LEVEL II

12

A066043

AD A090516

(9) FINAL REPORT ON THE WORK DONE UNDER CONTRACT

(15) N 000 14-77-C-0102

(11) 1 Nov 79

(12) 236

(6) EXCIMER POTENTIAL CURVES

(10) H.S. Taylor
M. Valley
C. Jung
C. Watts
F. Bobrowicz

University of Southern California
Los Angeles, California, 90007

NR 395-570
Code 421

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government.

361550

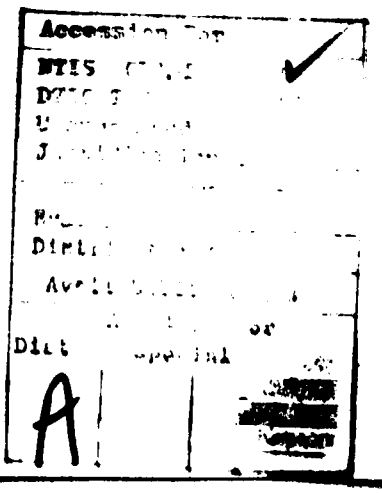
80 9 10 00

Summary

The purpose of this project is the development of semi-empir and theoretical methods of obtaining reasonably accurate potential curves for diatomic excimer systems. These curves are of importance for applications to fluorescence spectra, stimulated emission coefficients, excimer laser development and electron impact excitations. We have been mainly interested in the development of a new effective potential method which obtains potential curves by calculating the energy levels of one atom in the external field of the other atom. In order to be able to compare our new method to other well established methods, we have first done a CI calculation for GaKr and applied the Gordon-Kim method to GaKr. This provided some knowledge of the effect of a Kr atom on another atom which might be used later as input into the effective potential method.

The basic idea of the new method is as follows: In all low lying states of a metal-rare gas system one atom (the rare gas atom which has a high first excitation energy) is asymptotically in its ground state and the various molecular states correspond to the various low lying excited states of the metal atom. Therefore the basic assumption of our method is, that it is possible to calculate the molecular energy levels as the energy levels of a single metal atom in an external field which represents the rare gase atom. This external field which represents the second atom is our effective potential The general form of this potential can be derived from the many body Greens function. Phenomenological treatments of the many body theory leads to a form with free parameters. Our idea is to determine these parameters by fitting one molecular potential curve, calculated by our new method, to a known curve. We hope that

the effective potential resulting from one particular state describes the effect of the rare gas atom also in other molecular states and even in other molecules, i.e. independently of the type and state of the other atom. The big problem in this new method is, that new types of integrals have occured, which have not appeared in quantum chemistry before and for which no standard computer programs exist. Therefore, we spent most of the time developing and testing new computer programs. By now the programs seem to work properly but they are still very slow and at the moment we are still testing the whole method on a simple system, on LiHe. We have tried to parametrize the effective potential of the He ground state by fitting the resulting molecular potential for the $^2\pi$ state of HeLi to the known potential curve of this state. Probably we did not use the best fitting strategy so far and therefore we did not get a satisfactory fit of the molecular potential. It seems to be necessary to repeat the fitting procedure with a more efficient strategy, which we consider in the text. The next step in the program, after the fit of the molecular potential curve for one state, is to calculate the molecular potential curves for other states of HeLi using the same effective potential to represent the He atom. The result of this test will show if the whole idea works or not. We are continuing work on this idea even without the support of ARPA-ONR as we believe it to be a valuable new method.



§1

1.1 Introduction

The general purpose of this project is to continue the development and implementation of semi-empirical and theoretical methods of obtaining reasonably accurate potential curves for diatomic excimer systems.[1] Once the potential curves have been obtained, they can be used to predict fluorescence spectra[2], and to compute laser gain, stimulated emission coefficients[3] and total cross sections for excitation by electron impact.[4] While highly accurate potential curves are always a goal, our immediate goal is the relatively rapid production of potential curves that are accurate enough to guide experimentalists in choosing or rejecting possible laser systems on the basis of inexpensive theoretical calculations rather than on the basis of expensive and time-consuming experiments.

We are particularly interested in developing methods that in the future would be applicable to excimer systems because of the current emphasis on these systems as candidates for efficients, high-power visible and ultraviolet lasers.[1] The specific systems that motivated this proposal were the group IIA-,IIB- and IIIA-rare gas systems (such as MgXe and TlXe), alkali-group IIB systems and Thallium-group IIB systems. Our immediate goal is to develop a new efficient method to calculate potential curves for non-bounding diatomic systems in which the asymptotic atoms have greatly different ionization potentials (as do most metal-rare gas systems). We will then view the lowest several states of the molecule as roughly represented by a specific state of the metal in the effective "external" field of the high ionization potential (i.e. here a rare gas) atom.

The next step will be to parametrize this potential (and the effective electron potentials that arise therein) using a form suggested by field theory and data from any known molecular state of any metal-rare gas system that goes at large internuclear distance to the ground state of the rare gas. This effective potential will then be used to describe unknown states of the same or different metals by viewing their calculation as one of the metal atom in the field of this potential. If this aim is attained a basis empirical transfer of information from known to unknown systems and states will be achieved with an in principle accompaning simplification in calculations. This in turn would allow us to go to bigger systems.

Although extensive work has been done on calculating potential curves, the currently available methods are inadequate for large systems such as TlXe. While self-consistent field (SCF) and configuration interaction (CI) programs are the most accurate, at this time those computer codes cannot handle molecular systems with f electrons, such as TlXe.

Now, simplified methods, that are alternatives to CI do exist: The Gordon-Kim[5] (GK) method is available but has not been adequately tested for large systems with open shells. We believe that our method, theoretically is better based than this method, as the reader can judge for himself in later sections, and it takes in such effects as mutual atomic polarization that the GK leaves out.

The most promising method for treating big systems rigorously is the pseudopotential method. The pseudopotential method[6] is currently being extended to large systems.[7,8]

The new method we are developing, the effective potential method, is essentially a means of calculating the potential curve for a diatomic system AB (where A is a closed shell atom) by doing an atomic SCF or CI calculation on B in which A is treated as an external (effective) potential. The effective potential is the sum of the Hartree-Fock potential for A and a polarization potential. The polarization potential consists of a one-particle part describing the polarization of A by an external charged particle and a two-particle part which corrects for the "depolarization" due to the presence of the other electrons and nucleus of B. The parameter of the effective potential will be obtained by using perturbation theory and a known potential curve of AX (X is any atom). If X is not highly polarized, a first order approximation to the effective potential U^A may be obtained by fitting the parameters to

$$V^A_{X,e}(R) = \int d\tau\ U^A(R,\tau)\rho_X(\tau)$$

where ρ_X is the density of atom X and $V^A_{X,e}$ is the electronic part of the known potential curve for AX. The theory of the effective potential and the perturbation treatment of it are discussed in more detail in §4.

This effective potential method has several advantages over the Gordon-Kim and pseudopotential methods. Because the effective potential allows for polarization of A and the SCF calculation allows for polarization of B, this method should be more accurate than GK. At the same time, LS coupling and relativistic effects can be included as easily in our method: the treatment of LS coupling would be the same and relativistic effects can be included in the core potentials. In comparing our effective potential method to pseudopotential methods, the following points should be noted. Methods which use only a pseudopotential to replace the valence-core inter-

actions do not include polarization of the core. In contrast, our method includes core polarization; consequently, the core-core interaction is correct at large distances so the correct van der Waals interaction will be obtained. Because we take advantage of the fact that there is no real bonding in the systems we are studying, our calculations will be essentially atomic calculations and will include fewer electrons explicitly than a pseudopotential calculation. Use of a semi-empirical potential which is parameterized to molecular data should result in better molecular potential curves than use of semi-empirical potentials parameterized to atomic data. In addition, use of molecular data will allow us to obtain effective potentials for electrons outside of rare gas atoms and for electrons outside of highly charged cores; these are two cases in which parameterization to atomic data would be extremely difficult.

Since the effective potential method is semi-empirical and since the potential parameters will be fit to molecular data, some potential curves are needed. Basically, there are two options available for parameterizing the effective potential of A in order to calculate potential curves for AB: 1) The effective potential for A can be fit to known potential curves for XA (where X is any atom) or 2) The effective potential for A can be fit to whatever ground and excited state curves for AB are known. The second option recognizes the facts that it is easier to obtain ground state curves than excited state curves by other methods and that both are needed to pick possible laser systems. The curves needed for fitting can be obtained by a variety of techniques varying from rigorous CI calculations to analysis of experimental data. As examples of available techniques, consider the following: When experimental band spectra data exist, the quasi-static theory of line shapes could be used to extract potentials.[9] Atomic beam data (such as that supplied by Yuan Lee in Berkeley) could also be used to fit potential forms.[10] The RKR method[11] could be used when, as is rarely the case in excimers, vibrational data exists. For positive-negative ion systems Rittner potentials[12] can be fit to available data. For systems with closed-shell separated atoms and ions or for certain types of open-shell

systems, the Gordon-Kim method[5] can be used to predict short range potentials, and long range corrections can be added to this model. Short range repulsive potentials for closed-shell atoms and ions can also be fitted, correlated and interpolated using: (a) the delta-function model for homonuclear systems[10]; (b) the delta-function and distortion model with combination rules for heteronuclear systems[10]; (c) the related softness parameter methods that require atomic Hartree-Fock charge densities.[13]

Now after listing the advantages of our method we must admit to slow progress and a lack of success in so far completing our objectives due to two problems. The first is that our effective potential leads to new two center (atom-external potential) integrals such as those representing the interaction with the part of the effective potential representing the polarized rare gas (see later sections for details). We have developed methods and programs for treating these integrals but unfortunately the programs are still very consuming of computer time. These integrals have never appeared in quantum chemistry before and we had to develop new methods of of doing them. We do not believe our methods are presently optimum. Big savings in computer time can probably be achieved with a few years more efforts ("years" may seem long but not when one considers the thirty or so years it took to reach the state of the art in present molecular programs). Our strategic problem was that it was senseless to develop speedy packages to use in a method that might not work. Hence we used our slow packages and went on to the second problem, that of parametrizing the effective potential. In a sense the problems are linked, in that the trial and error fittings necessary to parametrize are severly limited in application by the computer cost for doing integrals as parameters are varied. The

second problem is therefore a so far less than ideal but perhaps not unacceptable fit of the parameters in the effective potential to the known curves. We will return to this problem in section 4.IV.

Now of the outset of our work we decided that we would like as input from theory some accurate GaKr curves so that we could get on effective potential for Kr to use in InKr or TlKr computations. Hence we split our work in three parts - the first part reviewed in §2 was to do accurate CI calculations on GaKr.

§3 is an application of the Gordon-Kim method to GaKr and a comparison with the results from §2.

The third part was to develop and test our theory in §4. As it turned out the third part is still not able to take advantage of the first part since even for Li-He we have not, because of the above two mentioned problems, achieved success. In any case we will report here our progress ommitting the large sections of efforts that lead to even inferior methods of integration and parametrization and reporting only the most promising methods. At the writing of this report we are close to a critical test calculation of LiHe, having achieved what may be a satisfactory (how satisfactory it is future results will show) calculation of the effective Helium potential. Time and personnel changes have prevented us from making this last step which we shall certainly do in the near future even without further support. If we are successful we shall return to the problems of speeding up our integral computation methods and routines. This will prepare the method to move on to bigger systems and get us to where we had hoped to be now, at the end of our contract period. Of course if the calculations are not successful theoretical

reasons need be sought and the promise of the method would be in question. While doing this work Dr. Taylor and Dr. Jung had a discussion about electron scattering in a strong laser field and wrote a paper about this subject. A copy is included in Appendix E. This work was not in the original grant but as many things in science it just happened in discussion. No computing fonds from the grant were used for this project.

References for §1

1. For a discussion of excimers and their relation to lasers see A. V. Phelps, JILA Report 110, "Tunable Gas Lasers Using Ground State Dissociation," (1972) and references therein.
2. This requires highly accurate curves as it depends on the derivative with respect to internuclear distance of the difference between the exicted and ground state potentials. As such we must consider these as being only semi-quantitative.
3. B. Cheron, R. Scheps and A. Gallagher, J. Chem. Phys. 65, 326 (1976).
4. We shall not discuss the methods of going from potential curves and wave functions to these properties as they are standardly given in textbooks. The quasistatic theory of line broadening is reviewed in JILA Report 110 [1].
5. V. I. Gaydaenko and V. K. Nikulin, Chem. Phys. Letters 7, 360 (1970); V. K. Nikulin, Zh. Tekh. Fiz. 41, 41 (1970) [English transl. Soviet Phys. Tech. Phys. 16, 28 (1971)]; R. G. Gordon and Y. S. Kim, J. Chem. Phys. 56, 3122 (1972); M. J. Clugston and R. G. Gordon, J. Chem. Phys. 66, 239 (1977).
6. C. F. Melius and W. A. Goddard III, Phys. Rev. A10, 1528 (1970); C. F. Melius, B. D. Olafson and W. A. Goddard III, Chem. Phys. Lett. 28, 457 (1974).
7. Y S. Lee, W. C. Ermler and K. S. Pitzer, J. Chem. Phys. 67, 5861 (1977).
8. W. R. Wadt, P. J. Hay and L. R. Kahn, J. Chem. Phys. 68, 1752 (1978).
9. R. Scheps, C. Ottinger, G. York and A. Gallagher, J. Chem. Phys. 63, 2581 (1975) and references therein on alkali-rare gas systems.
10. N. A. Sordergaard and E. A. Mason, J. Chem. Phys. 62, 1299 (1974) on rare gas-rare gas systems.
11. J. T. Vanderslice, E. A. Mason, W. G. Maisch and E. R. Lippincott, J. Mol. Spec. 3, 17 (1959); J. Chem. Phys. 32, 515 (1960).
12. E. S. Rittner, J. Chem. Phys. 19, 1030 (1951).
13. T. L. Gilbert, J. Chem. Phys. 63, 4061 (1975).

§2

Electronic States of GaKr : Ab initio calculations of a prototype for TlKr

2.I Introduction

Among the metal-rare gas eximers, the TlXe system is believed to be an excellent candidate for an efficient visible, high power, tunable laser.[1] Since ab initio calculations on this system are beyond the scope of present computer programs, we present here a configuration interaction (CI) calculation on GaKr , which is the largest group IIIB-rare gas system for which CI calculations can be done. We use the calculated GaKr curves to model the potential curves for InKr and TlKr . Although our model does not allow further extrapolation from TlKr to TlXe , it is hoped that these calculations will yield some insight into the properties of the TlXe eximer. In addition, Gallagher[2] has recently raised the possibility of using GaXe as a laser if the Ga can be obtained from dissociation of GaI_3 . Consequently the GaKr curves should also be of intrinsic interest.

In this paper, the CI calculations on GaKr are presented along with the model calculations on InKr and TlKr . These potential curves are used as the basis of a classical calculation of the emission and absorption coefficients for these systems.

2.II Details of the calculation

A. Basis set

The calculations use (14s11p6d) primitive Gaussian bases for gallium and krypton[3] as a starting point. The core orbitals (1s,2s,2p,3s,3p) are singly contracted to the Hartree-Fock atomic orbitals while the valence orbitals (4s and 4p) and the 3d orbitals are each described by two contracted functions(see Table I). The resulting (5s4p2d/5s4p2d) contracted bases are constructed using the general contraction scheme of Raffenetti.[4]

These basis sets are extended to include polarisation by adding two diffuse s functions (5s and 5s') and a diffuse p function (5p) to describe the lowest Rydburg orbitals. The exponents for these orbitals (ζ(5s) = .026, ζ(5s') = .011, ζ(5p) =.01) are obtained from atomic calculations on the excited states of Ga . The final basis set thus consists of a (16s12p6d) primitive basis contracted to [7s5p2d] for Ga and a (14s11p6d) primitive basis contracted to [5s4p2d] for Kr .

B. SCF calculation

The starting point for the CI calculation is a Hartree-Fock calculation on the $^2\Sigma^+$ state

$$\ldots\ 13\sigma^2 14\sigma^2 7\pi^4 15\sigma^2 16\sigma$$

The inner core molecular orbitals (twelve σ , twelve π and four δ) are held doubly occupied from this point on and are replaced by the rigorous nonlocal Hartree-Fock potential.

$$V_{core} = V_N + \sum_{i=core} (2J_i - K_i)$$

With the core orbitals removed from consideration, it is convenient to renumber the valence orbitals so that the Hartree-Fock configuration is written as

$$1\sigma^2 2\sigma^2 3\sigma^2 4\sigma 1\pi^4$$

At large R the correspondence for the valence orbitals is

$1\sigma \rightarrow$ 4sKr

$2\sigma \rightarrow$ 4pKr

$3\sigma \rightarrow$ 4sGa

$4\sigma \rightarrow$ 4pGa

$1\pi \rightarrow$ 4pKr

In addition to the valence orbitals, nine σ , six π and four δ virtual orbitals are used in the CI calculations. The lowest virtual orbitals (5σ, 6σ, 7σ, 2π, 3π), which correspond to the Ga 5s, 5s', 4p and 5p atomic orbitals for large internuclear separations, are obtained by the improved virtual orbital (IVO) procedure.[5] The IVO orbitals are obtained by removing the electron from the 4σ (valence) orbital of the above configuration and calculating the virtual orbitals for the (N-1)-electron Hamiltonian.

C. CI calculations

Full polarization CI (POL-CI) calculations[6] which provide a balanced description of all states of the 4s-4p manifold were carried out. A set of reference configurations was chosen (see Table II) to describe the dominant configurations for the $^2\Sigma^+$ and $^2\Pi$ states of the molecule and the $^1\Sigma^+$ state of the ion.

The full POL-CI calculations include all (1+2) electron excitations relative to each reference configuration subject to the restrictions that no more than one electron occupy the Rydberg 5σ orbital and no more than one electron occupy any virtual orbital (6σ, 2π etc.). This results in 764 spatial and 2314 spin configurations for the $^2\Sigma^+$ states, 556 spatial and 1565 spin configurations for the $^2\Pi$ states and 368 spatial and 558 spin configurations for the $^1\Sigma$ state. A total of 15σ , 14π and 2δ occupied and virtual orbitals are used in the POL-CI calculations.

The potential energy curves (Fig. 2 and Table III) from the POL-CI calculations generally follow the behavior predicted from these theoretical considerations. These calculations are not, however, designed to treat long-range dispersion forces. Many excitations which contribute to a $C_6 r^{-6}$ attraction are not included in the wavefunction. For these reasons, one would expect deeper wells in all the potential curves including the two lowest, generally repulsive, states. As predicted, the $1^2\Pi$ state is less repulsive than the $1^2\Sigma^+$ state. The bound excited state $?^2\Sigma^+$ has a minima at 6.36 a.u. about 3.0 eV above the ground state curve, while the minima in the ion curve is at 6.28 a.u. and 5.6 eV above the ground state. The well depths are compared with those obtained by Gallagher[11] in Table IV, and as expected, Gallagher's wells are deeper.

The dipole moments of these states and the transition moments between $^2\Sigma^+$ and the lower states have also been calculated and are given in Table V and Figs. 3 and 4.

B. Electronic states including spin-orbit coupling

A complete treatment of the electronic states of GaKr must include the effects of spin-orbit coupling. The states considered here, which dissociate to the closed-shell ground state of Kr and an open-shell state of Ga or Ga^+ are influenced only by the soin-orbit matrix elements of the open-shell atom.

Following the procedure used previously,[7,8,9,10] we have adopted a simple model for including the effects of spin-orbit coupling on the calculated potential energy curves and wavefunctions. The experimental spin-orbit parameters for the open-shell atom (Ga and later In , Tl) are used to determine the matrix elements of the spin-orbit interaction, H_{so} , coupling the molecular states at infinite separation and these matrix elements are assumed to be independent of internuclear distance. The resulting spin-orbit matrix H_{so} is added to the diagonal matrix of electronic energies $H_{elec}(R) = \delta_{ij}E_i(R)$:

$$H(R) = H_{elec}(R) + H_{so}$$

and the total matrix H is then diagonalized at each internuclear distance R . Thus, in addition to the assumption that the spin-orbit matrix elements do not change as functions of R , this model assumes that only one-center terms need be included so that only the spin-orbit coupling on Ga is important. We would expect this procedure to provide reasonable results to the extent that the molecular states retain the identity of the atomic

states from which they are formed. The spin-orbit matrices are given in Table VI along with the atomic parameters used in these calculations. The parameter λ is chosen so that the atomic $^2P_{3/2}$ and $^2P_{1/2}$ states have energies of $+\lambda$ and -2λ respectively.

We shall label the molecular states using the convention of Hund's case (c) where Ω, the projection of total angular momentum along the molecular axis, is the only good quantum number. Ω is defined as $\Lambda + S_z$, where Λ and S_z are the orbital and spin angular momentum projection, respectively. The molecular $^2\Sigma^+$ states have only a $\Omega = 1/2$ component, while the $^2\Pi$ states yield a $\Omega = 3/2$ and $\Omega = 1/2$ state. The states in the Ω representation are labeled according to increasing energy by a Roman numeral. So the $\Omega = 1/2$ states are designated as I 1/2, II 1/2 and the $\Omega = 3/2$ states are I 3/2, II 3/2.

The coupled states are expressed as follows:

$$|\mathrm{I}\ 1/2\rangle = C_\Sigma |1^2\Sigma^+\rangle + C_\Pi |1^2\Pi\rangle$$
$$|\mathrm{II}\ 1/2\rangle = -C_\Pi |1^2\Sigma^+\rangle + C_\Sigma |1^2\Pi\rangle$$
$$|\mathrm{I}\ 3/2\rangle = |1^2\Pi\rangle$$

The spin-orbit coefficient C_Σ and C_Π are given in Table VII. The potential curves and transition moments for GaKr including spin-orbit coupling are given in Table VIII and Figs. 5 and 6.

2.IV Extrapolations to InKr and TlKr

The potential curves for InKr and TlKr are modeled on the GaKr curves. The lowest excitation energies and the ionization potentials for the series Ga , In , Tl are given in Table IX. As can be seen, this series does not form a steady progression. In has a lower ionization potential and lower excitation energies than Ga , as expected for a heavier atom. However, Tl has a higher ionization potential and higher excitation energies. This is due in part to the presence of a filled 4f shell in Tl and the larger spin-orbit effects.[12] These effects should be considered when extrapolating the GaKr curves to InKr and TlKr.

To simulate InKr and TlKr , the experimental spin-orbit parameters for In and Tl are used to couple the GaKr curves. The curves are also shifted to give the correct atomic excitation energies at $R = \infty$ (see Table IX). This procedure should give at best a qualitative description of states of InKr and TlKr , since the non-spin-orbit coupled states are expected to have quantitatively different well depths and equilibrium separations.

