013	0.63339	1.03119
1.60	0.90973	0.78970	0.72678	0.68839	0.66179	0.64162	0.62540	1.00358
1.70	0.88914	0.77504	0.71546	0.67891	0.65337	0.63386	0.61807	0.97936
1.80	0.87019	0.76183	0.70526	0.67032	0.64568	0.62673	0.61131	0.95820
1.90	0.85289	0.74989	0.69601	0.66245	0.63860	0.62013	0.60504	0.93942
2.00	0.83714	0.73906	0.68756	0.65522	0.63205	0.61400	0.59918	0.92253
2.20	0.80974	0.72013	0.67263	0.64230	0.62024	0.60288	0.58853	0.89330
2.40	0.78679	0.70409	0.65978	0.63102	0.60984	0.59302	0.57904	0.86878
2.60	0.76729	0.69027	0.64852	0.62102	0.60054	0.58416	0.57047	0.84789
2.80	0.75050	0.67817	0.63852	0.61204	0.59214	0.57612	0.56267	0.82988
3.00	0.73586	0.66745	0.62953	0.60390	0.58448	0.56875	0.55551	0.81420
3.20	0.72296	0.65784	0.62138	0.59646	0.57743	0.56196	0.54889	0.80039
3.40	0.71142	0.64915	0.61392	0.58960	0.57091	0.55566	0.54274	0.78811
3.60	0.70109	0.64122	0.60704	0.58324	0.56485	0.54978	0.53699	0.77710
3.80	0.69186	0.63396	0.60067	0.57732	0.55918	0.54427	0.53159	0.76716
4.00	0.68333	0.62723	0.59473	0.57177	0.55385	0.53909	0.52652	0.75812
4.20	0.67538	0.62097	0.58917	0.56655	0.54883	0.53420	0.52172	0.74985
4.40	0.66796	0.61512	0.58395	0.56163	0.54409	0.52957	0.51717	0.74226
4.60	0.66104	0.60964	0.57902	0.55698	0.53959	0.52518	0.51285	0.73524
4.80	0.65456	0.60447	0.57436	0.55255	0.53531	0.52099	0.50874	0.72872
5.00	0.64849	0.59959	0.56993	0.54835	0.53123	0.51700	0.50481	0.72262
5.50	0.63483	0.58847	0.55976	0.53864	0.52180	0.50776	0.49571	0.70896
6.00	0.62293	0.57861	0.55065	0.52991	0.51330	0.49941	0.48747	0.69709
6.50	0.61243	0.56976	0.54241	0.52198	0.50555	0.49180	0.47997	0.68661
7.00	0.60305	0.56173	0.53489	0.51471	0.49845	0.48481	0.47307	0.67724
7.50	0.59458	0.55438	0.52797	0.50801	0.49189	0.47835	0.46668	0.66877
8.00	0.58685	0.54761	0.52156	0.50179	0.48579	0.47235	0.46075	0.66104
8.50	0.57977	0.54133	0.51559	0.49599	0.48011	0.46674	0.45521	0.65394
9.00	0.57321	0.53548	0.51001	0.49056	0.47477	0.46148	0.45001	0.64736

9.50	0.56713	0.52999	0.50477	0.48546	0.46976	0.45653	0.44512	0.64124
10.00	0.56144	0.52484	0.49983	0.48064	0.46582	0.45186	0.44050	0.63551
11.00	0.54185	0.50685	0.48252	0.46371	0.44837	0.43543	0.42425	0.61571
14.00	0.52592	0.49200	0.46813	0.44962	0.43450	0.42173	0.41070	0.59951
15.00	0.51250	0.47935	0.45585	0.43758	0.42264	0.41002	0.39912	0.58578
16.00	0.50090	0.46835	0.44515	0.42708	0.41229	0.39980	0.38901	0.57386
20.00	0.49070	0.45863	0.43567	0.41777	0.40312	0.39075	0.38006	0.56333
25.00	0.46955	0.43836	0.41590	0.39835	0.38399	0.37186	0.36139	0.54137
30.00	0.45267	0.42212	0.40003	0.38277	0.36864	0.35672	0.34643	0.52370
35.00	0.43866	0.40860	0.38682	0.36980	0.35587	0.34413	0.33399	0.50894
40.00	0.42670	0.39704	0.37553	0.35871	0.34496	0.33337	0.32337	0.49627
45.00	0.41628	0.38696	0.36568	0.34905	0.33546	0.32400	0.31412	0.48518
50.00	0.40706	0.37803	0.35696	0.34050	0.32705	0.31572	0.30594	0.47533
60.00	0.39132	0.36280	0.34209	0.32593	0.31272	0.30160	0.29202	0.45844
70.00	0.37823	0.35013	0.32973	0.31381	0.30082	0.28989	0.28047	0.44432
80.00	0.36704	0.33930	0.31918	0.30348	0.29068	0.27991	0.27063	0.43220
90.00	0.35730	0.32988	0.30999	0.29449	0.28186	0.27123	0.26209	0.42160
100.00	0.34868	0.32154	0.30187	0.28655	0.27407	0.26357	0.25454	0.41220
150.00	0.31635	0.29033	0.27152	0.25689	0.24500	0.23501	0.22643	0.37671
200.00	0.29424	0.26904	0.25085	0.23673	0.22526	0.21564	0.20738	0.35224
250.00	0.27758	0.25302	0.23531	0.22159	0.21046	0.20113	0.19313	0.33370
300.00	0.26429	0.24025	0.22296	0.20956	0.19870	0.18961	0.18182	0.31884
350.00	0.25328	0.22969	0.21274	0.19963	0.18900	0.18011	0.17250	0.30649
400.00	0.24391	0.22072	0.20407	0.19120	0.18077	0.17206	0.16460	0.29595
450.00	0.23578	0.21294	0.19655	0.18390	0.17366	0.16510	0.15778	0.28678
500.00	0.22861	0.20608	0.18994	0.17747	0.16740	0.15898	0.15179	0.27867
550.00	0.22221	0.19997	0.18404	0.17175	0.16183	0.15354	0.14646	0.27142