The effect of the increasing spin-orbit perturbation in going from Ga to In and Tl is evident in the calculated curves which are given in Figs. 7 and 8. Only the well depths and positions for the I 1/2 and II 1/2 states are affected by the spin-orbit coupling. The other states

are the same as those for GaKr except that they have the correct asymptotic spacing.

The mixing parameters from the spin-orbit coupling calculations for TlKr are also used as the coefficients of the GaKr wave function to estimate the transition moments for this system. The TlKr transition moments are given in Table X and Fig. 9.

2.V Absorption and stimulated emission coefficients for possible laser transitions

The interest in the group IIIB-rare gas systems arises from the possibility of their use as visible laser systems. In order to judge their usefulness as lasers it is convenient to calculate the absorption $k_\nu(t)$ and stimulated emission $g_\nu(T)$ coefficients. Obtaining quantum-mechanical results for these quantities would require a complex calculation which would be inconsistent with the extrapolations used to obtain the InKr and TlKr curves. Consequently, we have used Gallagher's analysis [17], which is based on the classical Frank-Condon principle.

In order to obtain g_ν and k_ν, the CI curves are first fit by Morse potentials. The parameters for these Morse potentials are given in Table XI. These parameters can then be used in Gallagher's equations, along with the atomic transition rate, to obtain absorption and stimulated emission coefficients for pressure and excitation conditions of interest to experimentalists. We have calculated these coefficients for two different types of conditions. The high temperature results correspond to the case where the concentration of the metal is obtained from the vapor pressure of the metal itself, while the low temperature results correspond to obtaining the required concentration of the metal from vaporization of MI_3 (M = Ga, In, or Tl). This latter condition has been suggested by Gallagher as a possible means of obtaining high concentrations of the metal at low temperatures. In both cases the densities used are $10^{20}/cm^3$ for Kr, $10^{16}/cm^3 = 3[M\ ^2P_{1/2}] = 1.5[M\ ^2P_{3/2}]$ and $2 \times 10^{14}/cm^3 = [M\ ^2S_{1/2}]$. The resulting absorption and stimulated

emission coefficients for GaKr , InKr and TlKr are given in Figures 10 to 15.

A paper has been published about the work presented in §2. See Appendix A.

References for §2

1. B. Cheron, R. Scheps and A. Gallagher, J. Chem. Phys. 65, 326 (1976).
2. Gallagher, private communication.
3. T. H. Dunning, Jr., J. Chem. Phys. 66, 1382 (1977).
4. R. C. Raffenetti, J. Chem. Phys. 58, 4452 (1973).
5. W. J. Hunt and W. A. Goddard III, Chem. Phys. Lett. 3, 414 (1969).
6. P. J. Hay and T. H. Dunning, Jr., J. Chem. Phys. 64, 5077 (1976).
7. P. J. Hay and T. H. Dunning, Jr., J. Chem. Phys. 66, 1306 (1977).
8. P. J. Hay, T. H. Dunning, Jr. and Richard C. Raffenetti, J. Chem. Phys. 65, 2679 (1976).
9. D. C. Eckstrom, R. A. Gutcheck, R. M. Hill, D. Huestis, and D. C. Lorents, "Studies of E-Beam Pumped Molecular Lasers", Semiannual Report No. 2, Stanford Research Institute, Menlo Park, CA. (1973).
10. J. S. Cohen and B. I. Schneider, J. Chem. Phys. 61, 3230 (1974).
11. B. Cheron, R. Scheps and A. Gallagher, J. Chem. Phys. 65, 326 (1976).
12. P. Bagus, Y. S. Lee, K. S. Pitzer, Chem. Phys. Lett. 33 408 (1975) (discussion of Lanthanide contraction).

Table I. Gaussian exponents and contraction coefficients

Exponents	Contraction coefficients				
	Gallium atom				
	1s	2s	3s	4s	4s'
457600.	.000222	-.000069	.000026	-.000006	0.0
68470.	.001732	-.000535	.000205	-.000048	0.0
15590.	.008952	-.002814	.001070	-.000247	0.0
4450.	.035874	-.011275	.004337	-.001007	0.0
1472.	.114000	-.038495	.014707	-.003399	0.0
541.3	.274138	-.100714	.039748	-.009279	0.0
214.8	.414793	-.211832	.084475	-.019587	0.0
88.81	.275395	-.175448	.079654	-.019104	0.0
27.18	.029561	.479840	-.291821	.072753	0.0
11.54	-.006815	.634145	-.527118	.134137	0.0
3.303	.002253	.069592	.583707	-.181778	0.0
1.334	-.001017	-.012299	.674103	-.358241	0.0
.1947	.000251	.002774	.028077	.615164	0.0
.07158	0.0	0.0	0.0	0.0	1.0

	5s	5s'
.026	1.0	0.0
.011	0.0	1.0

	2p	3p	4p	4p'	5p
3274.	.001513	-.000576	.000094	0.0	0.0
765.4	.013070	-.004981	.000800	0.0	0.0
241.6	.067263	-.026421	.004337	0.0	0.0
89.39	.219542	-.089529	.014443	0.0	0.0
36.36	.421107	-.186734	.031377	0.0	0.0
15.60	.376515	-.144494	.021501	0.0	0.0
6.472	.089425	.258956	-.046233	0.0	0.0
2.748	-.000502	.570187	-.125293	0.0	0.0
1.090	.001761	.325305	-.045636	0.0	0.0
.2202	-.000247	.016563	.452811	0.0	0.0
.06130	0.0	0.0	0.0	1.0	0.0
.01	0.0	0.0	0.0	0.0	1.0

	3d	4d
59.66	.031949	0.0
17.10	.163546	0.0
6.030	.367457	0.0
2.171	.456851	0.0
.6844	.305161	0.0
.160	0.0	1.0

Table I. Gaussian exponents and contraction coefficients

Exponents	Contraction coefficients				
			Krypton atom		
	1s	2s	3s	4s	4s'
605700.	.000231	-.000073	.000029	-.000009	0.0
90300.	.001755	-.000551	.000221	-.000070	0.0
20920.	.009076	-.002894	.001159	-.000369	0.0
5889.	.036990	-.011834	.004781	-.001522	0.0
1950.	.116154	-.039826	.016056	-.005118	0.0
718.2	.278401	-.104801	.043454	-.013886	0.0
285.4	.415746	-.217093	.091899	-.029537	0.0
118.6	.267204	-.175562	.083789	-.027309	0.0
38.16	.027870	.471395	-.303023	.103498	0.0
16.45	-.005998	.636794	-.570620	.208810	0.0
5.211	.002217	.082255	.501751	-.235737	0.0
2.291	-.001092	-.014138	.760483	-.553570	0.0
.4837	.000306	.003289	.044857	.701123	0.0
.1855	0.0	0.0	0.0	0.0	1.0
	2p	3p	4p	4p'	
4678.	.001392	-.000569	.000156	0.0	
1120.	.011666	-.004777	.001286	0.0	
357.1	.060858	-.025631	.007059	0.0	
131.4	.210040	-.092159	.024990	0.0	
52.86	.421000	-.200936	.056870	0.0	
22.70	.383515	-.160784	.040225	0.0	
9.547	.097383	.267789	-.084756	0.0	
4.167	-.001087	.585908	-.240291	0.0	
1.811	.002209	.291397	-.038636	0.0	
.5337	-.000509	.015484	.599154	0.0	
.1654	0.0	0.0	0.0	1.0	
	3d	4d			
125.6	.019168	0.0			
33.31	.125638	0.0			
12.15	.366069	0.0			
4.350	.502482	0.0			
1.494	.264377	0.0			
.35	0.0	1.0			

Table II. Reference Configurations

$^2\Sigma^+$ states

1	$1\sigma^2 2\sigma^2 3\sigma^2 4\sigma 1\pi^4$
2	$1\sigma^2 2\sigma^2 3\sigma^2 5\sigma 1\pi^4$
3	$1\sigma^2 2\sigma^2 3\sigma^2 6\sigma 1\pi^4$

$^2\Pi$ states

1	$1\sigma^2 2\sigma^2 3\sigma^2 1\pi^4 2\pi$
2	$1\sigma^2 2\sigma^2 3\sigma^2 1\pi^4 3\pi$

$^1\Sigma^+$ state ($GaKr^+$)

1	$1\sigma^2 2\sigma^2 3\sigma^2 1\pi^4$

Table III. POL-CI calculations on the low-lying states of GaKr and the ground state of $GaKr^+$. All energies are relative to -4674. hartrees.

R	$1^2\Sigma^+$	$2^2\Sigma^+$	$1^2\Pi$	$1^1\Sigma^+$
∞	-1.200042	-1.095050	-1.198917	-0.992860
15.00	-1.200121	-1.095025	-1.198978	-0.993044
10.00	-1.200290	-1.094061	-1.199328	-0.994018
8.00	-1.198650	-1.094202	-1.200204	-0.996703
7.00	-1.193963	-1.095454	-1.200450	-0.999505
6.00	-1.179381	-1.096768	-1.197718	-1.002018
5.00	-1.135164	-1.088552	-1.179245	-0.992868
4.50	-1.086883	-1.064404	-1.148099	-0.967583
4.00	-1.018697	-0.992381	-1.079327	-0.906827
3.75	-0.968656	-0.933786	-1.025394	-0.856247

Table IV. Potential well depths

Molecule	State	CI		Morse Fit		Gallagher	
		R	ΔE(eV)	R	ΔE(eV)	R	ΔE(eV)
GaKr	$I_{1/2}(X_{1/2})$		∿.018	7.55	.021		
	$I_{3/2}(X_{3/2})$		∿.040	7.18	.040		
	$II_{1/2}(A_{1/2})$	∿10	.006	10.15	.00642		
	$III_{1/2}(B^2\Sigma_{1/2})$	6.36	.064	6.26	.080		
$GaKr^+$	I_o	6.28	.26	5.95	.252		
InKr	$I_{1/2}$	∿8	∿.013	7.86	.00778		
	$I_{3/2}$[a]						
	$II_{1/2}$	∿10	∿.006	9.98	.0064		
	$III_{1/2}$[a]						
	I_o[a]						
TlKr	$I_{1/2}$	∿8	∿.010	8.5	.012	7.01	.024
	$I_{3/2}$[a]					6.58	.062
	$II_{1/2}$	∿10	∿.006	9.88	.064		
	$III_{1/2}$[a]					6.09	.107
	I_o[a]						

[a] same as GaKr

Table V. Dipole and transition moments for the low-lying states of GaKr

R	$X^2\Pi$	$1^2\Sigma^+$	$2^2\Sigma^+$	$1^2\Sigma^+$-$X^2\Pi$	$2^2\Sigma^+$-$X^2\Pi$	$2^2\Sigma^+$-$1^2\Sigma^+$
15.00	0.00598	0.00156	0.07558	0.00183	-1.29167	-1.31011
10.00	0.02605	0.02048	0.56872	0.01631	-1.28976	-1.29762
8.00	0.09152	0.14168	0.89914	0.04761	-1.27767	-1.23098
7.00	0.18861	0.30443	0.95672	0.07894	-1.26770	-1.19024
6.00	0.41543	0.62092	0.81671	0.11663	-1.25536	-1.18926
5.00	0.90382	1.20144	0.45889	0.09916	-1.23487	-1.44020
4.50	1.24437	0.96841	1.19373	-0.13653	-1.20897	-2.06755
4.00	1.64453	-1.27575	5.47300	-0.84619	-0.86801	- .30554
3.75	1.88349	-1.13125	6.45764	-0.90367	-0.78769	.40946

Table VI. Quantities for spin-orbit matrices

$\Omega = 1/2$	$^2\Sigma^+$	$^2\Pi$
$^2\Sigma^+$	0	$\sqrt{2}\,\lambda$
$^2\Pi$	$\sqrt{2}\,\lambda$	$-\lambda$

$\Omega = 3/2$	$^2\Pi$
$^2\Pi$	$+\lambda$

Ga λ = .001255 au

In λ = .00336

Tl λ = .011835

Table VII. Spin-orbit coefficients for the $\Omega = 1/2$ states

	$R(a_o)$	C_Σ	C_Π
GaKr	3.75000	.99955	.02998
	4.00000	.99960	.02813
	4.50000	.99961	.02787
	5.00000	.99927	.03811
	6.00000	.99641	.08468
	7.00000	.98199	.18892
	8.00000	.93379	.35782
	10.00000	.82915	.55902
	15.00000	.81649	.57736
InKr	3.75000	.99704	.07690
	4.00000	.99738	.07238
	4.50000	.99742	.07175
	5.00000	.99534	.09644
	6.00000	.98063	.19588
	7.00000	.93712	.34900
	8.00000	.87677	.48091
	10.00000	.82132	.57047
	15.00000	.81649	.57736
TlKr	3.75000	.97505	.22197
	4.00000	.97731	.21180
	4.50000	.97763	.21034
	5.00000	.96502	.26219
	6.00000	.91737	.39803
	7.00000	.86679	.49867
	8.00000	.83612	.54854
	10.00000	.81788	.57540
	15.00000	.81649	.57736

Table VIII. Rydberg-valence transition moments in GaKr (with spin-orbit corrections)

	III 1/2 - I 1/2		III 1/2 - II 1/2		III 1/2 - I 3/2
R	Z	(X,Y)	Z	(X,Y)	(X,Y)
15.00	-0.75641	0.74574	-1.06969	-0.52733	-0.91335
10.00	-0.72540	0.75618	-1.07592	-0.50983	-0.91200
8.00	-0.44047	0.84363	-1.14948	-0.32327	-0.90345
7.00	-0.22486	0.88026	-1.16880	-0.16935	-0.89640
6.00	-0.10071	0.88449	-1.18499	-0.07517	-0.88767
5.00	-0.05489	0.87255	-1.43915	-0.03328	-0.87318
4.50	-0.05762	0.85454	-2.06674	-0.02383	-0.85487
4.00	-0.00859	0.61353	-0.30542	-0.01727	-0.61378

Table IX. Atomic states of Ga , In , and Tl

State	Excitation Energy					
	Ga		In		Tl	
	cm^{-1}	eV	cm^{-1}	eV	cm^{-1}	eV
$^2P_{1/2}$	0.0	0.0	0.0	0.0	0.0	0.0
$^2P_{3/2}$	826.24	.10241	2212.56	.274228	7792.7	.965840
$^1S_{1/2}$	24788.58	3.07234	24372.87	3.020814	26477.5	3.281665
I.P.	48380.	5.9963	46669.93	5.784348	49264.2	6.105886

Table X. Rydberg-to-valence transition moments in model TlKr (with spin-orbit corrections)

	III 1/2 - I 1/2		III 1/2 - II 1/2		III 1/2 - I 3/2
R	Z	(X,Y)	Z	(X,Y)	(X,Y)
15.00	-0.75641	0.74574	-1.06969	-0.52733	-0.91335
10.00	-0.74665	0.74590	-1.06130	-0.52476	-0.91200
8.00	-0.67524	0.75539	-1.02925	-0.49558	-0.90345
7.00	-0.59354	0.77699	-1.03169	-0.44701	-0.89640
6.00	-0.47336	0.81433	-1.09099	-0.35332	-0.88767
5.00	-0.37761	0.84264	-1.38982	-0.22894	-0.87318
4.50	-0.43489	0.83575	-2.02130	-0.17981	-0.85487
4.00	-0.06471	0.59985	-0.29861	-0.13000	-0.61378

Table XI. Morse fitting parameters:

$V(R) - V(\infty) = D_e[u^2 - 2u]$ where $u = \exp[B(R_e - R)]$

Molecule	State	D_e	$R_e(a_o)$	$B(a_o^{-1})$
GaKr	$I_{1/2}$	.000772	7.56	.724
	$I_{3/2}$	.00147	7.19	.722
	$II_{1/2}$	.000625	10.14	.471
	$III_{1/2}$	.00295	6.26	.853
InKr	$I_{1/2}$	.000286	7.86	.817
	$I_{3/2}$[a]			
	$II_{1/2}$	.000474	10.24	.483
	$III_{1/2}$[a]			
TlKr	$I_{1/2}$	.00439	8.52	.633
	$I_{3/2}$[a]			
	$II_{1/2}$	.000408	10.12	.502
	$III_{1/2}$[a]			

[a]Same as for GaKr

Fig. 1. Orbital diagrams for the electronic states of Ga + Kr and Ga^+ + Kr. The lobes and circles represent the in-plane and out-of-plane p orbitals; the dashed circle denotes the Rydberg orbital.

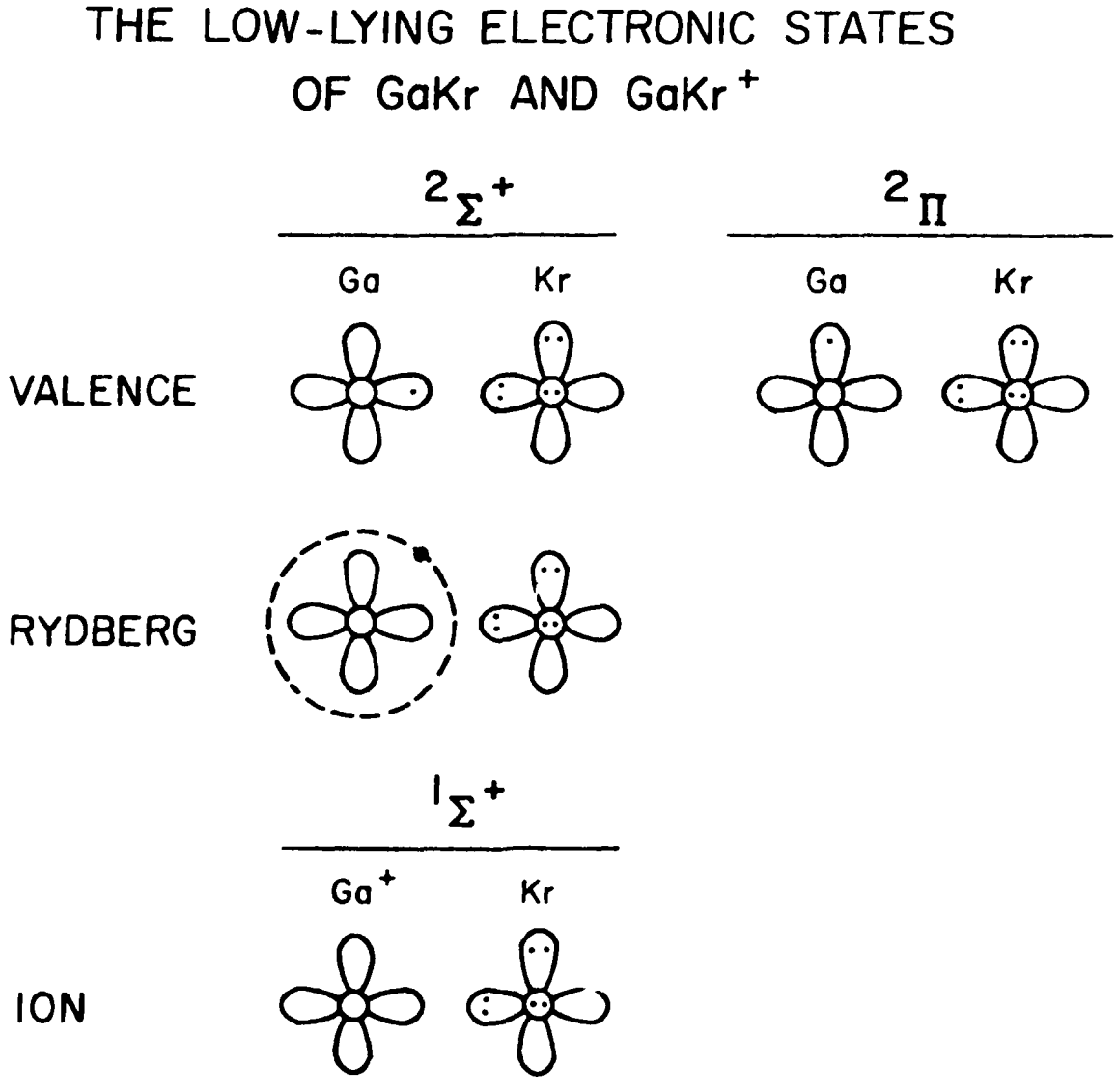


Figure 2.

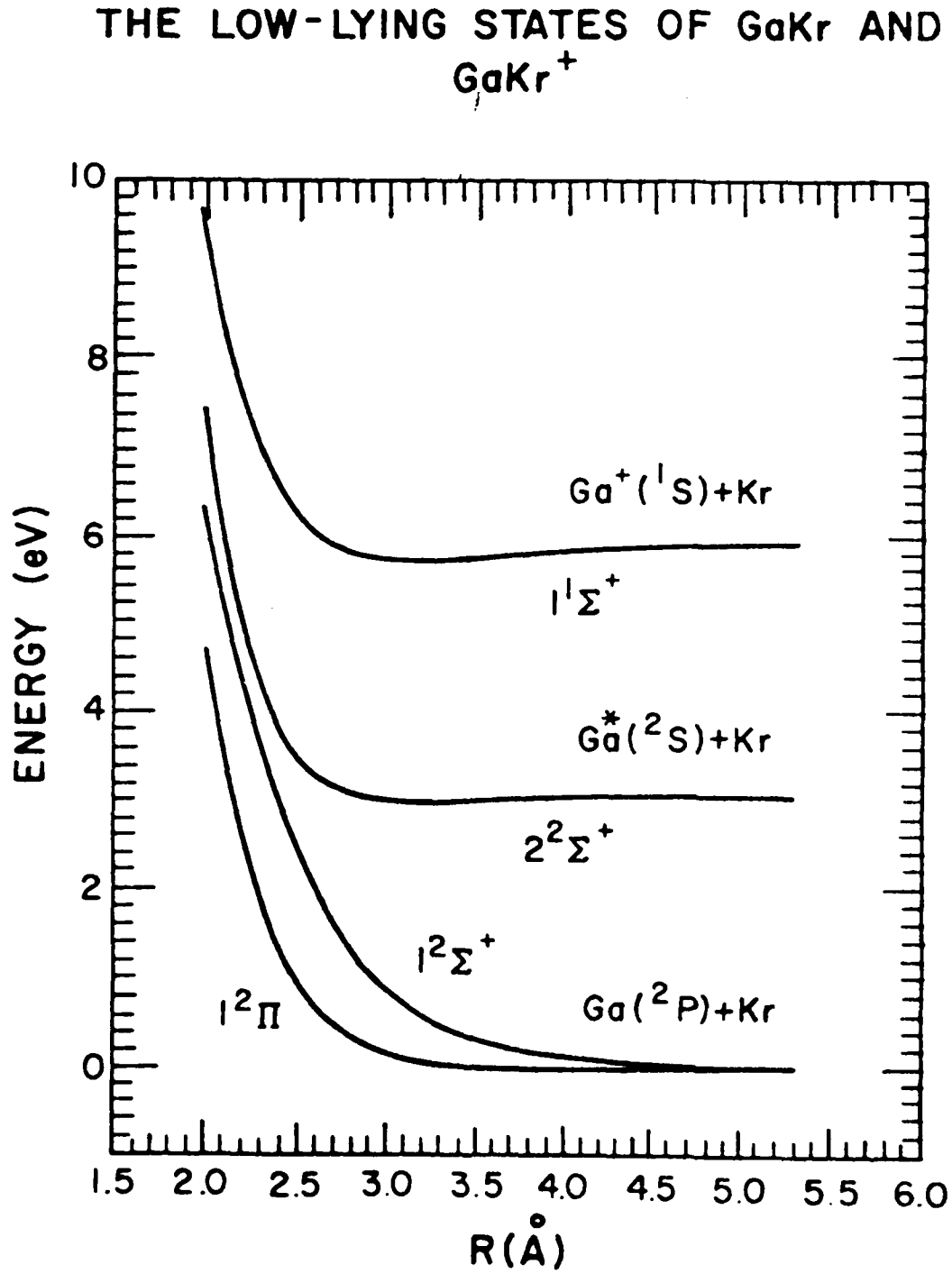

THE LOW-LYING STATES OF GaKr AND GaKr+
ENERGY (eV)
10
8
6
4
2
0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
R(Å)
Ga+(1S)+Kr
1 1Σ+
Ga*(2S)+Kr
2 2Σ+
1 2Σ+
1 2Π
Ga(2P)+Kr

Figure 3.

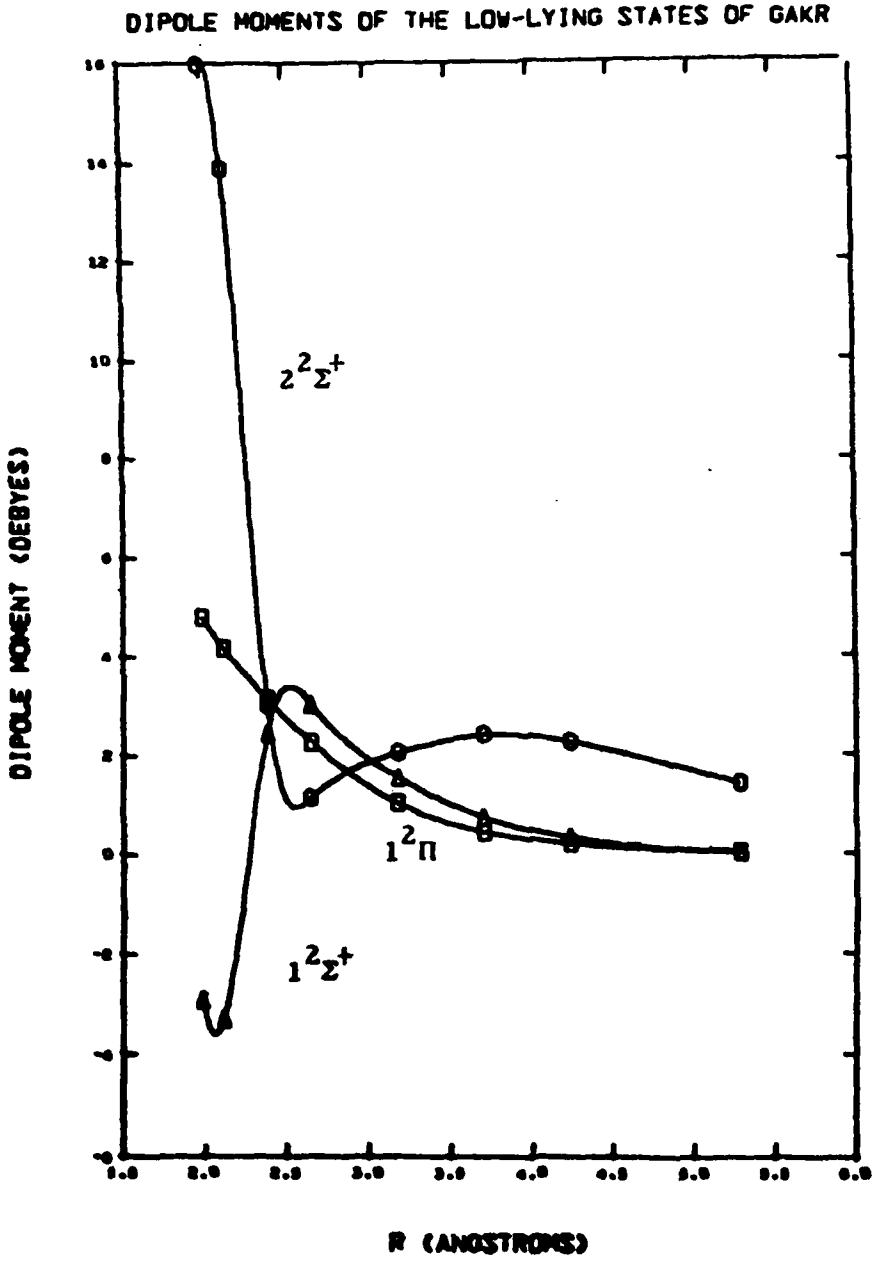

DIPOLE MOMENTS OF THE LOW-LYING STATES OF GAKR
DIPOLE MOMENT (DEBYES)
R (ANGSTROMS)
$2^2\Sigma^+$
$1^2\Pi$
$1^2\Sigma^+$

Figure 4.

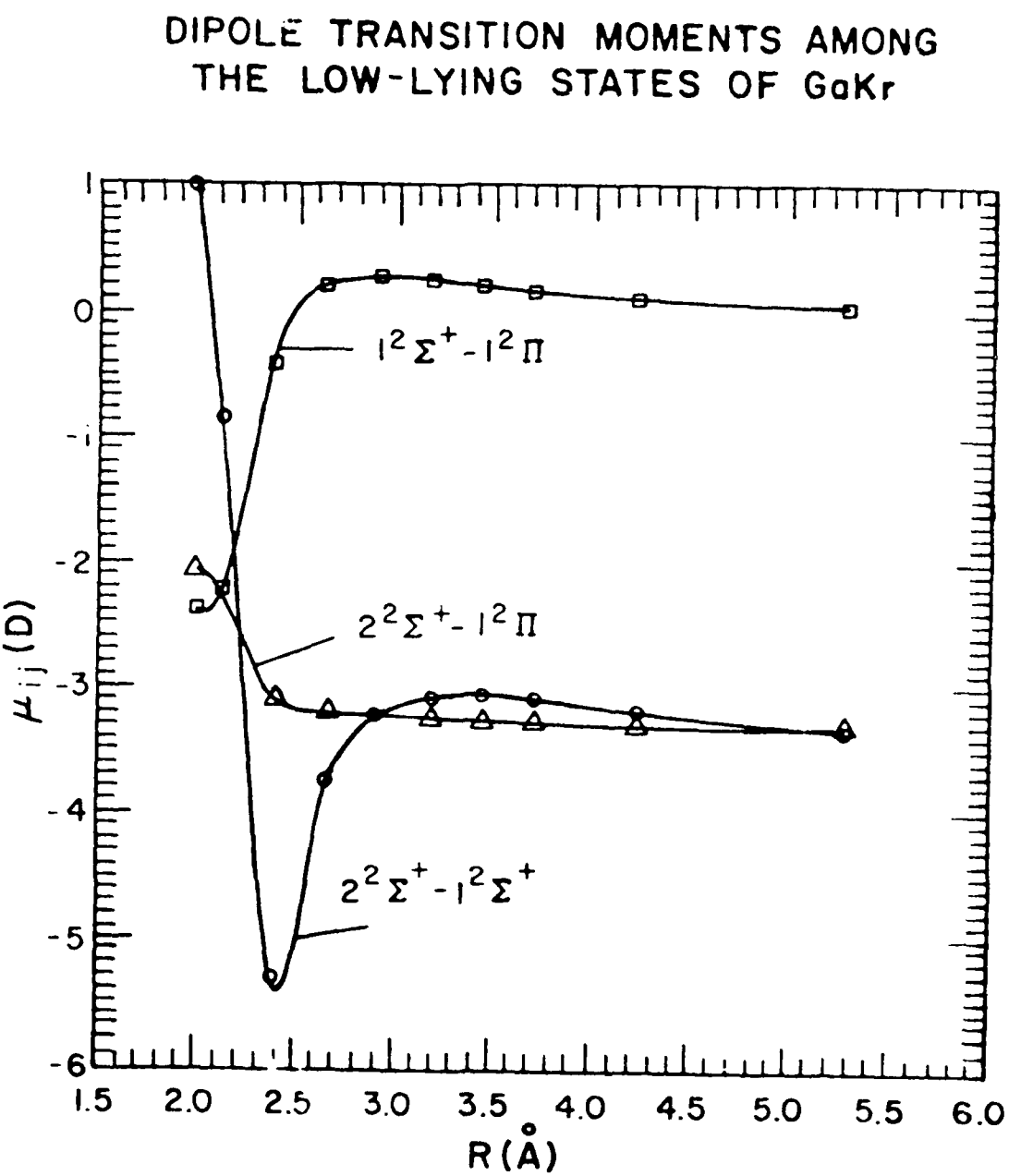

DIPOLE TRANSITION MOMENTS AMONG
THE LOW-LYING STATES OF GaKr
1
0
-1
-2
-3
-4
-5
-6
μ_{ij} (D)
$1^2\Sigma^+ - 1^2\Pi$
$2^2\Sigma^+ - 1^2\Pi$
$2^2\Sigma^+ - 1^2\Sigma^+$
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
R(Å)

Figure 5.

THE LOW-LYING STATES OF GaKr AND $GaKr^+$ WITH SPIN-ORBIT CORRECTIONS

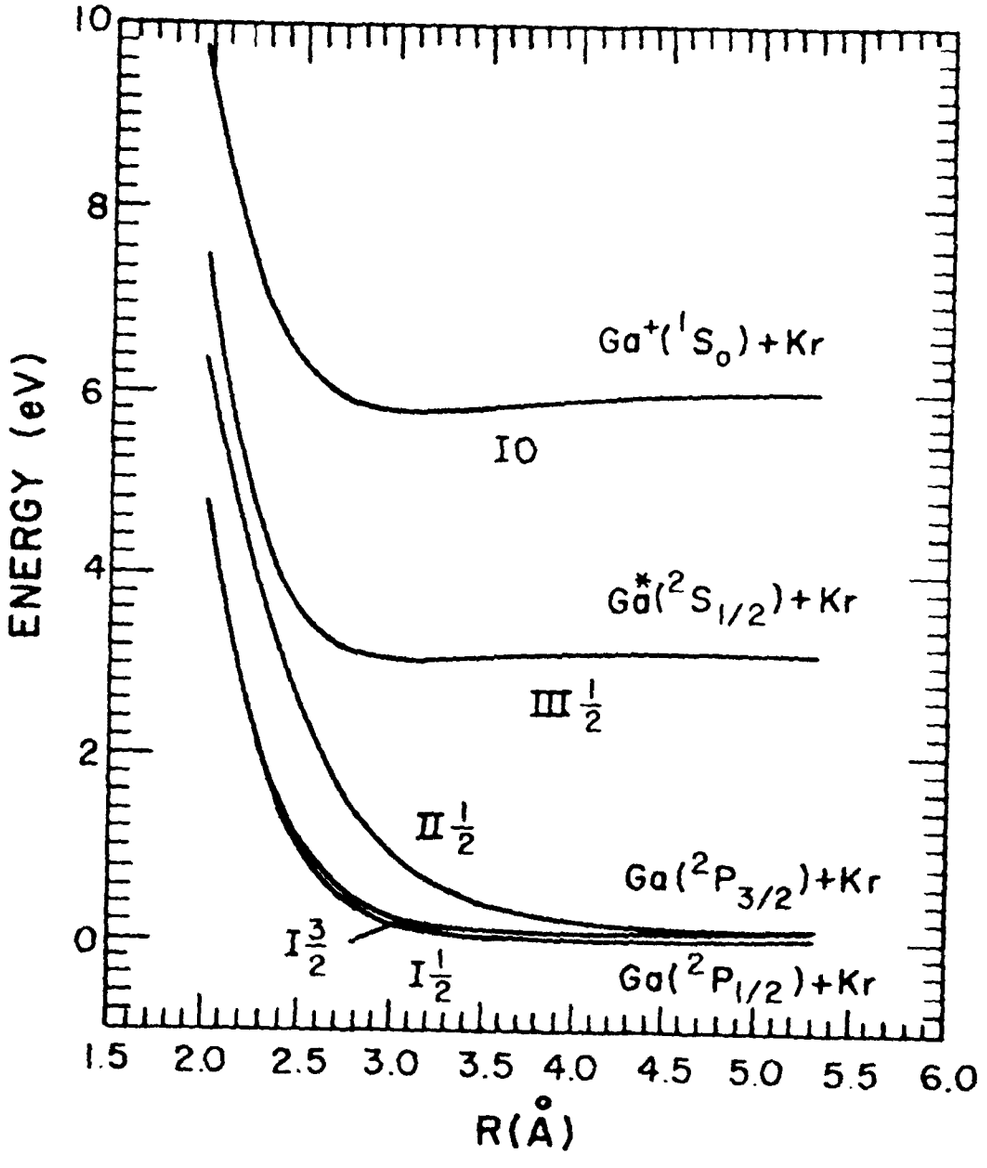


Figure 6.

DIPOLE TRANSITION MOMENTS AMONG THE LOW-LYING STATES OF GaKr WITH SPIN-ORBIT CORRECTIONS

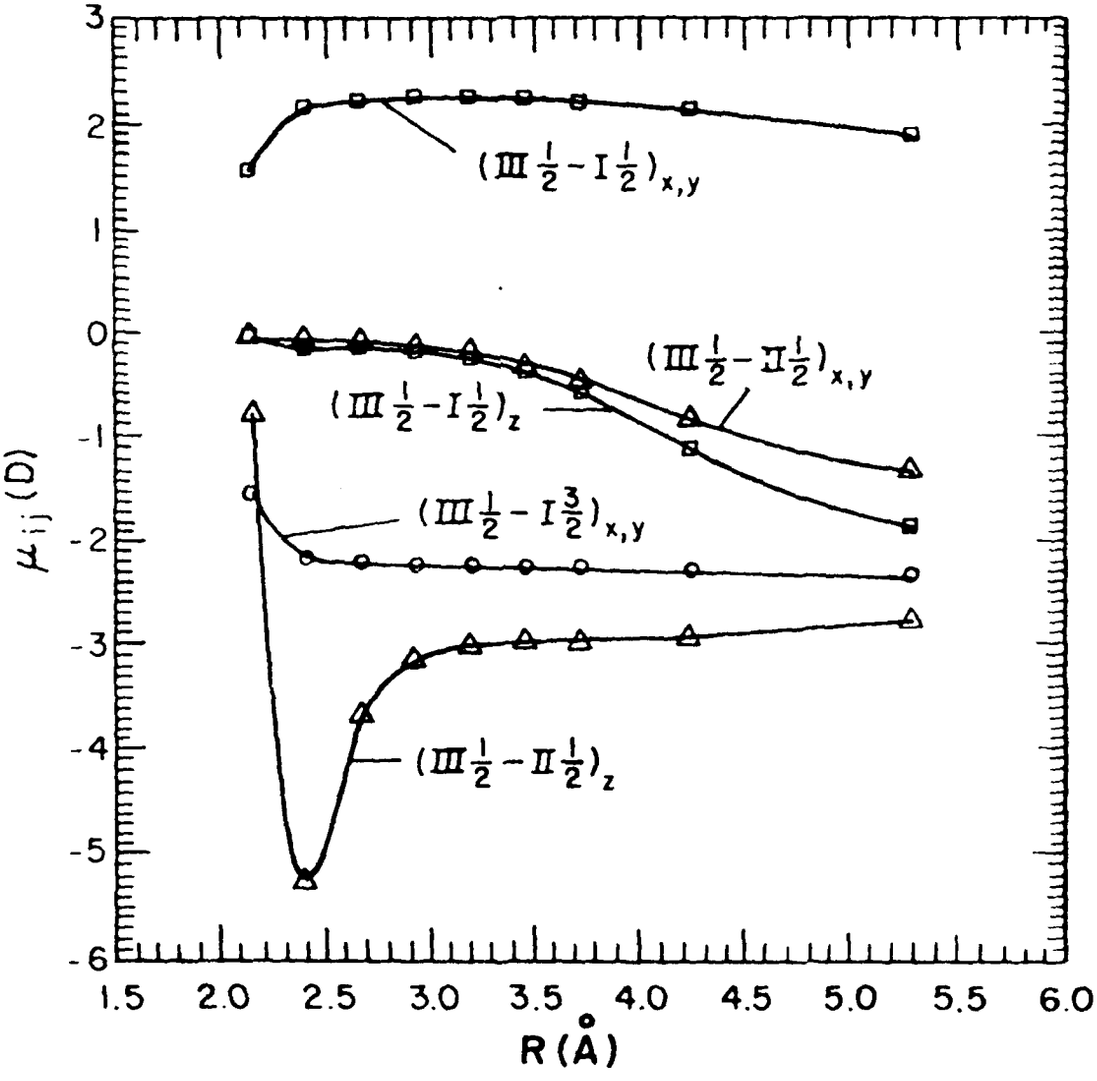


Figure 7.

THE LOW-LYING STATES OF InKr AND $InKr^+$ WITH SPIN-ORBIT CORRECTIONS

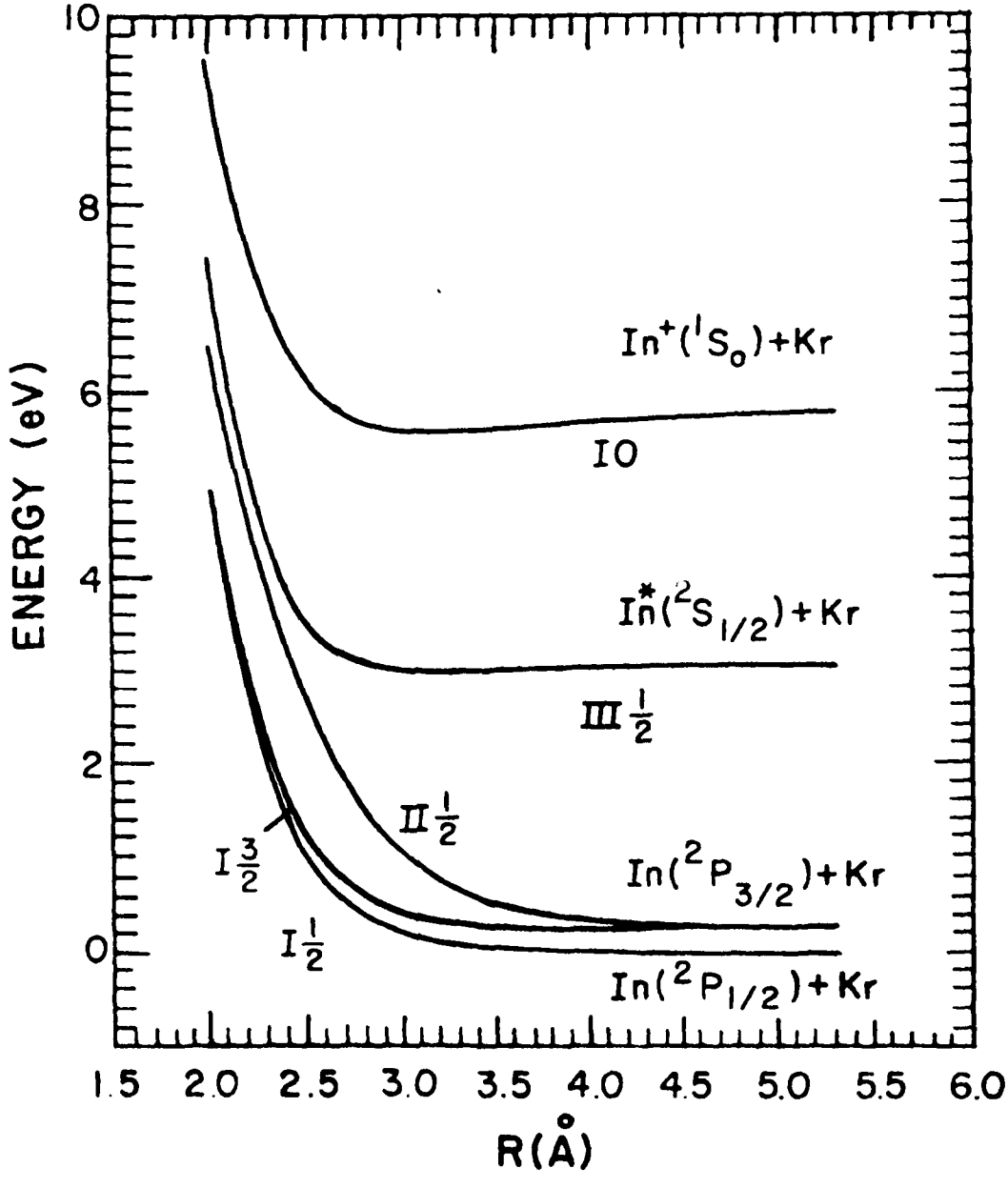


Figure 8.