TSTAP	2.3	2.4	2.5	2.6	3.3	3.4	3.5	4.4
0.10	2.75478	2.58074	2.44621	2.33864	2.72775	2.54839	2.40647	2.77109
0.20	2.22383	2.08718	1.97580	1.87694	2.14257	1.97856	1.84292	2.23752
0.30	1.94102	1.79445	1.66474	1.54778	1.81353	1.65493	1.52582	1.94050
0.40	1.71950	1.56356	1.43056	1.31880	1.59256	1.44066	1.32035	1.71138
0.50	1.54861	1.39193	1.26674	1.16859	1.42545	1.28553	1.17789	1.53668
0.60	1.40678	1.26106	1.15043	1.06731	1.30505	1.17486	1.07794	1.39746
0.70	1.30067	1.16540	1.06758	0.99659	1.20718	1.08958	1.00393	1.29103
0.80	1.21458	1.09189	1.00594	0.94482	1.13224	1.02464	0.94819	1.20619
0.90	1.14475	1.03451	0.95879	0.90550	1.07265	0.97373	0.90497	1.13857
1.00	1.08930	0.98924	0.92179	0.87463	1.02260	0.93241	0.87045	1.08419
1.10	1.04405	0.95267	0.89197	0.84968	0.98098	0.89851	0.84230	1.03954
1.20	1.00625	0.92254	0.86744	0.82906	0.94641	0.87041	0.81897	1.00237
1.30	0.97423	0.89726	0.84686	0.81166	0.91715	0.84669	0.79926	0.97100
1.40	0.94681	0.87575	0.82929	0.79671	0.89199	0.82637	0.78235	0.94424
1.50	0.92310	0.85720	0.81409	0.78368	0.87017	0.80876	0.76765	0.92120
1.60	0.90246	0.84105	0.80077	0.77219	0.85105	0.79336	0.75474	0.90119
1.70	0.88435	0.82683	0.78897	0.76193	0.83412	0.77972	0.74326	0.88360
1.80	0.86837	0.81421	0.77842	0.75269	0.81899	0.76755	0.73297	0.86804
1.90	0.85414	0.80289	0.76890	0.74429	0.80541	0.75660	0.72367	0.85414
2.00	0.84134	0.79267	0.76024	0.73661	0.79315	0.74668	0.71520	0.84167
2.20	0.81924	0.77488	0.74502	0.72299	0.77193	0.72938	0.70030	0.82012