THE LOW-LYING STATES OF TlKr AND $TlKr^+$ WITH SPIN-ORBIT CORRECTIONS

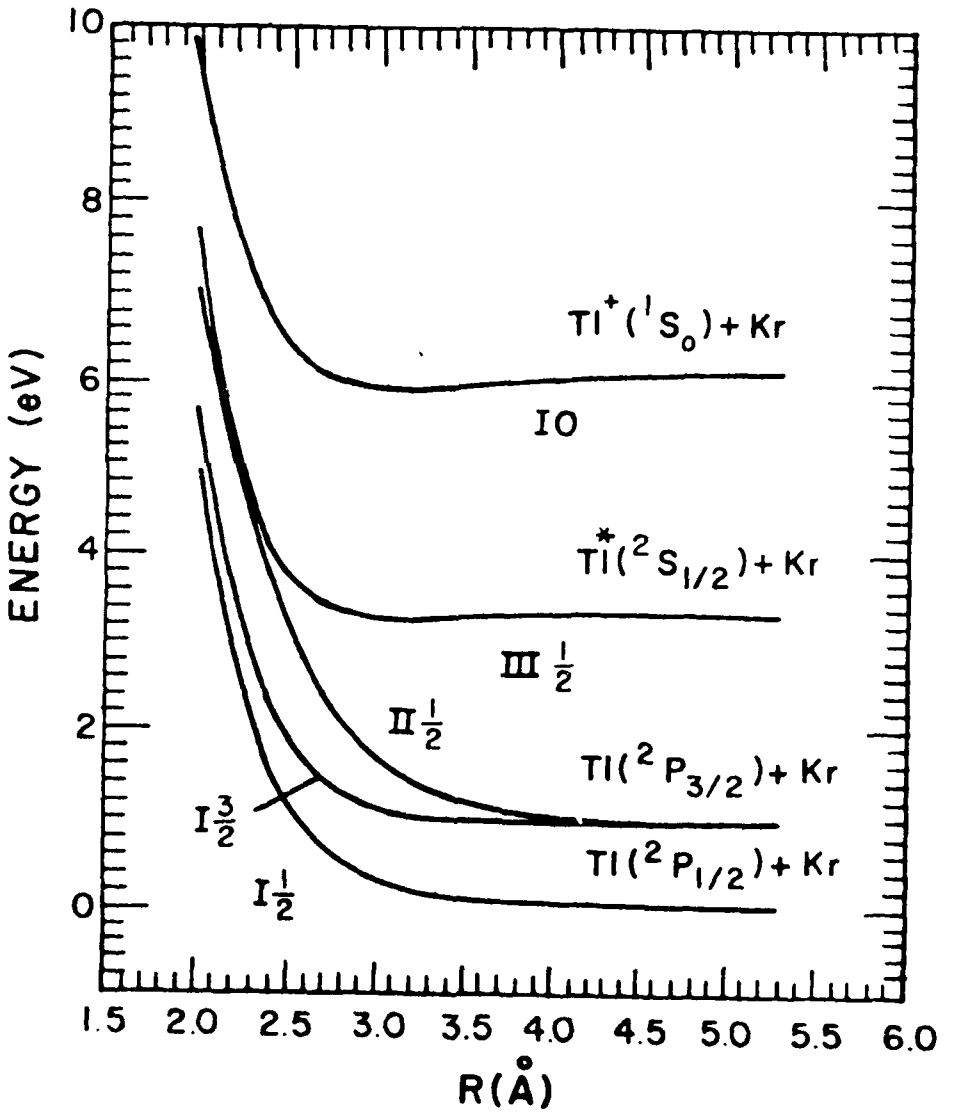


Figure 9.

DIPOLE TRANSITION MOMENTS CONNECTING THE LOW-LYING STATES OF TIKr WITH SPIN-ORBIT CORRECTIONS

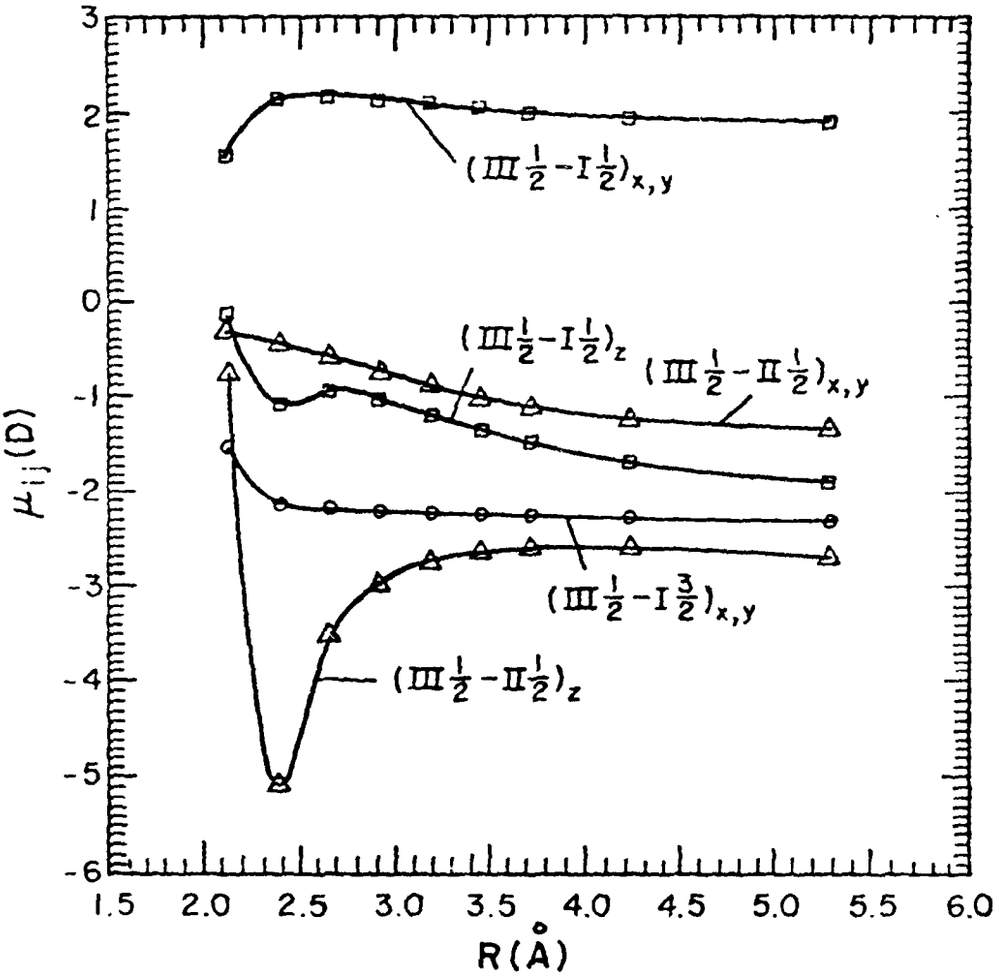


Figure 10.

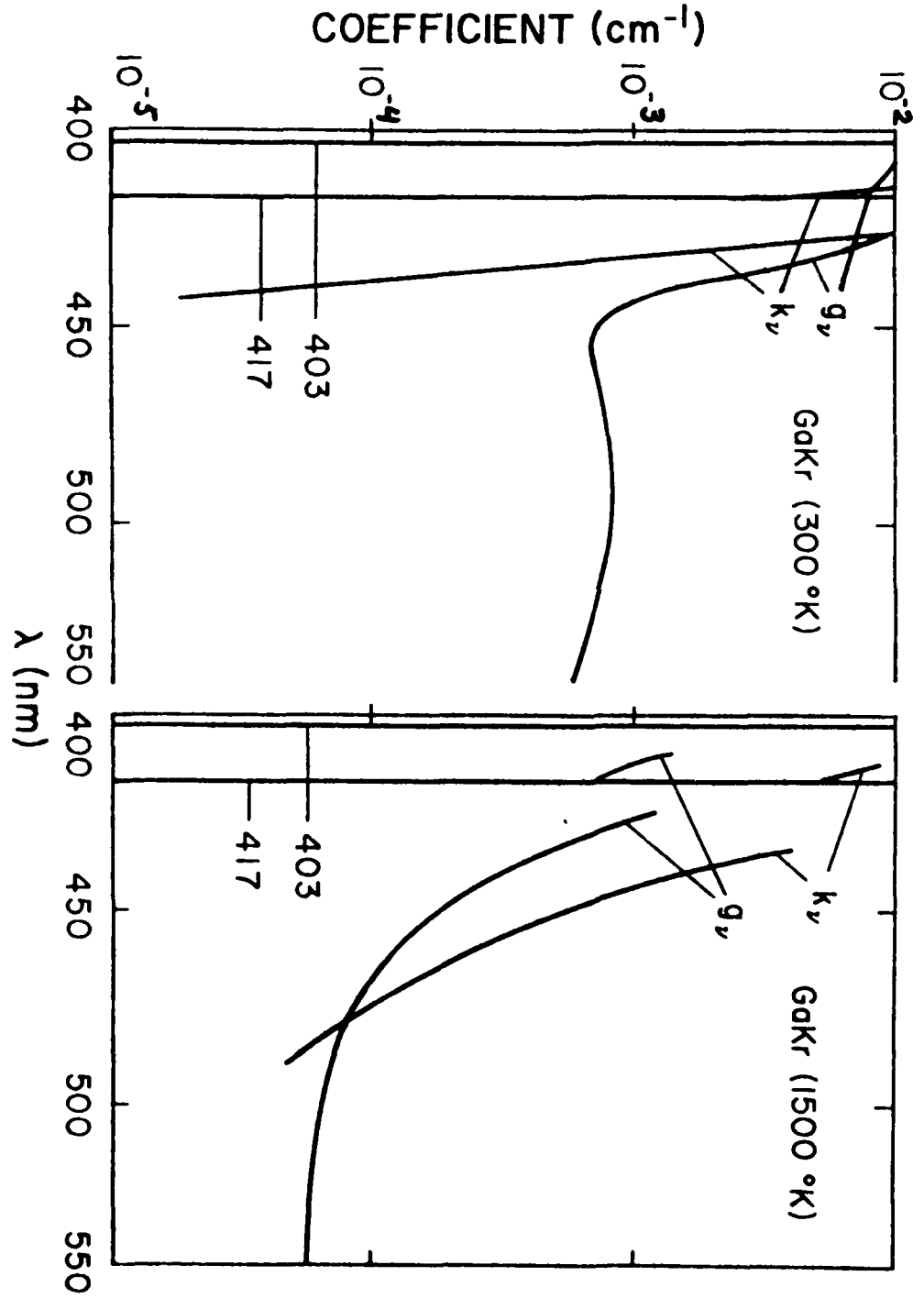

COEFFICIENT (cm⁻¹)
10⁻⁵
10⁻⁴
10⁻³
10⁻²
400
450
500
550
λ (nm)
417
403
k_ν
g_ν
GaKr (300 °K)
417
403
g_ν
k_ν
GaKr (1500 °K)

Figure 11.

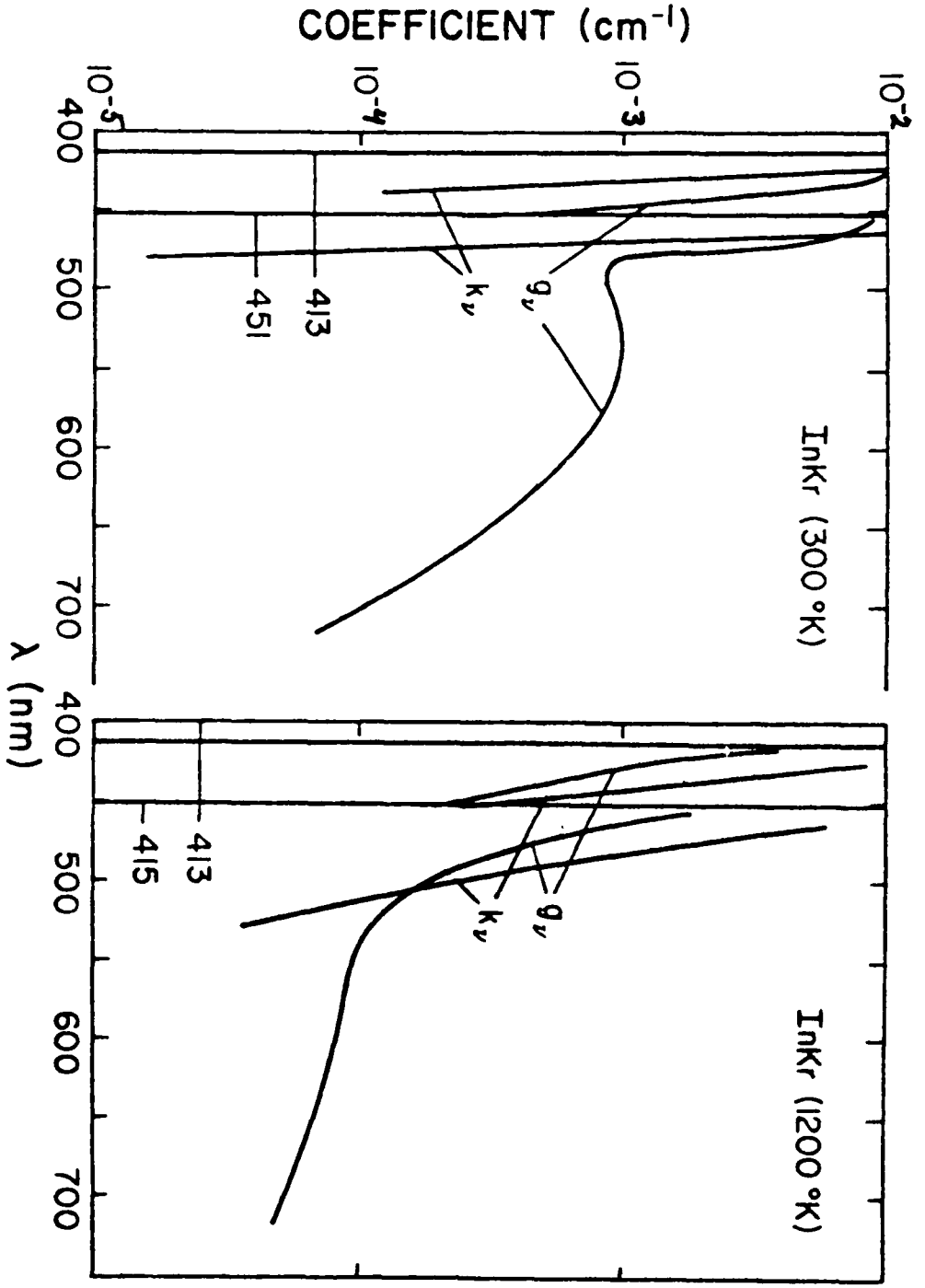
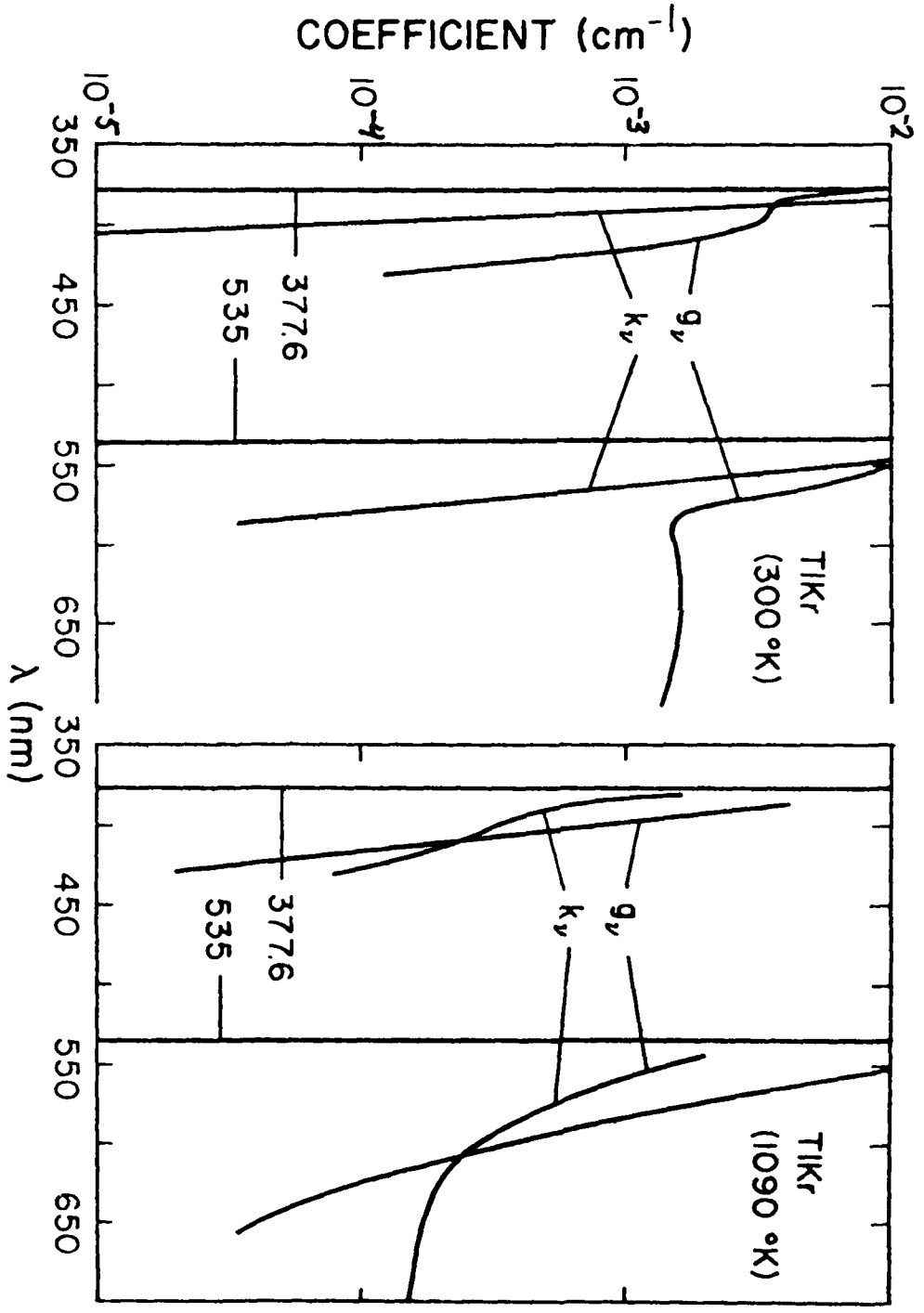

COEFFICIENT (cm⁻¹)
10⁻⁵
10⁻⁴
10⁻³
10⁻²
400
500
600
700
λ (nm)
451
413
k_ν
g_ν
InKr (300 °K)
415
413
k_ν
g_ν
InKr (1200 °K)

Figure 12.


COEFFICIENT (cm⁻¹)
10⁻⁵
10⁻⁴
10⁻³
10⁻²
350
450
550
650
λ (nm)
377.6
535
k_ν
g_ν
TlKr
(300 °K)
TlKr
(1090 °K)

§3

Application of the Gordon-Kim theory to the group IIIB-rare gas systems.

In recent years, the electron gas methods developed by Gaydaenko and Nikulin [1], and Gordon and Kim [2] and modified by Rae [3] and Cohen and Pack [4] have proved successful in calculating the interaction energies of pairs of closed shell atoms or ions [1-5], of atom-molecule systems[6-7], of molecule-molecule systems [8], and, recently, of systems with one open shell atom [9]. These interaction energies are much more accurate than the simplicity of the Gordon-Kim (GK) method would lead one to expect.

The recent success of the GK method in calculating closed shell - open shell interaction energies has prompted us to use this method to calculate the interaction of group IIIB-noble gas pairs. The GK theory has been most successful in cases where the interaction is non-covalent and where the atoms are relatively undistorted. For these reasons, systems such as GaKr would seem to be ideal candidates for a GK calculation.

The electron-gas theory is briefly reviewed in part A, and our preliminary results on GaKr are given in part B.

A. The electron gas method

The method we have used is that of Gordon and Kim [2] as modified by Cohen and Pack [4]. A more detailed description of the theory is available in these two papers. Briefly the GK theory approximates the intermolecular

potential $V(R)$ at the distance R by

$$V(R) \simeq V^{GK}(R) = V^{GK}_{HF}(R) + V^{GK}_{corr}(R)$$

where

$$V^{GK}_{HF} = V_k + V_c + V_e$$

and these three terms represent the kinetic, Coulomb and exchange interaction energies, respectively. To calculate these interaction energies, the electronic charge density ρ is approximated by the sum of the atomic charge densities,

$$\rho \simeq \rho_a + \rho_b \ .$$

With this approximation, the Coulomb interaction can be calculated directly, but the other terms are all estimated by the formulas for the energy density of a uniform electron gas [4]. An additional modification [3,4] is made to the exchange energy to avoid self-exchange contributions.

B. Calculations

We have modified the molecule-molecule GK interaction program of Parker, Snow and Pack [8], which allows for non-spherical potentials, to calculate interaction energies for closed shell - open shell atomic pairs. The density of the open shell atom, in this case Ga , is divided into

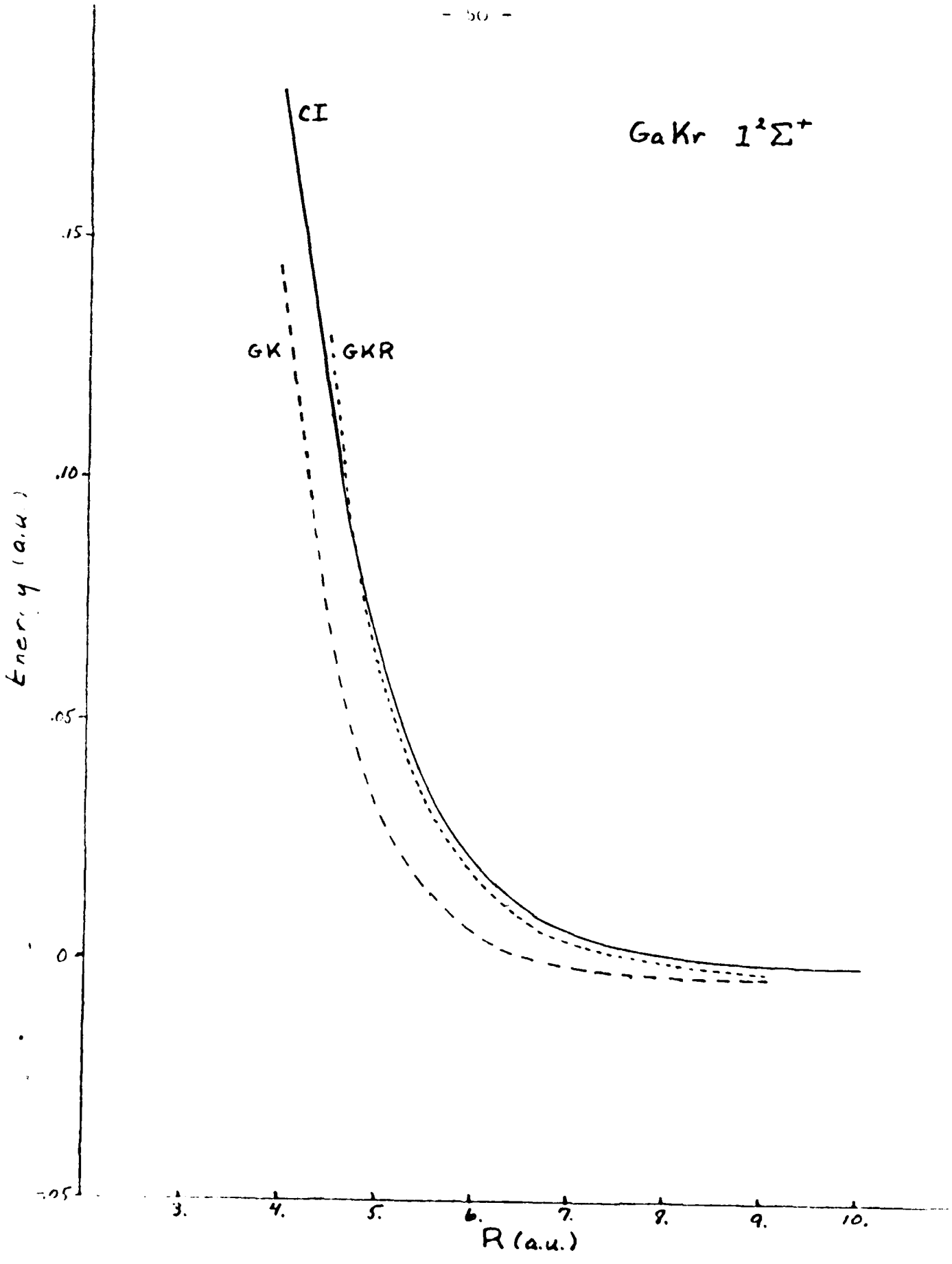
the spherically symmetric core density plus the valence density. For Ga in the ground state ($4s^2 4p^1$), the valence density is constrained to be in a p orbital directed along the internuclear axis, to form a Σ molecular state, or perpendicular to it to form a Π state.

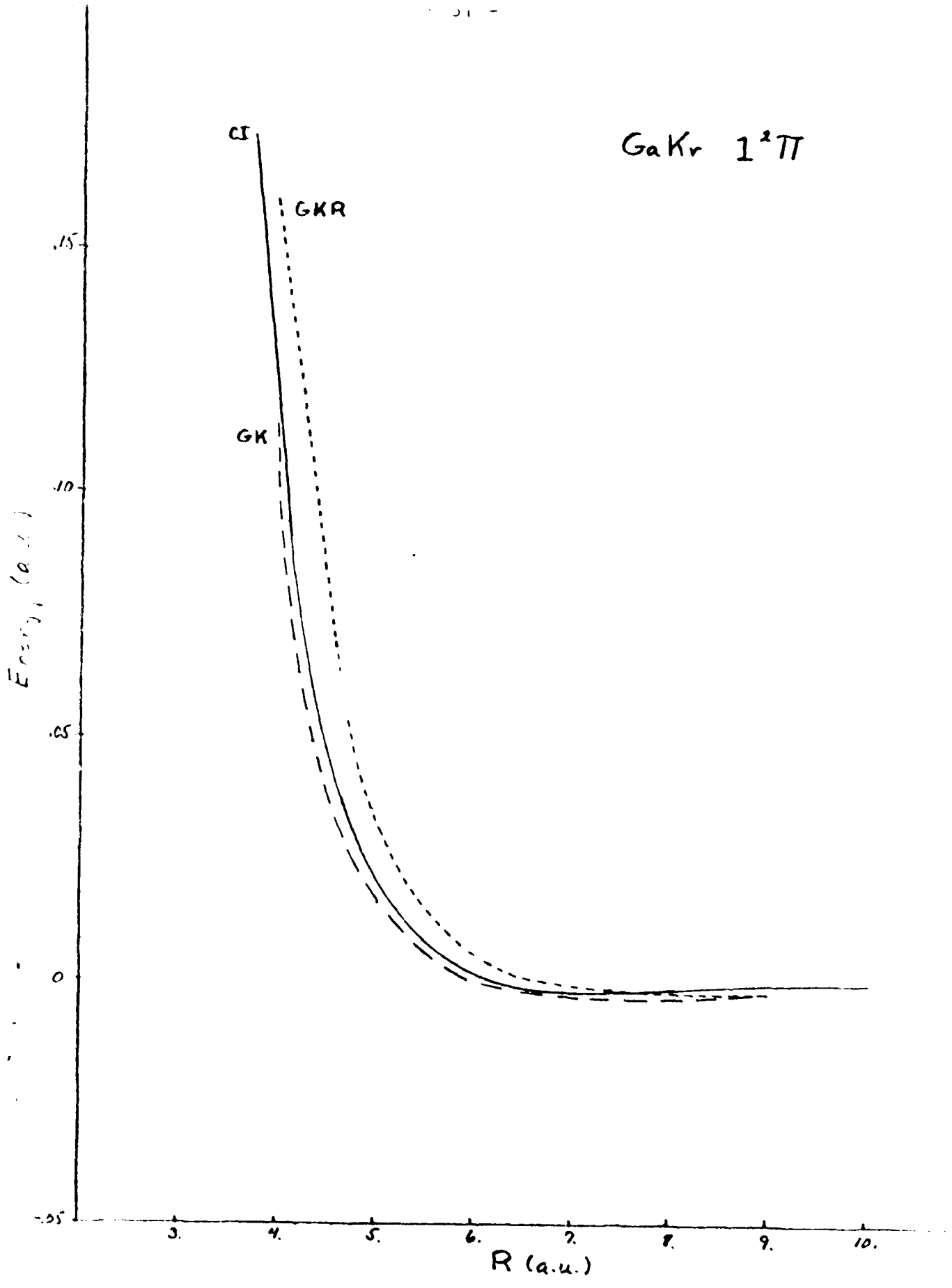
In order to expedite the evaluation of integrals, the atomic density is expanded in a set of Slater basis functions with the coefficients determined by a least-squares fit [4]. The basis set expansions for the density are then read into the GK program which calculates the interaction energy by three-dimensional numerical quadriture.

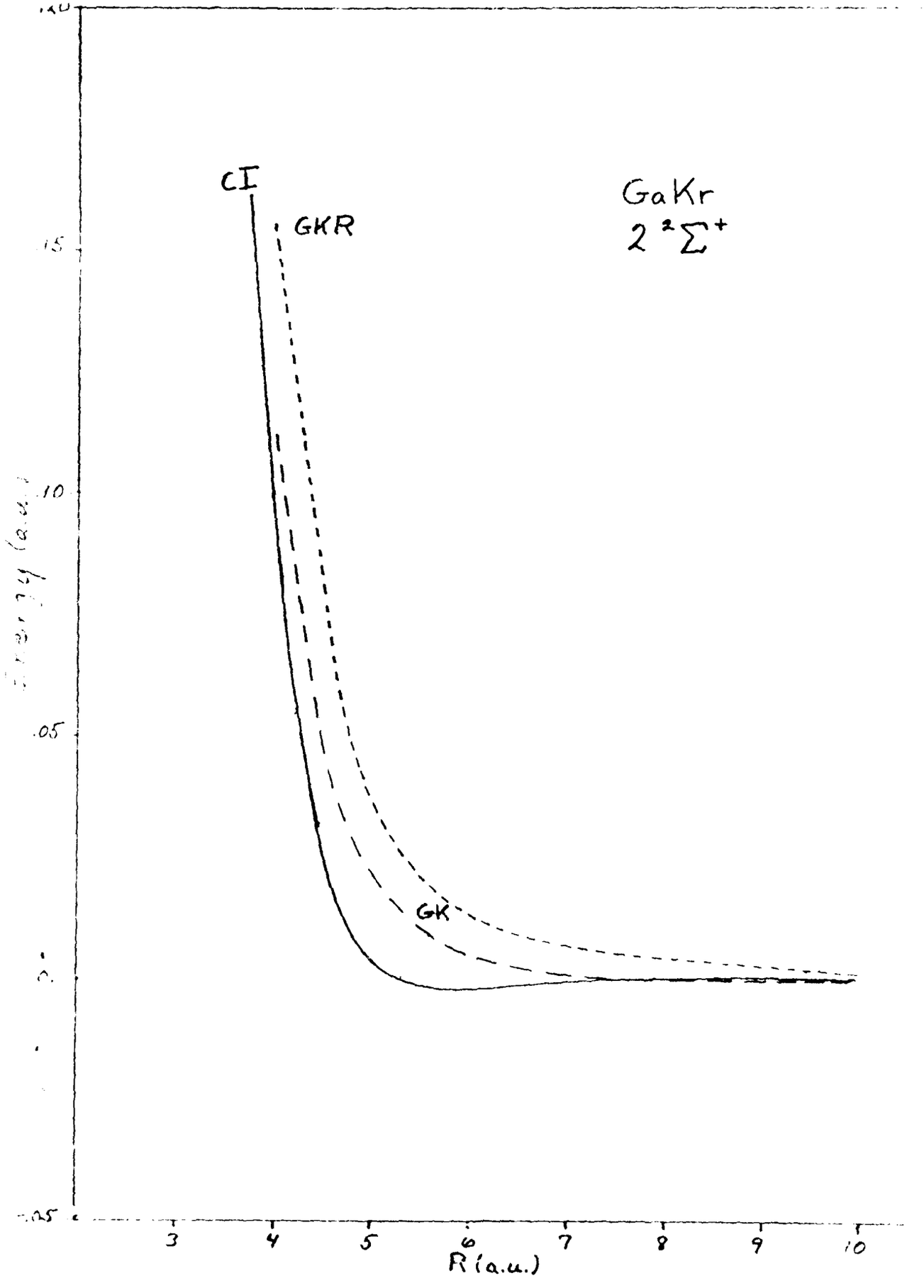
We have obtained numerically tabulated, relativistic Hartree-Fock densities for the group IIIB and rare gas atoms from Joseph Mann [10]. The densities of the $4s^2 4p^1$ and $4s^2 5s^1$ states of Ga and the ground state of Kr were fit with small sets of Slater functions. At this time the fit of the basis set expansion is not very good ($\sim$ 10%).

The basis set expansions obtained from this fitting procedure were then used to calculate the GK interaction energies. Because the fitting procedure does not normalize the density, the inaccuracy of the present expansions results in a spurious Coulomb repulsion between the two atoms. We hope that improving the basis set fit will correct this error.

The GK potential curves are compared with the CI results (see Section II) in Figures 1-3. The energies are plotted with respect to the asymptotic energy of each state. The curve labeled GKR includes Rae's correction while the GK curve is the unmodified GK theory. The GKR curve for the $1^2\Sigma^+$ state agrees remarkably well with the CI result. Unfortunately, the agreement for both the $1^2\Pi$ and $2^2\Sigma^+$ states are not as good. Specifically the depths and positions of the potential wells are not predicted accurately. We are currently working on improving the basis set fit and thereby the interaction energies.


GaKr $1^2\Sigma^+$
CI
GK
GKR
.15
.10
.05
0
-.05
Energy (a.u.)
3.
4.
5.
6.
7.
8.
9.
10.
R (a.u.)


GaKr $1^2\Pi$
CI
GKR
GK
.15
.10
.05
0
-.05
Energy (a.u.)
3.
4.
5.
6.
7.
8.
9.
10.
R (a.u.)


CI
GKR
GK
GaKr
$2\,^2\Sigma^+$
.15
.10
.05
-.05
3
4
5
6
7
8
9
10
R (a.u.)

1. V. I. Gaydaenko and V. K. Nikulin, Chem. Phys. Letters 7, 360 (1970); V. K. Nikulin, Zh. Tekh, Fiz. 41, 41 (1970) [English transl. Soviet Phys. Tech. Phys. 16, 28 (1971)].

2. R. G. Gordon and Y. S. Kim, J. Chem. Phys. 56, 3122 (1972); Y. S. Kim and R. G. Gordon, J. Chem. Phys. 60 1842, 4323, 4332, (1974); 61 1, (1974).

3. A.I.M. Rae, Chem. Phys. Letters 18, 574 (1973).

4. [illegible] Cohen and R. T. Pack, J. Chem. Phys. 61, 2372 (1974).

5. B. Schneider, J. Chem. Phys. 58 4447 (1973).

6. Y. S. Kim, Thesis, Harvard University (1973).

7. [illegible] Green and P. Thaddeus, Astrophys. J. 191, 653 (1974).

8. G. A. Parker, R. L. Snow and R. T. Pack, Chem. Phys. Lett. 33 399 (1975).

9. M. J. Clugston and R. G. Gordon, J. Chem. Phys. 66, 239 (1977).

10. Joseph Mann, private communication.

§4

4.I Theoretical basis

A. Introduction

This section describes the theoretical basis of our calculations using the effective potential method. For an excimer system AB, where A is a closed-shell system, most of the states of interest correspond to the asymptotic situation where B is excited but A is in its ground state. Fundamentally, what the electron scattering theory says is that any charged particle in B sees a potential (Σ_A), due to the closed-shell system A, which is the same as if the charged particle were scattered off of A. This scattering potential is corrected for the fact that A is "depolarised" relative to the scattering problem, due to the presence of the nucleus and other electrons of B. The unique aspect of this work, which distinguishes it from many other high-quality attempts to produce such curves, is that it is based on the model interaction potentials and response functions that arise out of many body theory (using Schwinger Functional Derivatives). The use of this type of many body theory has characterized the work of this group over the last few years.[16]

The effective potential method will clearly give incorrect results when any of three situations are met: First, if charge transfer states mix with the excited states of interest; second, when states go asymptotically to two excited atoms; third, when covalent bonding is important. Only for charge transfer do we have a possible Class I solution. For the moment it is hoped that all three problem situations will be avoided for the lowest excimer excited states in the systems of interest to this proposal. If Zn_2 and Mg_2 are any guide the first two problems occur only for the higher excited states.[17]

In part B the basic equations are presented and possible methods for calculating the interaction energy are discussed in part C. The use of semiempirical forms for the effective potentials are discussed in part D.

B. The Basic Equations

Using many-body field theoretic methods it has been shown that the change in energy, ε , resulting from the addition of an electron to a closed-shell reference system (referred to here as A) is given by the one-particle Dyson equation

$$T(r)\phi(r) + \int dr' \, \Sigma^A(r;r':\varepsilon)\phi(r) = \varepsilon\phi(r) \qquad (1)$$

where T is the kinetic energy operator and ϕ is the Dyson amplitude with r and r' being space-spin coordinates. Thus, the problem reduces to an effective one-particle problem in which this particle experiences an effective potential, Σ^A , which represents all the other particles collectively, taking into account all effects such as polarization, correlation and exchange, etc. As might be expected, the cost of this simple formulation is that Σ^A is an extremely complicated entity which is both nonlocal and energy dependent and which cannot rigorously be brought into closed form. However, it has been possible to develop

excellent closed form approximations to this potential which are based on well founded physical concepts. Most notable among these is the Random Phase Approximation (RPA) potential, Σ_{RPA} , which has been very successfully used in calculating the ionization potentials, excitation energies, oscillator strengths, and elastic -scattering phase shifts for He [18]. Σ_{RPA} has also been used to accurately calculate the ionization and excitation energies of Li [19] Moreover, it has been shown that this ab initio potential encompasses other phenomenologically derived semiempirical potentials which have been used by other workers with great success [20].

It has also been recently demonstrated that by applying the same many-body techniques to the problem of two electrons added to a closed-shell reference system (A) one obtains an effective two-particle equation for the resulting change in energy [21].

$$[\sum_{i=1}^{2}\left(T(r_i) + \int dr' \Sigma^A(r_i;r_i':\varepsilon)\right) + \frac{1}{|r_1-r_2|} + \int dr_1' dr_2' W^A(r_1,r_2;r_1'r_2')]\Phi(r_1,r_2) =$$

$$\varepsilon\Phi(r_1,r_2) , \qquad (2)$$

where in addition to the individual one-particle potentials there now appears a two-particle effective potential, W^A , which represents how the presence of one particle affects the potential seen by the other particle and which reflects the fact that system A can act as a dielectric medium to shield the Coulombic interaction between two charged particles. As with Σ^A , this potential cannot be rigorously brought into closed form.

However, an excellent ab initio closed form approximation to this potential can be obtained with the Random Phase Approximation, W_{RPA} , which is completely compatible with the similarly obtained one-particle potential, Σ_{RPA} .

If we now proceed to the case of adding m electrons to our closed-shell reference system it follows by induction and can be proven formally that the change in energy is given by

$$\left[\sum_{i=1}^{m}\left(T(r_i) + \int dr_i'\, \Sigma^A(r_i;r_i':\varepsilon)\right) + \sum_{i>j}^{m}\left(\frac{1}{|r_i-r_j|} + \int dr_i'\, dr_j'\, W^A(r_i r_j;r_i',r_j':\varepsilon)\right)\right.$$

$$\left. \sum_{j>k}^{m} \int dr_i'\, dr_j'\, dr_k'\, U^A(r_i,r_j,r_k;r_i',r_j',r_k':\varepsilon) + \ldots + \int dr_1' \ldots dr_m' V^A(r_1 \ldots r_m;r_1' \ldots r_m':\varepsilon)\right] \Psi(r_1 \ldots r_m$$

$$= \varepsilon\Psi(r_1 \ldots r_m)\,, \qquad (3)$$

where our notation for three-particle and higher potentials is obvious.

In the above we only considered the addition of electrons to our closed-shell reference system A. However, we can also add nuclei as well; the only difference being that when acting on a nuclear coordinate all potential terms involving electron exchange must of course vanish. Since this can be trivially accomplished by choosing an appropriate (artificial) nuclear spin coordinate, we can immediately generalize (3) to include both nuclei and electrons by allowing the particles to carry different charges, z , (where z = -1 for an electron) to give

$$[\sum_{i=1}^{m}(T(r_i) - \int dr_i' \Sigma^A(r_i;r_i':\epsilon,z_i)) + \sum_{i>j}^{m} (\frac{z_i z_j}{|r_i - r_j|} + \int dr_i' dr_j' W^A(r_i,r_j;r_i',r_j':\epsilon,z_i,z_j))$$

$$+ \ldots] \Psi(r_1 \ldots r_m) = \epsilon \Psi(r_1 \ldots r_m) . \qquad (4)$$

We of course cannot solve (4) since the potentials involved cannot be written in closed form. However, we can replace these potentials by their RPA approximates which are in closed form. Furthermore, realizing that we are deriving a theory for intermolecular forces which is essentially perturbative in nature, we will now assume that all three-particle and higher potentials can be neglected so that our equation becomes simply

$$[\sum_{i=1}^{m}(T(r_i) - \int dr_i' \Sigma^A(r_i;r_i':\epsilon,z_i)) + \sum_{i>j}^{m} (\frac{z_i z_j}{|r_i - r_j|} + \int dr_i' dr_j' W^A(r_i,r_j;r_i',r_j':\epsilon,z_i,z_j))] \Psi(r_1 \ldots r_m)$$

$$= \epsilon \Psi(r_1 \ldots r_m) . \qquad (5)$$

Since our interest lies in the calculation of potential energy surfaces, what we really want is an equation for the change in energy when m_e electrons and m_n nuclei are added to A with these nuclei held fixed at specific points in space (which we will refer to collectively as R with the actual spatial coordinate of nuclei i being R_i). That is, we want to be able to separate electronic motion from nuclear motion so that this quantity, $\mathcal{E}(R)$, will be the total potential experienced by these nuclei and will satisfy the equation

$$[\sum_{i=1}^{m_n} T(R_i) + \mathcal{E}(R)] X(R) = \epsilon X(R) \qquad (6)$$

where $X(R)$ is a function of the nuclear coordinates only. As it stands, (5) precludes such a separation because of the energy dependence of the potentials involved. To overcome this problem we will now assume that for those solutions to (5) we seek this energy dependence is not strong. Furthermore, we will also assume that the response of A to the added particles is instantaneous. In this way we can replace the non-adiabatic energy dependent potentials in (5) with their hermitian energy independent adiabatic approximates such as those given in ref.21. With these substitutions (5) does become separable, and by taking

$$\psi(r_1 \ldots r_m) = \sum_j X_j(R)\psi_j(r_1 \ldots r_{m_e}) \; ; \; \langle \psi_i | \psi_j \rangle = \delta_{ij} \qquad (7)$$

where $\psi(r_1 \ldots r_{m_e})$ is a function of only the electronic coordinates we find that $\mathcal{E}(R)$ is given by

$$\mathcal{E}(R) = E^A(R) - \sum_{i=1}^{m_n} \Sigma^A(R_i;z_i) + \sum_{i>j}^{m_n} \left(\frac{z_i z_j}{|R_i - R_j|} + W^A(R_i,R_j;z_i,z_j)\right), \qquad (8)$$

where E^A is given by the equation

$$\left[H(r_1 \ldots r_{m_e};R) + \sum_{i=1}^{m_e}\left(\int dr_i' \Sigma^A(r;r') - \sum_{j=1}^{m_n} \int dr_i' W^A(r_i,R_j;r_i';zj)\right)\right.$$

$$\left. + \sum_{i>j}^{m_e} \int dr_i' dr_j' W^A(r_i,r_j;r_i'r_j')\right] \psi(r_1 \ldots r_{m_e}) = E^A(R)\, \psi(r_1 \ldots r_{m_e}) \qquad (9a)$$

where H is the usual hamiltonian for m_e electrons in the field of m_n fixed nuclei

$$H(r_1 \ldots r_{m_e};R) = \sum_{i=1}^{m_e}\left(T(r_i) - \sum_{j=1}^{m_n}\frac{z_j}{|r_i-R_j|}\right) + \sum_{i>j}^{m_e}\frac{1}{|r_i-r_j|}$$

$$= \sum_{i=1}^{m_e} h(r_i) + \sum_{i>j}^{m_e}\frac{1}{r_{ij}} \qquad (9b)$$

and where all one- and two-particle effective potentials are taken to be hermitian adiabatic approximates to the true potentials. Note also that all potential terms involving the m_n nuclei are now written as local quantities thereby taking into account the previously mentioned fact that there are no exchange (i.e. nonlocal) terms in the potentials when nuclei are being considered.

If we now collectively refer to these m_e electrons and m_n nuclei which have a fixed internuclear geometry as being system B, then $\mathcal{E}(R)$ is the change in energy resulting from the creation of system B in the vicinity (as measured by R) of A. If $R \to \infty$ then $\mathcal{E} \to \mathcal{E}_B^o$ which is just the energy B itself. Therefore, the intermolecular potential of the system A-B as a function of the separation between A and B is

$$V_B^A(R) = E_B^A(R) - E_B^o + V_{B,nuc}^A(R) \qquad (10a)$$

where

$$V_{B,nuc}^A(R) = -\sum_{i=1}^{m_n} \Sigma^A(R_i;z_i^B) + \sum_{i>j}^{m_n} W^A(R_i,R_j;z_i^B,z_j^B) \qquad (10b)$$

and where

$$E_B^o = \mathcal{E}_B^o + \sum_{i>j}^{m_n} z_i^B z_j^B \frac{1}{|R_i - R_j|} \tag{10c}$$

is just the electronic energy of isolated system B and where $E_B^A(R)$ is given by

$$[H(r_1 \ldots r_{m_e};R) + \sum_{i=1}^{m_e} \int dr_i' (\Sigma^A(r_i;r_i') - \sum_{j=1}^{m_n} W^A(r_i,R_j;r_i';Z_j^B))$$

$$+ \sum_{i>j}^{m_e} \int dr_i' dr_j' W^A(r_i,r_j;r_i'r_j')] \Psi_B^A(r_1 \ldots r_{m_e}) = E_B^A(R) \Psi_B^A(r_1 \ldots r_{m_e}) \tag{10d}$$

Therefore, given that we know E_B^o and that we have available good closed form adiabatic approximates for Σ^A and W^A our problem reduces to finding the solutions (or rather a particular solution) to (10d). In the following section we will look at some ways of doing this.