2.40	0.80074	0.75984	0.73199	0.71118	0.75411	0.71470	0.68752	0.80212
2.60	0.78496	0.74686	0.72061	0.70078	0.73886	0.70202	0.67635	0.78678
2.80	0.77128	0.73548	0.71053	0.69149	0.72561	0.69089	0.66646	0.77350
3.00	0.75927	0.72538	0.70149	0.68309	0.71395	0.68099	0.65759	0.76184
3.20	0.74859	0.71630	0.69330	0.67542	0.70356	0.67210	0.64955	0.75148
3.40	0.73901	0.70808	0.68581	0.66838	0.69423	0.66403	0.64220	0.74218
3.60	0.73033	0.70055	0.67891	0.66185	0.68575	0.65665	0.63543	0.73376
3.80	0.72241	0.69362	0.67252	0.65577	0.67802	0.64985	0.62916	0.72607
4.00	0.71513	0.68719	0.66655	0.65009	0.67090	0.64356	0.62332	0.71900
4.20	0.70841	0.68121	0.66097	0.64474	0.66432	0.63769	0.61786	0.71246
4.40	0.70216	0.67561	0.65572	0.63970	0.65820	0.63221	0.61272	0.70638
4.60	0.69632	0.67035	0.65077	0.63492	0.65249	0.62705	0.60787	0.70070
4.80	0.69085	0.66538	0.64607	0.63039	0.64713	0.62219	0.60328	0.69537
5.00	0.68570	0.66068	0.64161	0.62608	0.64208	0.61759	0.59893	0.69034
5.50	0.67401	0.64992	0.63135	0.61611	0.63063	0.60706	0.58891	0.67894
6.00	0.66369	0.64033	0.62214	0.60713	0.62054	0.59769	0.57993	0.66886
6.50	0.65446	0.63167	0.61378	0.59896	0.61151	0.58924	0.57179	0.65983
7.00	0.64610	0.62377	0.60613	0.59145	0.60335	0.58154	0.56435	0.65165
7.50	0.63847	0.61651	0.59907	0.58452	0.59589	0.57447	0.55749	0.64416
8.00	0.63143	0.60979	0.59252	0.57807	0.58903	0.56793	0.55113	0.63725
8.50	0.62491	0.60354	0.58641	0.57205	0.58268	0.56185	0.54520	0.63084
9.00	0.61884	0.59769	0.58068	0.56640	0.57676	0.55616	0.53965	0.62487
9.50	0.61315	0.59219	0.57529	0.56107	0.57122	0.55082	0.53443	0.61926
10.00	0.60779	0.58700	0.57020	0.55604	0.56601	0.54579	0.52950	0.61398
12.00	0.58908	0.56878	0.55226	0.53829	0.54783	0.52815	0.51217	0.59549
14.00	0.57356	0.55359	0.53727	0.52344	0.53280	0.51347	0.49772	0.58013
16.00	0.56031	0.54058	0.52440	0.51068	0.51999	0.50092	0.48533	0.56697
18.00	0.54874	0.52919	0.51313	0.49951	0.50882	0.48995	0.47450	0.55547
20.00	0.53848	0.51907	0.50311	0.48957	0.49892	0.48022	0.46489	0.54525
25.00	0.51698	0.49785	0.48210	0.46873	0.47824	0.45986	0.44476	0.52380
30.00	0.49964	0.48071	0.46513	0.45191	0.46159	0.44345	0.42854	0.50646
35.00	0.48512	0.46637	0.45093	0.43783	0.44768	0.42974	0.41500	0.49192
40.00	0.47265	0.45405	0.43874	0.42575	0.43576	0.41798	0.40338	0.47942
45.00	0.46174	0.44327	0.42808	0.41519	0.42534	0.40771	0.39323	0.46847
50.00	0.45204	0.43370	0.41861	0.40582	0.41609	0.39860	0.38424	0.45874
60.00	0.43542	0.41730	0.40240	0.38978	0.40025	0.38300	0.36885	0.44203
70.00	0.42153	0.40360	0.38887	0.37641	0.38704	0.37000	0.35602	0.42805
80.00	0.40961	0.39186	0.37728	0.36495	0.37572	0.35887	0.34506	0.41605
90.00	0.39920	0.38161	0.36717	0.35496	0.36584	0.34916	0.33549	0.40557
100.00	0.38997	0.37252	0.35821	0.34612	0.35709	0.34056	0.32703	0.39626
150.00	0.35518	0.33832	0.32453	0.31289	0.32419	0.30828	0.29527	0.36116
200.00	0.33125	0.31484	0.30143	0.29014	0.30163	0.28617	0.27356	0.33698
250.00	0.31315	0.29710	0.28400	0.27297	0.28459	0.26950	0.25721	0.31867
300.00	0.29865	0.28291	0.27007	0.25927	0.27098	0.25620	0.24417	0.30401
350.00	0.28662	0.27114	0.25852	0.24791	0.25969	0.24518	0.23337	0.29183
400.00	0.27636	0.26110	0.24868	0.23824	0.25008	0.23580	0.22419	0.28144
450.00	0.26743	0.25238	0.24013	0.22984	0.24173	0.22766	0.21622	0.27240
500.00	0.25955	0.24469	0.23259	0.22243	0.23436	0.22048	0.20920	0.26442
550.00	0.25251	0.23781	0.22585	0.21581	0.22779	0.21407	0.20294	0.25728

Distribution

EXTERNAL

TIC, UC-34a (111)

H. N Hill, DOE, Dayton Area Office

R. K. Flitcraft, Monsanto Research Corporation

Monsanto Technical Library, St. Louis

W. J. Haubach, DOE, Basic Energy Sciences

INTERNAL

W. T. Cave

D. Cain

C. W. Huntington

J. R. McClain

G. T. McConville

W. M. Rutherford

W. L. Taylor (5)

R. E. Vallee

W. R. Wilkes

Publications

Library (15)