C. Determination of E_B^A

Our equation for E_B^A is of the form

$$(H + U^A)\,\psi_B^A = E_B^A\,\psi_B^A \tag{11a}$$

where

$$U^A(r_1 \ldots r_{m_e};R) = \sum_{i=1}^{m_e} \int dr_i' [\Sigma^A(r_i;r_i') - \sum_{j=1}^{m_n} W^A(r_i,R_j;r_i';Z_j^B)]$$

$$+ \sum_{i>j}^{m_e} \int dr_i' dr_j' W^A(r_i,r_j;r_i',r_j')$$

$$= \sum_{i=1}^{m_e} p^A(r) + \sum_{i>j}^{m_e} q_{12}^A(r_i,r_j) \tag{11b}$$

where we note that ψ_B^A is an eigenfucntion of the hermitian 'hamiltonian' $(H + U^A)$ and that H is the electronic hamiltonian for isolated system B. If we now explicitly assume that U^A is small compared to H (this of course was implicit in our derivation of U^A in the first place) then U^A can be regarded as being a small perturbation on H. As a consequence, the solution, ψ_B^A, to (11) which we seek should resemble the electronic wavefunction for isolated system B, ψ_B^o, and this wavefunction should therefore provide us with a proper starting point determining ψ_B^A and E_B^A.

1. First-Order Perturbation Treatment

Projecting (11a) against ψ_B^A and normalizing ψ_B^A to unity gives

$$E_B^A = \langle \Psi_B^A | H + U^A | \Psi_B^A \rangle \; ; \; \langle \Psi_B^A | \Psi_B^A \rangle = 1 \, , \tag{12a}$$

whereas if Ψ_B^0 is a self-consistent solution for isolated system B its energy is

$$E_B^0 = \langle \Psi_B^0 | H | \Psi_B^0 \rangle \; ; \; \langle \Psi_B^0 | \Psi_B^0 \rangle = 1 \, . \tag{12b}$$

If we assume that U^A is a quite small perturbation to H then we can solve for E_B^A using standard first-order perturbation theory to obtain

$$E_B^A = E_B^0 + \langle \Psi_B^0 | U^A | \Psi_B^0 \rangle \tag{13}$$

so that our first-order perturbative expression for V_B^A is simply

$$V_B^A = V_{B,nuc}^A + \langle \Psi_B^0 | U^A | \Psi_B^0 \rangle \, . \tag{14}$$

This of course is equivalent to assuming that the wavefunction for system B remains virtually unchanged when in the presence of A .

In terms of the one- and two-particle density matrices, ρ_1^0 and ρ_2^0 , associated with Ψ_B^0 , (14) becomes

$$V_B^A = V_{B,nuc}^A + \int_{r'=r} dr \; p^A(r') \rho_1^0(r;r') + \int_{r'=r} dr_1 dr_2 \; q_{12}^A(r_1', r_2') \rho_2^0(r_1, r_2; r_1' r_2') . \tag{15}$$

If Ψ_B^0 is a single Slater determinant so that

$$\rho_2^0(r_1,r_2;r_1',r_2') = 1/2[\rho_1^0(r_1;r_1')\rho_1^0(r_2;r_2') - \rho_1^0(r_1;r_2')\rho_1^0(r_2;r_1')] \quad (16a)$$

with

$$\rho_1^0(r;r') = \sum_{i=1}^{n} \phi_i^*(r)\,\phi_i(r') \quad (16b)$$

where $\{\phi_i\}$ are the n one-electron spin-orbitals comprising Ψ_B^0, then V_B^A takes on the form

$$V_B^A = V_{B,nuc}^A + \sum_{i=1}^{n} \langle\phi_i|p^A|\phi_i\rangle + 1/2 \sum_{i\neq j}^{n} [\langle\phi_i\phi_j|q_{12}^A|\phi_i\phi_j\rangle - \langle\phi_i\phi_j|q_{12}^A|\phi_j\phi_i\rangle] \quad (17)$$

where the two-electron integrals over q_{12}^A are written in the standard $\langle 12||12\rangle$ notation.

If Ψ_B^0 cannot be written as a single Slater determinant but can be written as a linear combination of determinants involving n one-electron spatial orbitals $\{\phi_i^0\}$ then (12b) can always be written as

$$E_B^0 = \sum_{i,j}^{n} D_i^j \langle\phi_i^0|h|\phi_j^0\rangle + \sum_{i,j,k,l}^{n} D_{ij}^{kl} \langle\phi_i^0\phi_j^0|\frac{1}{r_{12}}|\phi_k^0\phi_l^0\rangle \quad (18)$$

where $\{D_i^j, D_{ij}^{kl}\}$ are fixed coefficients which depend on the precise form of Ψ_B^0 and on orbital overlaps. Since U^A is of the same form as H and, like H, is spin independent (recall that isolated system A must be closed-shell) we can immediately write our first-order approximation for V_B^A as

$$V_B^A = V_{B,nuc}^A + \sum_{i,j}^{n} D_i^j \langle \phi_i^o | p^A | \phi_j^o \rangle + \sum_{i,j,k,l}^{n} D_{ij}^{kl} \langle \phi_i^o \phi_j^o | q_{12}^A | \phi_k^o \phi_l^o \rangle \qquad (19)$$

where all operators and integrals are now explicitly takent to involve only spatial coordinates. Therefore, we can determine $\langle \Psi_B^o | U^A | \Psi_B^o \rangle$ by using the general energy expression (18) but employing one- and two-electron integrals over the operators p^A and q_{12}^A instead of the usual integrals over h and $\frac{1}{r_{12}}$.

We could continue this perturbation treatment by going on to determine second-order and higher corrections. However we will stop here and next consider a self-consistent approach instead.

2. Self-Consistent Treatment

Let us now assume that while U^A is a small perturbation to H it is not small enough to justify a simple first-order perturbation treatment. That is, we will now assume that Ψ_B^A can still be taken to be functionally the same as Ψ_B^o but because of the presence of A the spatial orbitals themselves distort away from $\{\phi_i^o\}$ to a meaningful extent. The problem is then to determine these new orbitals $\{\phi_i\}$. Since Ψ_B^A is an eigenfunction of electronic motion satisfying (11) this can be done variationally. That is, we can determine these new distorted orbitals by requiring that E_B^A be stationary with respect to changes in these orbitals. If for simplicity (but not necessity) we assume that Ψ_B^o can be written as a Hartree-Fock type wavefunction involving orthonormal

spatial orbitals (or orthonormal spin-orbitals for the case of an Unrestricted Hartree-Fock wavefunction) then (18) takes on the simple Roothaan form

$$E_B^o = 2 \sum_{i=1}^{n} f_i \langle \phi_i^o | h | \phi_i^o \rangle + \sum_{i,j}^{n} [a_{ij} J_{\phi_i^o \phi_j^o} + b_{ij} K_{\phi_i^o \phi_j^o}] \tag{20a}$$

where

$$J_{\phi_i^o \phi_j^o} = \langle \phi_i^o | J_{\phi_j^o} | \phi_i^o \rangle = \langle \phi_j^o | J_{\phi_i^o} | \phi_j^o \rangle = \langle \phi_i^o \phi_j^o | \frac{1}{r_{12}} | \phi_i^o \phi_j^o \rangle \tag{20b}$$

$$K_{\phi_i^o \phi_j^o} = \langle \phi_i^o | K_{\phi_j} | \phi_i \rangle = \langle \phi_j | K_{\phi_i} | \phi_j^o \rangle = \langle \phi_i^o \phi_j^o | \frac{1}{r_{12}} | \phi_j^o \phi_i^o \rangle \tag{20c}$$

and where $\{f_i; a_{ij}; b_{ij}\}$ are fixed coefficients (for the case of a multi-configurational wavefunction these coefficients are simply related to the variationally determined configurational coefficients) which depend on the precise form of ψ_B^o. Since ψ_B^A is being taken to have the same form as ψ_B^o and because of the similarities between H and U^A mentioned above, we can immediately write our equation for E_B^A as

$$E_B^A = 2 \sum_{i=1}^{n} f_i \langle \phi_i | h^A | \phi_i \rangle + \sum_{i,j} [a_{ij} \langle \phi_i \phi_j | g_{12}^A | \phi_i \phi_j \rangle + b_{ij} \langle \phi_i \phi_j | g_{12}^A | \phi_j \phi_i \rangle] \tag{21a}$$

where the operator h^A and g_{12}^A are given by

$$h^A = h + p^A \; ; \; g_{12}^A = \frac{1}{r_{12}} + q_{12}^A \tag{21b}$$

Therefore, $\{\Phi_i\}$ and therefore E_B^A can be determined using standard basis set expansion (LCAO) SCF techniques. The only difference is that instead of using the usual one- and two-electron integrals we must use integrals over the operators h^A and g_{12}^A instead. However, since these integrals serve only as imput this difference is transparent to whatever available SCF procedure we employ.

3. Configuration Interaction Treatment

For the sake of completeness it should be mentioned that since Ψ_B^A can be variationally determined there is no need to stop at the SCF level and we could solve for this function as a Configuration Interaction (CI) problem. The only difference form a standard CI calculation is that instead of using the usual one- and two-electron integrals we must use integrals over the h^A and g_{12}^A operators defined in (21b). It should be pointed out however that any solution for Ψ_B^A which differs significantly from Ψ_B^o implies that for that solution U^A can no longer be regarded as a small perturbation to H and in such a case the validity of the approximations made in our choice of U^A would become subject to question. This of course also applies in our SCF treatment as well.

D. The Potentials

The one- and two-particle effective potentials Σ^A and W^A appearing in our final equations in section II are hermitian adiabatic approximates to the true field theoretic potentials. As we have mentioned,

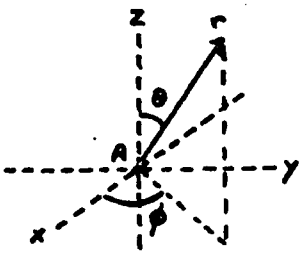
such potentials can be obtained in closed ab initio form using many-body theory within the framework of the RPA approximation and taking the adiabatic limits. However, while these potentials are tractable they are nonetheless quite complicated and their use would entail considerable computational effort. In view of the perturbative nature of our theory it is reasonable to expect that we could use potentials having simpler forms. Such simpler potentials can be obtained by making moment expansions of the RPA potentials and truncating these expansions in a physically meaningful manner [21]. When this is done, the resulting potentials can be cast in forms which are very similar to phenomenologically derived semiempirical potentials which have been used by other workers with considerable success [22]. Therefore, it would seem that the use of complicated ab initio potentials is not warranted (although we do reserve the option to do so) and that we can take our potentials to have semiempirical forms similar to those used by Dalgarno and by Victor [22], namely

$$\int dr'\ \Sigma^A(r;r') = -\frac{Z^A}{|r|} + \int dr'\ \Sigma^A_{HF}(r;r') - \frac{\alpha_d^A}{2|r|^4} W_6(k|r|) - \frac{\alpha_q^A}{2|r|^6} W_8(k|r|) + (a_0 + a_1|r| + a_2|r|^2)e^{-k|r|} \tag{22a}$$

and

$$\int dr_1'\, dr_2'\ W^A(r_1,r_2;r_1',r_2') = -\frac{\alpha_d^A}{|r_1|^2|r_2|^2} W_3(k|r_1|)W_3(k|r_2|)P_1(\cos\gamma_{12}) - \frac{\alpha_q^A}{|r_1|^3|r_2|^3} W_4(k|r_1|)W_4(k|r_2|)P_2(\cos\gamma_{12}) \tag{22b}$$

where we have chosen our coordinate system to be centered on A which for simplicity we now take to be an atom having a nuclear charge of z^A



and where

$W_n(\chi) = (1-e^{-\chi^n})$ is a cutoff function

γ_{12} = angle between vectors r_1 and r_2

$P_\ell(\chi)$ = legendre polynomial of the ℓ^{th} degree

α_d^A = dipole polarizability of A

α_q^A = approximate quadrupole polarizability of A (adjustable)

k = approximately 1/2 r_o where r_o is the effective radius of A (adjustable)

$\{a_i\}$ = adjustable monopole parameters

and

$$\Sigma^A_{HF}(r;r') = \sum_{i=1}^{n_A} \frac{\phi_i^{A*}(r')[2-P_{r,r'}]\,\phi_i^A(r')}{|r-r'|}$$

is the static Hartree-Fock potential of A with $\{\phi_i^A\}$ being the n_A spatial Hartree-Fock orbitals for the electrons in A . $P_{r,r'}$ is the permutation operator if r is an electronic coordinate whereas $P_{r,r'} = 0$

if r is a nuclear coordinate. Note that all quantities are now purely spatial and that Σ^A and W^A are therefore explicitly spin independent.

In (22a) the first term is simply the potential due to the nucleus of A and the second term is the static Hartree-Fock potential for the electrons in A occupying the spatial orbitals $\{\phi_i^A\}$. The next two terms in (22a) are asymptotically correct induced dipole and quadrupole polarization potentials which die off rapidly at short distances from A. The final term is an induced monopole term which serves as a short range correction potential. The terms in (22b) describe an asymptotically correct dielectric potential which properly cancels out one-particle induced dipole and quadrupole polarizations of A due to two particles of the same charge when these particles are on opposite sides of A. That these potentials represent a significant simplification over the ab initio potentials is clear in that our two-particle potential is strictly local and the only nonlocal term in the one-particle potential is simply the usual Hartree-Fock exchange potential. However, despite their simplicity, potentials such as these have been used very successfully for a variety of problems in the past and should therefore be quite adequate for our purposes.

As it stands, (22b) is concise and to the point. However, one important property is obscured. This is that our two-particle potential can be written in terms of one-particle operators only. To see this we need only express the angle γ_{12} in terms of the spherical angles

for each vector. Thus, by making use of the expansion

$$P_\ell(\cos \gamma_{12}) = P_\ell(\cos \theta_1)P_\ell(\cos \theta_2) + 2 \sum_{m=1}^{\ell} \frac{(\ell-m)!}{(\ell+m)!} P_\ell^m (\cos \theta_1)P_\ell^m(\cos \theta_2)$$

$$[\cos m\phi_1 \cos m\phi_2 + \sin m\phi_1 \sin m\phi_2] \qquad (24)$$

where $P_\ell^m(\chi)$ is an associated Legendre polynomial, we find that (22b) can be rewritten as

$$\int dr_1' \, dr_2' \, W^A(r_1,r_2;r_1',r_2') = - \sum_{i=1}^{8} Q_i(r_1)Q_i(r_2) \qquad (25a)$$

where

$$\begin{aligned}
Q_1(r) &= \sqrt{\alpha_d^A} \, |r|^{-2}W_3(k|r|)P_1(\cos \theta) \\
Q_2(r) &= \sqrt{\alpha_d^A} \, |r|^{-2}W_3(k|r|) \, P_1^1(\cos \theta) \cos \phi \\
Q_3(r) &= \sqrt{\alpha_d^A} \, |r|^{-2}W_3(k|r|) \, P_1^1(\cos \theta) \sin \phi \\
Q_4(r) &= \sqrt{\alpha_q^A} \, |r|^{-3}W_4(k|r|) \, P_2(\cos \theta) \qquad (25b) \\
Q_5(r) &= 1/3\sqrt{\alpha_q^A} \, |r|^{-3}W_4(k|r|) \, P_2^1(\cos \theta) \cos \phi \\
Q_6(r) &= 1/3\sqrt{\alpha_q^A} \, |r|^{-3}W_r(k|r|) \, P_2^1(\cos \theta) \sin \phi \\
Q_7(r) &= 1/12\sqrt{\alpha_q^A} \, |r|^{-3}W_4(k|r|) \, P_2^2(\cos \theta) \cos 2\phi \\
Q_8(r) &= 1/12\sqrt{\alpha_q^A} \, |r|^{-3}W_4(k|r|) \, P_2^2(\cos \theta) \sin 2\phi
\end{aligned}$$

Consequently, the implementation of this method involves the addition of only one-electron integrals to existing SCF or CI programs. In addition, if Σ_{HF}^{A} is replaced by a pseudopotential, only the atomic orbitals of B will be used to calculate the potential for AB.

4.II Details of the Calculation

In developing a program package to implement the effective potential method, our goal was to make the package as general as possible. To achieve this, we have made our effective potential program EFFPOT compatible with the GVB II[10] SCF and CI programs. This program package is a gaussian basis function, molecular SCF and CI routine which will enable the present version of EFFPOT to be extended to large molecules. The version of GVB II that we are using has an option for replacing the atomic cores by a pseudopotential; however, the form of this built-in pseudopotential is different from those being developed for large atoms[11]. The modifications needed to make use of available pseudopotentials would be relatively easy.

The GVB II progam package consists of a number of sequential program steps which are outlined below.

1. Integral generation	a) PLYLABS -	creates list of unique integral labels
	b) PLYINTS -	evaluates integrals
2. Integral preprocessing	a) PLYIJLK b) PLYPAIR	convert output of PLYINTS to form needed for input to GVB II
3. SCF	GVB II	
4. CI	a) CORTRN	transforms to molecular basis
	b) CIPROG	

To perform an effective potential calculation, the program EFFPOT is inserted between steps 2a and 2b.

EFFPOT calculates the additonal integrals required, $\langle\phi_i|\Sigma^A|\phi_j\rangle$ and $\langle\phi_i|Q_n|\phi_j\rangle$, and modifies the integral tape so that the insertion of an effictive potential is transparent to GVB II. These integrals have not been previously evaluated for gaussian basis sets, and because of the cut-off functions in Σ^A and Q_n and the angular dependence of Q_n, the evaluation of these integrals is rather complicated, especially for two-center integrals containing basis functions for $\ell > 0$.

Although it is relatively easy to construct an algorithm for evaluating these integrals, the algebraic detail rapidly becomes overwhelming. To alleviate this problem, we used the algebraic programming system REDUCE[3] to produce the FORTRAN code required to evaluate these integrals. An explanation of the algorithm used to derive the integral expressions and an example of the input to REDUCE are given below for one of the more complicated integrals.

The integral we shall consider is

$$\langle P_x^E | Q_4^B | P_x^F \rangle \tag{13}$$

where the superscripts indicate the atomic center on which the (basis or potential) function is centered, Q_4 is defined by equation 10b and P_x indicates a "p_x" gaussian basis function. This integral can be written explicitly as

$$\langle P_x^E | Q_4^B | P_x^F \rangle =$$

$$\sqrt{\alpha_q^A/3}\int d\tau\, x_E\, e^{-\alpha r_E^2}\, r_D^{-3}\, W_4(kr_D)\, P_2(\cos\theta_D)\, x_F\, e^{-\beta r_F^2}$$

$$= \sqrt{\alpha_q^A/3}\ I. \tag{14}$$

Using the relation

$$P_2(\cos\theta_B) = \frac{3z_B^2 - r_B^2}{2r_B^2}\ , \tag{15}$$

we can write

$$I = \frac{3}{2}\int d\tau\, x_E e^{-\alpha r_E^2} W_4(kr_B) z_B^2 r_B^{-5} x_F e^{-\beta r_F^2} - \frac{1}{2}\int d\tau\, x_E e^{-\alpha r_E^2} W_4(kr_B) r_B^{-3} x_F e^{-\beta r_F^2}$$

$$\equiv I_1 + I_2 . \tag{16}$$

In order to obtain the term z_B^2, we will set up a fake gaussian on center B, $e^{-cr_B^2}$, and move the $W4(kr_B)/r_B^5$ term to a fake center, D; after evaluating the resulting integral, we will take the limits $D \to B$ and $c \to 0$. Using the relation

$$z_B^2 e^{-cr_B^2} = \frac{1}{2^2c^2}\frac{\partial^2}{\partial B_z^2} e^{-cr_D^2} + \frac{1}{2c} e^{-cr_D^2}\ , \tag{17}$$

we obtain

$$I_1 = \lim_{c\to 0}\lim_{D\to B} \frac{3}{2}\frac{1}{2^2\alpha\beta}\frac{\partial^2}{\partial E_x \partial F_x}\left\{\frac{1}{2^2c^2}\frac{\partial^2}{\partial B_z^2} I_3 + \frac{1}{2c} I_3\right\} \tag{18}$$

where

$$I_3 = \int d\tau\, e^{-\alpha r_E^2} e^{-cr_D^2} r_D^{-5} W_4(kr_D) e^{-\beta r_F^2}. \tag{19}$$

We can also write

$$I_2 = \frac{-1}{2^3\alpha\beta}\frac{\partial^2}{\partial E_x \partial F_x} I_4 \tag{20}$$

where

$$I_4 = \int d\tau\, e^{-\alpha r_E^2} e^{-\beta r_F^2} r_B^{-3} W_4(kr_B) . \tag{21}$$

We can then apply the rule for combining gaussians on two centers,

$$e^{-\alpha r_E^2} e^{-\beta r_F^2} = e^{-\frac{\alpha\beta}{\gamma} R_{EF}^2} e^{-\gamma r_A^2} \tag{22}$$

where $\gamma = \alpha + \beta,$ (22a)

REF is the distance between centers E and F, and center A is defined by

$$A_x = (\alpha E_x + \beta F_x)/\gamma. \tag{22b}$$

We then use the Fourier Convolution Theorem to obtain the following expressions for I_3 and I_4:

$$I_3 = e^{-\frac{\alpha\beta}{\gamma} R_{EF}^2} e^{-\frac{\gamma c}{\delta} R_{AD}^2} I_5 \tag{23}$$

where

$$I_5 = \int d\tau\, e^{-\delta r_G^2} r_D^{-5} W_4(kr_D) \tag{24}$$

$$= \frac{\pi}{\delta R_{DG}} \int_0^\infty dr\, r^{-4} W_4(kr) \left\{ e^{-\delta(r - R_{DG})^2} - e^{-\delta(r + R_{DG})^2} \right\},$$

and

$$I_4 = e^{-\frac{\alpha\beta}{\gamma} R_{EF}^2} I_6 \tag{25}$$

where

$$I_6 = \int d\tau\, e^{-\gamma r_A^2} r_D^{-3} W_4(kr_B) \tag{26}$$

$$= \frac{\pi}{\delta R_{AB}} \int_0^\infty dr\, r^{-2} W_4(kr) \left\{ e^{-\gamma(r - R_{AB})^2} - e^{-\gamma(r + R_{AB})^2} \right\}.$$

In these equations $\delta = \gamma + c$ and G is the center obtained by combining Gaussians on A and D. The derivatives and limits in equation 18 are obtained analytically by REDUCE and the result is that equation 18 is written as a sum of integrals with forms similar to those given by equations 24 and 26. These basic integrals are evaluated numerically and summed to give the result for $\langle P_x^E | Q_4^B | P_x^F \rangle$. To obtain expressions for this integral in the case where any of centers E,F,B coincide, REDUCE is used to obtain the appropriate limit ($E \rightarrow F$, for example). The REDUCE commands used to evaluate all Q_4 integrals are given in Appendix B. The output from this REDUCE code is rearranged, using additional REDUCE commands, to produce FORTRAN code which can be inserted directly in the integral evaluation subroutines.

The present versions of the EFFPOT integral routines are restricted to two centers and s or p basis functions. Our tests of this program on LiHe are described in the next section.

The integration itself is done by a 32 point Gauß integration with the IBM library program DQG 32. The other FORTRAN-programs of the EFFPOT routine, developed by ourselves, are contained in Appendix C.

4.III Test Calculations of EFFPOT on LiHe

In this section we present the results of the first stage of our tests of the effective potential method. We have chosen LiHe as the test system, and in this stage of the tests we assume that the parameters of our semi-empirical potential are known.

Values of α_d and α_q can easily be obtained; however, it should be pointed out that in a semi-empirical potential, each parameter plays a dual role. Besides describing the physical effect to which it most obviously corresponds, each parameter also serves to correct for the deficiencies of the semi-empirical model. Consequently in the best fit for the potential parameters, the values of α_d, α_q, etc., should not be expected to equal the physical quantities. The values for the cut-off functions and for the short-range part of the potential are harder to obtain. Our initial desire was to use the parameters obtained by Peach.[12] She calculated the parameters for a model potential describing one electron outside of He by fitting the parameters to electron-He atom scattering data. Her model also included a pseudopotential to represent the Hartree-Fock potential of He; this fact makes it impossible to separate the pseudopotential from the short-range terms needed for the effective potential (see equation 7). Consequently, for our test calculations, we use Peach's values for α_d, α_q', β and her cut-off functions. In addition, we have guessed a value for a_o (equation 7), so that the short-range potential has only one term.

The exact forms of the potentials used in our test calculations are

$$\int dr' \Sigma^A(r;r':z) = \frac{Z^A z}{r} + z\int dr' \Sigma^A_{HF}(r;r') + z^2 a_0 e^{-\gamma r} - \frac{\alpha_d z^2}{2r^4} W_2(\beta r) - \frac{\alpha' z^2}{2r^6} W_3(\beta' r) \quad (27)$$

$$\int dr_1' dr_2' W^A(r_1,r_2;r_1',r_2':z_1,z_2) = \frac{-\alpha_d z_1 z_2}{2 r_1^2 r_2^2} X_2(\beta r_1) X_2(\beta r_2) P_1(\cos\gamma_{12}) - \frac{\alpha_q' z_1 z_2}{2 r_1^3 r_2^3} X_3(\beta r_1) X_3(\beta r_2) P_2(\cos\gamma_{12}) \quad (28)$$

Where W_n is defined by

$$W_n(x) = [X_n(x)]^2 \quad (29a)$$

and

$$X_n(x) = 1 - \exp(-x)\sum_{m=0}^{n} \frac{x^m}{m!} \underset{x\to 0}{\approx} \frac{x^{n+1}}{(n+1)!} \quad (29b)$$

$$\alpha' = (\alpha_q' - 6\beta_1/M).$$

The values of the parameters used are given in table 1.

In tables 2 and 3 and figures 1 and 2, we compare the effective potential-SCF (EP-SCF) calculations with SCF[13] and restricted CI calculations for the same basis set. The basis set is that of Krauss, Maldonado and Wahl[13]. The restricted CI calculations presented here have the same asymptotic limit as the EP-CI calculations, but do not fully allow for distortion of He at intermediate distances. In comparing these results, it should be noted that the EP-SCF results have the same $R \to \infty$ limit as the SCF calculation, but that at intermediate distances, the EP-SCF results go as $1/R^6$ while the SCF calculations cannot give this dependence. Also, the full CI calculations, which allow for more He polarization, should give more attractive interaction

Table 1. Parameters for the potential defined by equations 27 to 29, for He[a]

Z_a	2.
a_o	.05
γ	1.59
α_d	1.3834
β	2.09928
α'	-2.1222
β'	0.551429
α'_q	2.11380

[a]All the parameters except a_o and γ are taken from Peach.[12]

Table 2. LiHe total energies (in Hartrees)

	EP-SCF		SCF[a]		CI (restricted)	
R(a.u.)	$^2\Sigma$	$^2\Pi$	$^2\Sigma$	$^2\Pi$	$^2\Sigma$	$^2\Pi$
2.	-10.19137	-10.14751	-10.18797	-10.14029	-10.20229	-10.15458
3.	-10.27436	-10.22475	-10.27465	-10.22372	-10.29037	-10.23938
4.	-10.28454	-10.22723	-10.28397	-10.22646	-10.29993	-10.24232
5.	-10.28933	-10.22603	-10.28840	-10.22546	-10.30449	-10.24138
6.	-10.29166	-10.22533	-10.29113	-10.22500	-10.30722	-10.24090
8.	-10.29321	-10.22486	-10.29304	-10.22477	-10.30911	-10.24067
10.	-10.29339	-10.22476	-10.29335	-10.22473	-10.30944	-10.24065

[a] Krauss, Maldonado, and Wahl[13]

Table 3. LiHe interaction enrgies (in Hartrees)

	EP-SCF		SCF[a]		CI(restricted)		Full CI	
R(a.u.)	$^2\Sigma$	$^2\Pi$	$^2\Sigma$	$^2\Pi$	$^2\Sigma$	$^2\Pi$	$^2\Sigma$	$^2\Pi$
2.	.10201	.07735	.10541	.08457	.10723	.09608		
3.	.01902	-.00004	.01873	.00099	.01915	.00128	.02000	
4.	.00884	-.00252	.00941	-.00175	.00959	-.00166		
5.	.00405	-.00132	.00498	-.00075	.00503	-.00072		
6.	.00172	-.00062	.00225	-.00029	.00230	-.00024		
8.	.00017	-.00015	.00034	-.00006	.00041	-.00001		
10.	-.00001	-.00005	.00004	-.00002	.00008	.00001		

[a] Krauss, Maldonado, and Wahl[13]

Figure Captions

1. Interaction energy for LiHe $^2\Sigma$ state.

   ⊙ connected by solid line; EP-SCF results.

   ⊡ SCF results of Krauss, Maldonado, and Wahl[13].

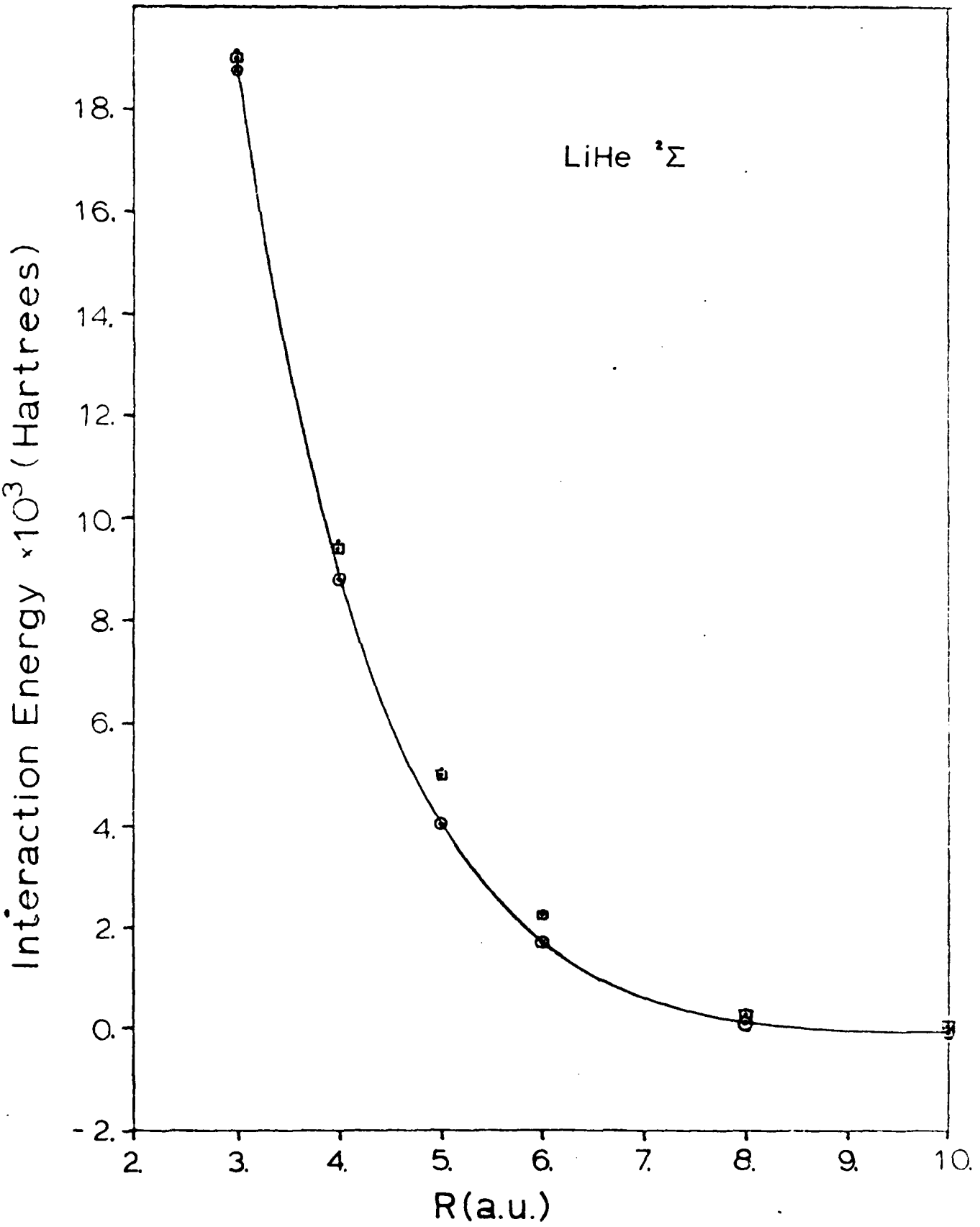
   · restricted CI calculation.

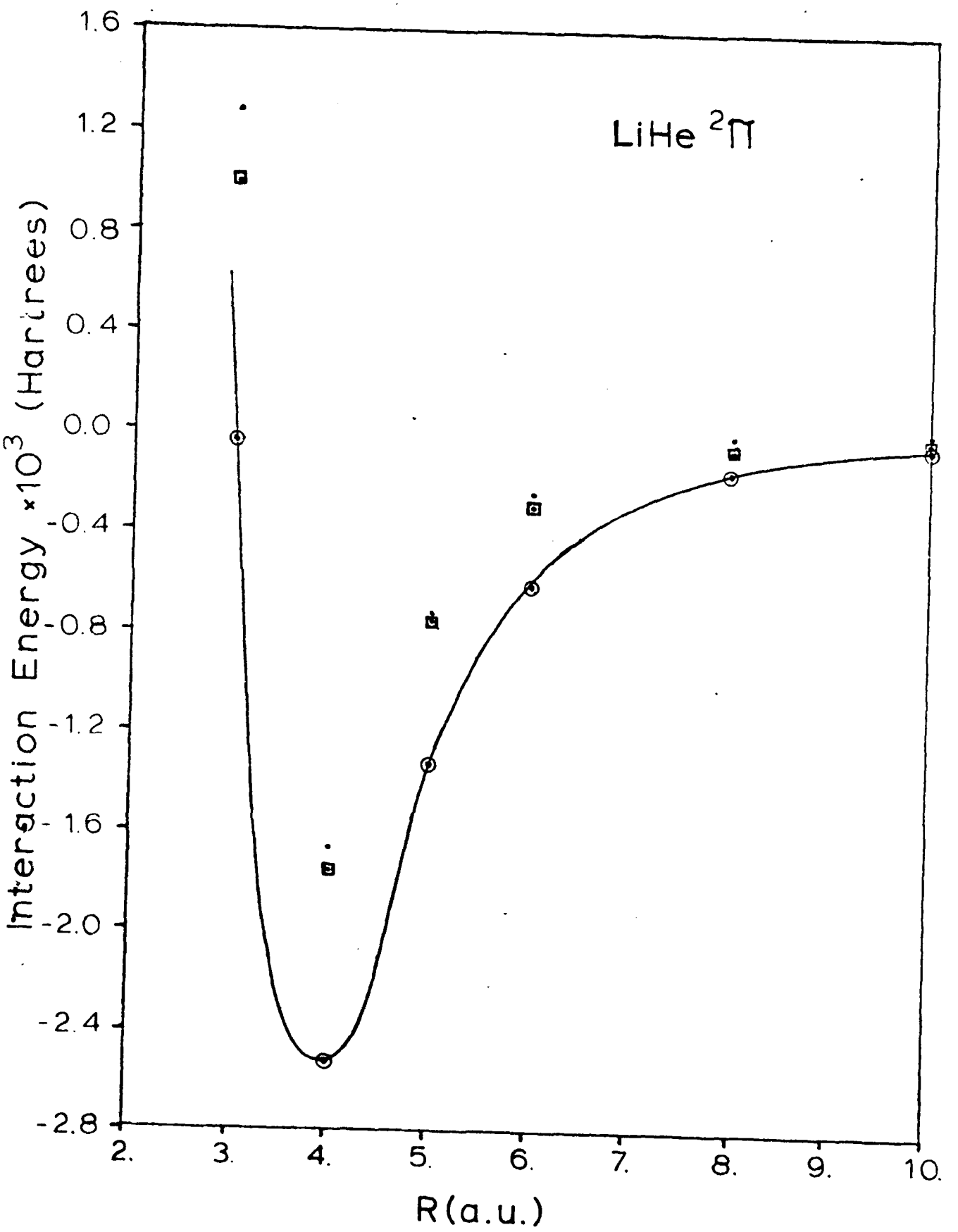
2. Interaction energy for LiHe $^2\Pi$ state.

   ⊙ connected by solid line; EP-SCF results.

   ⊡ SCF results of Krauss, Maldonado, and Wahl[13].

   · restricted CI calculation.


LiHe $^2\Sigma$
Interaction Energy $\times 10^3$ (Hartrees)
18.
16.
14.
12.
10.
8.
6.
4.
2.
0.
-2.
2.
3.
4.
5.
6.
7.
8.
9.
10.
R(a.u.)


LiHe $^2\Pi$
Interaction Energy $\times 10^3$ (Hartrees)
1.6
1.2
0.8
0.4
0.0
-0.4
-0.8
-1.2
-1.6
-2.0
-2.4
-2.8
2.
3.
4.
5.
6.
7.
8.
9.
10.
R(a.u.)

4.IVa Test calculations of the fit program

In fig. 2 in the last paragraph it has become obvious that the initial values for the parameters in the effective potential (the ones given in table 1 in section III, let us call this set of parameters p_o in the following) do not give a sufficiently good agreement between the calculated molecular potential curve and the curve of our CI calculation or the results of Krauss et al. One method to find a better fit would be to repeat the calculations for a large number of parameter values and so to calculate a potential curve for each point on a dense grid in the 10-dimensional parameter space. This would be the safest method to come close to the best possible fit but it would consume an enormous amount of computer time and therefore it would be contrary to our initial goal. A faster alternative is, to apply a fit program which uses as input the results of section III and tries to improve the molecular potential curve by some kind of first order perturbation calculation.

In order to explain what kind of fitting method has been used here let us for this section introduce the following notation. p is a set of 10 parameters which go into the effective potential, W(p,R) is the complete effective potential for the set p of parameters and for the nuclear distance R, $\Psi(p,R)$ is the exact electronic SCF wavefunction for the nuclear distance R and for the effective potential W(p,R) calculated according to the method given in section III. V(p,R) is the molecular potential calculated according to the method given in section III with the effective potential W(p,R) and at a nuclear distance R.

The idea of our first order perturbation approach is to calculate first the difference between the effective potentials for two sets

of parameters p_o and p , then calculate the expectation value of this difference potential with the exact wavefunction belonging to parameter set p, and interpret this expectation value as difference in the molecular potential. In formula

$$V(p , R) = V(p_o,R) + \langle\psi(p_o,R)|W(p,R) - W(p_o,R)|\Psi(p_o,R)\rangle \qquad (30)$$

as justification for the application of a first order perturbation method we have first to show that the results of the exact calculation are highly linear at variation of a parameter in the effective potential. In fig. 3 the electronic energy has been calculated with the method given in section III for various values of the dipole parameter α_d and always the same values for the other parameters at the same nuclear distance of R = 3.5. It can be seen that the dependence of the energy on the parameter α_d is highly linear. Similar results hold for the other parameters. These calculations have also shown that the wavefunction does not change significantly under a change of a parameter. Therefore it seems that a first order perturbation approach to the fitting problem is justified. One of the inputs in any fitting program must be a value to which we want to come close. In the case here, the assumed true molecular potential curve for the He-Li $^2\pi$ state is the one given by Krauss et al. We denote it in the following by VK(R). In addition, it is necessary to define the quantity, a measure of the error, which should be minimized.

We have chosen as measure of the error of the molecular potential curve V(p,R) the following quantity

$$D(p) = \sum_i [V_K(R_i) - V(p,R_i)]^2/V_K^2(R_i). \qquad (31)$$

The sum i goes over all those nuclear distances at which we try to improve the function V(p,R) at the moment.

This fitting for the relative error is a disadvantage in so far, that those nuclear distances R_i at which $V_K(R_i)$ is small have a higher weight than those distances at which $V_K(R_i)$ is big. Therefore the computer tries to fit $V(p,R_i)$ with the higher priority near zeros of $V_K(R_i)$.

The fitting program and all subroutines which belong to it and all those programs which are necessary in addition to the previous programs are shown in Appendix D.

Our experience running this fitting program can be summarized as follows: The fitting program is started with a particular set p_i of 10 parameters and in its 0th step it calculates the quantity $V(p_i,R)$ according to (30). For this initial step it takes about 6 minutes times the number of points, on which the curve is fitted, to calculate the integrals. Such a long time is necessary because up to now there are no optimised standard programs of quantum chemistry available for these types of integrals. It takes a long time to transform the expressions of the functions to be integrated into such a form that a library integration program can be applied. (The 32 point Gaussian integration program DQG32 by IBM ultimately has been used. If in a next step one of the parameters α_d, α_q, β, a_o, a_1, a_2, a_3 is changed then it is not necessary to evaluate the integrals again because these 7 parameters do not change the shape of W(p,R). They go linearly into W(p,R) and also the corresponding integrals are linear in these parameters. Therefore it takes only a few seconds to calculate a new $V(p_j,R)$ when p_j differs from p_i only in a parameter which goes into W(p,R) linearly. The situation

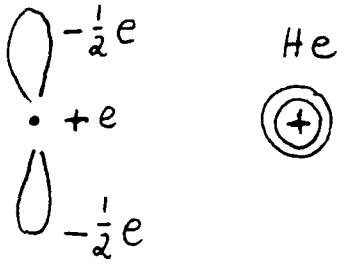
is completely different if one of the cut-off parameters is changed. Then the shape of the function W(p,R) is different and it is necessary to calculate from the start the corresponding integrals again. This takes at least several minutes time. Hence it is quite time consuming to fit cut-off parameters by this fitting program. Therefore we have done the fit of cut-off parameters partly by hand.

In Fig. 4 the results of the fit can be seen for two sets of parameters which gave the best fits we could obtain for the molecular potential curve of the $^2\pi$ state of HeLi. For comparison we show in addition the curve calculated by Krauss et al. as solid line. This curve agrees quite well with our CI calculations (see Fig. 2 in section III). The result of the SCF calculations from section III is shown as dotted line also for comparison. This line is the molecular potential curve which belongs to the parameter set p_o and it is the Oth order result of the fitting procedure in this section. The broken line and the chain line show the 1st order results for two different sets of parameters. It is obvious that the deviation between these first order curves and the Krauss et al. curve is a lot less than the deviation between the Oth order curve and the Krauss et al. curve. The worst error of our 1st order curves is in the region near R=5 to R=6. In this region it has not been possible to get a curve which is steep enough. All considered 1st order curves with a reasonable over all shape have been too horizontal in this interval. Up to now we did not yet have time to make calculations for molecular potential curves of other states of LiHe than the $^2\pi$ state. Before we do this, it would be useful to try out a new fitting strategy described in the next section.

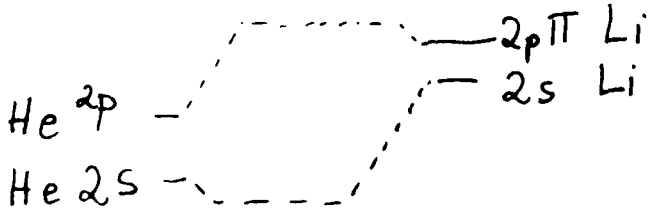
IVb. A strategy to achieve a better fit to the π state of LiHe

As was seen in IVa we did not get an ideal fit in the R=5 region as we did not seem to get the fitted curve to be steep anough. It is to be noted that so far we have fitted to Krauss et al. LiHe, π state. It is also to be noted that we have not exhausted, because of lack of computer time, the fit routine and tried to guess the best parameter regions. In this subsection we suggest a new strategy which we plan to carry out in the fall of 1980 when U.S.C. will replenish our computer budget. The strategy is based on the comparison of the shape of the ground $LiHe^+$ potential curve (which goes asymptotically to Li^+ +He) to that of the SCFπLiHe curve (which dissociates properly) as given in the figures in Krauss et al. In this we see the striking fact that the two curves are parallel in the R=5 to 6 region and almost parallel at short range; the LiHe has a slightly smaller Re. This suggests the much of the curve in π LiHe is like the $X^1\Sigma LiHe^+$ and this is clearly charge-induced dipole at large distance (R>4.5) as it goes at R^{-4} and Li^+ - Helium repulsion at short distance. Now the former effect is in our model due to the α/r^4 term and the latter in the static and monopole term. (in which e^{-ar}((a chosen so that the term effects the very short range part of the curve only)) is always automatically fit with a negative coefficient). This suggests that as a first run at fitting the π, that we work with the ionic state (a simpler problem) and set our quadrupole term to zero and perhaps again start with the physical value of the polarizability for Helium. This simpler system should be able to be fit. The ionic system then determines the α, its short range cut-off parameters and will give a first guess of the monopole term.

Now we will return to the π Li-He fit, and retain the $LiHe^+$ parameters as a start. We will then work at large R, R>7 only, fit the quadrupole term and its cut-off parameters. This is suggested again by the $^2\pi$ state to $X^1\Sigma$ ionic state comparison figure in Krauss where we see that at large R the LiHe system deviates from the ionic system and is less attractive, i.e. the effect of a R^6 comes in. This effect can only be found in the π state as at large R, the Li P state can be reviewed roughly as



i.e. Helium sees a $\{\frac{-}{+}\}$ charge distribution and the L π orbit feels the Helium quadrupole moment. The next step will be to now fit again at small R, the monopole term, starting with our ionic result. The small shift must be due to a chemical effect which can be described as:



We should be able to account for this with our monopole parameter.

Hopefully this new strategy will yield a better fit with little computing. The effective potential will then be used to construct the $X^2\Sigma$ state.

Failure in fitting by the procedure of this subsection or in the construction of the Σ state will forebode badly for the ultimate success of this theory.

Figure Captions

Figure 3:

Plot of the electronic energy for the He-Li $^2\pi$ state calculated according to the method given in section III for various values of the paramter α_d and otherwise always the parameters given in table 1 in section III. The nuclear distance is always R=3.5.

Figure 4:

Plot of potential curves for the He-Li $^2\pi$ state calculated with various parameters for the effective potential and the curve given by Krauss et al. for comparison.

Solid line: result of Krauss et al.

Dotted line: SCF calculation from section III with parameter set p_o

Broken line: calculation with the method given in section IV and parameter values

$\alpha_d = 0.8$
$\alpha_q = 1.65$
$\beta = 1.9$
$c_1 = 4$
$c_2 = 1.3$
$c_3 = 0.6$
$a_o = -0.8$
$a_1 = 0$
$a_2 = 0$
$a_3 = 0$

Chain line: calculation with the method given in section IV and parameter values

$a_d = 0.9$

$\alpha_q = 1.75$

$a_o = -0.46$

$a_1 = 0$

$\beta = 1.56$

$c_1 = 4$

$a_2 = 0$

$c_2 = 1.3$

$c_3 = 0.6$

$a_3 = 0$

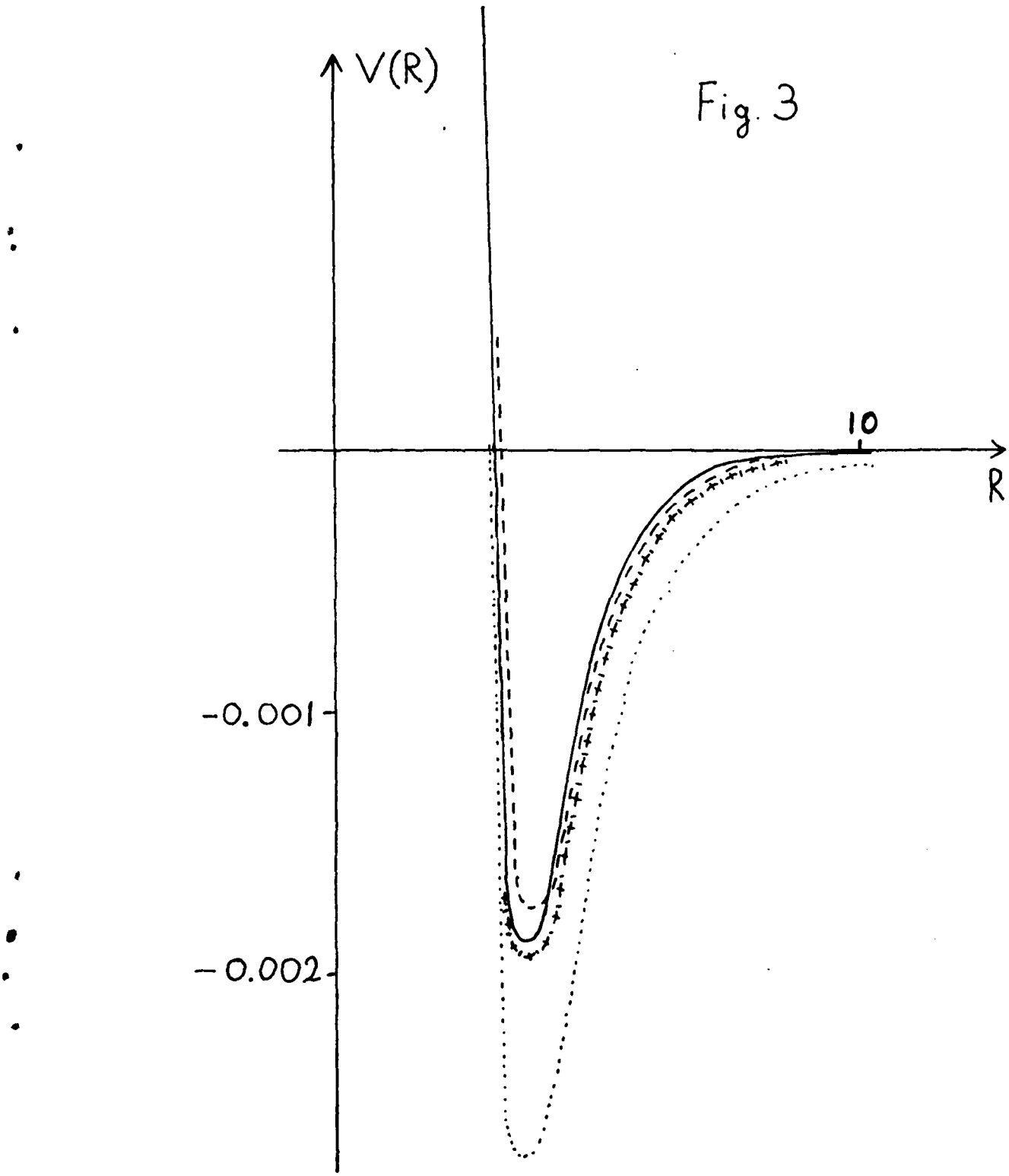


Fig. 3

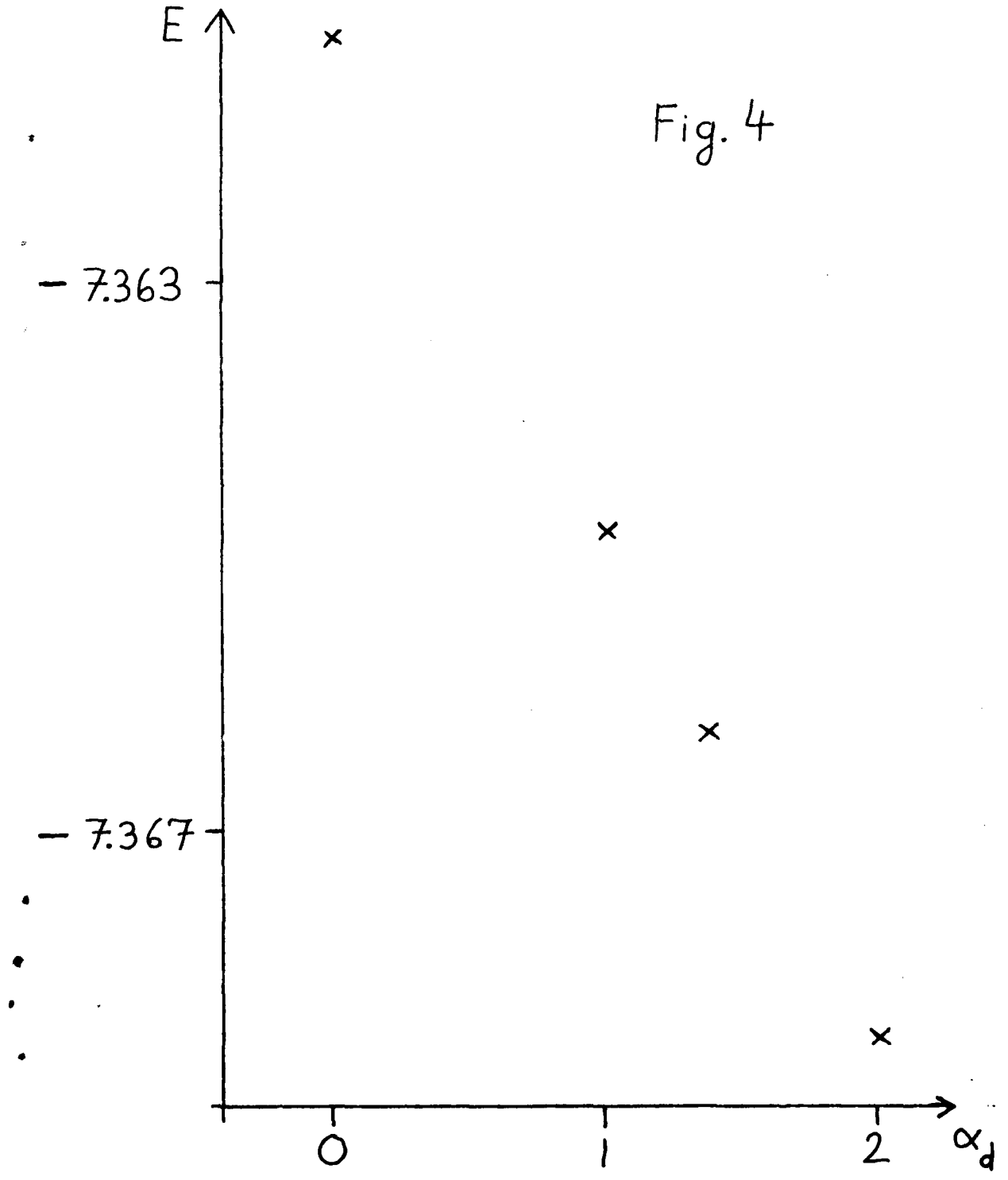


Fig. 4

References to §4

1. H.S. Taylor, M. Valley, C. Watts, and F. Bobrowicz, "Excimer potential curves," Annual Report No. 1, ARPA contract no. N00014-77-C-0102, Feb. 1, 1978.

2. T.H. Dunning, Jr., M. Valley and H.S. Taylor, J.Chem Phys. 69, 2672 (1978).

3. A.C. Hearn, REDUCE 2 Users's Manual, Second Edition, 1974.

4. B. Schneider, H.S. Taylor and R. Yaris, Phys. Rev. A1, 855 (1970), G. Csanak and H.S. Taylor, Phys. Rev. A6, 1843 (1972).

5. B.S. Yarlagadda, G. Csanak, H.S. Taylor, B. Schneider and R. Yaris, Phys. Rev. A7, 146 (1973); L.D. Thomas, G. Csanak, H.S. Taylor and B.S. Yarlagadda, J. Phys. B7, 1719 (1974).

6. B. Schneider, H.S. Taylor, R. Yaris, and B.S. Yarlagadda, Chem. Phys. Letters 22, 381 (1973).

7. G. Csanak and H.S. Taylor, Phys. Rev. A6, 1843 (1972); G. Csanak and H.S. Taylor, J. Phys. B6, 2055 (1973).

8. S.W. Wang, H.S. Taylor and R. Yaris, Chem. Phys. 14, 53 (1976).

9. A. Dalgarno, C. Bottcher and G.A. Victor, Chem. Phys. Lett. 7, 265 (1970), C. Bottcher and A. Dalgarno, Proc. R. Soc. Lond. A 340 187 (1974); D.K. Watson, C.J. Cerjan, S. Guberman and A. Dalgarno, submitted to Chem. Phys. Lett. 1977.

   C. Laughlin and G.A. Victor, Atomic Physics 3, 247 (1973).

10. N.W. Winter, F.W. Bobrowicz and W.A. Goddard III, J. Chem. Phys. 62 4325 (1975).

11. Y.S. Lee, W.C. Ermler and K.S. Pitzer, J. Chem. Phys. 67, 5861 (1977) W.R. Wadt, P.J. Hay and L.R. Kahn, J. Chem. Phys. 68 1752 (1978).

12. G. Peach, J. Phys. B 11, 2107 (1978).

13. M. Krauss, P. Maldonado and A.C. Wahl, J. Chem. Phys. 54, 4944 (1971).

16. B. Schneider, H. S. Taylor and R. Yaris, Phys. Rev. A1, 855 (1970), G. Csanak and H. S. Taylor, Phys. Rev. A6, 1843 (1972).

17. P. J. Hay, T. H. Dunning, Jr. and R. C. Raffenetti, J. Chem. Phys. 65, 2679 (1976); W. J. Stevens and M. Krauss, J. Chem. Phys. 67, 1977 (1977).

18. B. S. Yarlagadda, G. Csanak, H. S. Taylor, B. Schneider and R. Yaris, Phys. Rev. A7, 146 (1973); L. D. Thomas, G. Csanak, H. S. Taylor and B. S. Yarlagadda, J. Phys. B7, 1719 (1974).

19. B. Schneider, H. S. Taylor, R. Yaris and B. S. Yarlagadda, Chem. Phys. Lett. 22, 381 (1973).

20. G. Csanak and H. S. Taylor, Phys. Rev. A6, 1843 (1972); G. Csanak and H. S. Taylor, J. Phys. B6, 2055 (1973).

21. S. W. Wang, H. S. Taylor and R. Yaris, Chem. Phys. 14, 53 (1976).

22. A. Dalgarno, C. Bottcher and G. A. Victor, Chem. Phys. Lett. 7, 265 (1970), C. Bottcher and A. Dalgarno, Proc. R. Soc. Lond. A 340 187 (1974); D. K. Watson, C. J. Cerjan, S. Guberman and A. Dalgarno, submitted to Chem. Phys. Lett. 1977; C. Laughlin and G. A. Victor, Atomic Physics 3, 247 (1973).

Appendix A

Theoretical studies of the low-lying electronic states of GaKr, including extrapolation to InKr and TlKr[a)]

Thom. H. Dunning, Jr.[b)]

Chemistry–Nuclear Chemistry Division, Los Alamos Scientific Laboratory, Los Alamos, New Mexico 87545

Marcy Valley and Howard S. Taylor

Department of Chemistry, University of Southern California, Los Angeles, California 90007
(Received 25 April 1978)

We report *ab initio* configuration interfaction calculations on the states of the gallium krypton (GaKr) molecule arising from the $Ga(^2P_{1/2,3/2},{}^2S_{1/2}) + Kr(^1S_0)$ and $Ga^+(^1S_0) + Kr(^1S_0)$ separated atom limits. The potential energy curves for the states arising from the $Ga(^2P_{1/2,3/2})$ limits, the *I* 1/2, *II* 1/2, and *I* 3/2 states, are found to be repulsive. The potential energy curves for the states arising from the $Ga(^2S_{1/2})$ and $Ga^+(^1S_0)$ limits, the *III* 1/2 and 1 0 states, are both found to be weakly bound; D_e (*III* 1/2) = 0.047 eV and D(1 0) = 0.24 eV. The potential energy curves and transition moments obtained in the GaKr calculations have been used to simulate the curves and moments for InKr and TlKr. Using this data the absorption and emission coefficients of all three molecules have been calculated.

I. INTRODUCTION

The Group IIIA-rare gas excimer molecules are considered to be attractive candidates for developing an efficient, high power, tunable laser in the visible region of the spectrum.[1] The transition under consideration is a perturbed $(n+1)^2S_{1/2}$–$n^2P_{1/2,3/2}$ transition of the Group IIIA atom. While current experimental studies have concentrated on the thallium-rare gas systems, especially TlXe, as the most promising candidates, Gallagher[2] has recently raised the possibility of using the gallium-rare gas systems by obtaining gallium atoms from the dissociation of GaI_3.

To provide further information on the nature of the excimer states involved in these studies, we report here *ab initio* configuration interaction calculations on a prototype Group IIIA-rare gas diatomic molecule, GaKr. We present the potential energy curves for all of the states arising from the neutral $Ga(^2P_{1/2,3/2}, {}^2S_{1/2}) + Kr(^1S_0)$ and ionic $Ga^+(^1S_0) + Kr(^1S_0)$ separated atom limits and the dipole transition moments radiatively coupling the states. We then use the computed potential curves and transition moments for GaKr, along with the experimental atomic spin-orbit coupling constants, to model the curves and moments for InKr and TlKr. With this data we calculate the emission and absorption coefficients for all three systems using the classical technique developed by Gallagher and co-workers.[16]

II. ELEMENTARY THEORETICAL CONSIDERATIONS

Let us first consider the description of the states of the Group IIIA-rare gas molecules without spin-orbit corrections. The valence $Ga(^2P) + Kr(^1S)$ separated atom limit gives rise to a $^2\Pi$ state and a $^2\Sigma^+$ state, the Rydberg $Ga(^2S) + Kr(^1S)$ limit gives rise to another $^2\Sigma^+$ state and the ionic $Ga^+(^1S) + Kr(^1S)$ limit gives rise to a $^1\Sigma^+$ state. We will label the states which arise from the valence limit the $1\,^2\Pi$ and $1\,^2\Sigma^+$ states, that from the Rydberg limit the $2\,^2\Sigma^+$ state and that from the ionic limit the $1\,^1\Sigma^+$ state.

The orbital diagrams for the $1\,^2\Pi$, $1\,^2\Sigma^+$, and $1\,^1\Sigma^+$ states of a Group IIIA-rare gas molecule are given in Fig. 1. From these diagrams it is evident that none of the resulting potential energy curves are expected to be chemically bound. In the ionic state, however, weak binding can result from the charge-induced dipole interaction of the Group IIIA ions and the rare gas atoms. In fact, this interaction might also be expected to give rise to a weak binding in the $2\,^2\Sigma^+$ state. Since the Rydberg orbital of the Group IIIA atom is diffuse, the approaching rare gas atom can easily polarize the Rydberg orbital out of the

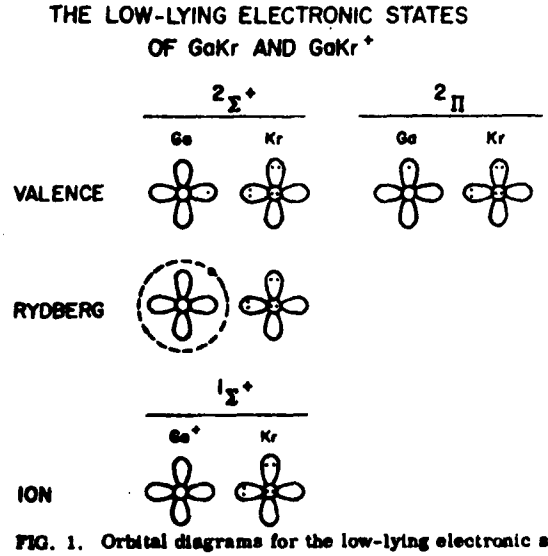


FIG. 1. Orbital diagrams for the low-lying electronic states of GaKr and GaKr$^+$. The two lobed figures represent 4*p* orbitals in the plane of the paper; the circle represents a 4*p* orbital perpendicular to the plane of the paper; the 5*s* Rydberg orbital is represented by a large dashed circle.

[a)]Research supported in part by the U. S. Department of Energy and by the Advanced Research Projects Agency under contract N00014-77-C-0102.

[b)]Present address: Theoretical Chemistry Group, Chemistry Division, Argonne National Laboratory, Argonne, Illinois 60439.

 0021-9606/78/6906-2872$01.00

interatomic region, thus partially unshielding the Group IIIA ionic core. The binding in the $2\,^2\Sigma^+$ state should, of course, be substantially less than that in the $1\,^1\Sigma^+$ state.

For the valence states the repulsiveness of the curves should be roughly proportional to the number of $p\sigma$ electrons since the overlap of the two atomic charge distributions is dominated by the overlap of the $p\sigma$ orbitals. Thus, the potential curve for the $1\,^2\Pi$ state with two $p\sigma$ electrons should be less repulsive than that for the $1\,^2\Sigma^+$ state with three $p\sigma$ electrons.

As we shall later see, at short internuclear separations the energies of the $1\,^2\Sigma^+$ and $2\,^2\Sigma^+$ states are nearly equal and there is a strong interaction between the two states. This mixing is strong in spite of the fact that one state is a valence state and the other a Rydberg state because the two *zero-order configurations* (see the next section) differ by only a single excitation.

A complete treatment of the electronic states of the Group IIIA-rare gas molecules must include the effects of spin-orbit coupling. Using the simple model developed in earlier papers for including the effects of spin-orbit coupling in diatomic molecules,[3] only the $1\,^2\Pi$ and $1\,^2\Sigma^+$ states are affected by the spin-orbit interaction, and the coupling arises solely from the spin-orbit interaction in the Group IIIA atom. For gallium the $^2P_{1/2}$–$^2P_{3/2}$ splitting is only 0.10 eV[4]; however, by thallium this splitting has increased to 0.97 eV.[4]

In a molecule only the z component of the total angular momentum,

$$\Omega = \Lambda + S_z \ ,$$

is a good quantum number. Thus, from the $\mathrm{Ga}(^2P_{1/2,3/2}) + \mathrm{Kr}(^1S_0)$ separated atom limits we obtain two $\Omega = 1/2$ states and one $\Omega = 3/2$ state (the *I* 1/2, *II* 1/2, and *I* 3/2 states), from the $\mathrm{Ga}(^2S_{1/2}) + \mathrm{Kr}(^1S_0)$ limit another $\Omega = 1/2$ state (the *III* 1/2 state), and from the $\mathrm{Ga}^*(^1S_0) + \mathrm{Kr}(^1S_0)$ limit an $\Omega = 0$ state (the *I* 0 state). At large internuclear distances the *I* 3/2 and *II* 1/2 potential curves are degenerate and are separated from the *I* 1/2 (ground state) curves by the group IIIA $^2P_{3/2}$–$^2P_{1/2}$ splitting. As the distance decreases the $1\,^2\Pi$ and $1\,^2\Sigma^+$ curves rapidly separate, the separation increasing approximately exponentially. As a result at short distance the atomic spin-orbit coupling is nearly quenched and the *I* 1/2 and *II* 3/2 curves approximate the $1\,^2\Pi$ curve and the *II* 1/2 curve approximates the $1\,^2\Sigma^+$ curve. The potential curves of the $2\,^2\Sigma^+$ (*III* 1/2) and $1\,^1\Sigma^+$ (*I* 0) states are unaffected by the inclusion of spin-orbit coupling.

III. DETAILS OF THE CALCULATIONS

A. Basis sets

The calculations employed an atomic (14*s*11*p*5*d*) primitive Gaussian basis set for both the gallium and krypton atoms.[5] The core orbitals (1*s*, 2*s*, 3*s*, 2*p*, 3*p*, and 3*d*) were contracted to a single function while the valence orbitals (4*s* and 4*p*) were contracted to two functions using the general contraction method of Raffenetti[6] (see also Ref. 7). These atomic basis sets were augmented with a single set of 3*d* functions, $\zeta_{Ga} = 0.16$ and $\zeta_{Kr} = 0.35$, to describe molecular polarization effects. The values of these exponents were based on calculations on other excimer laser systems.[3,8]

Since the transition of interest involves the first Rydberg state of the gallium atom, the gallium basis set was augmented with two diffuse *s* functions, $\zeta_s = 0.026$ and 0.011, and a diffuse *p* function, $\zeta_p = 0.010$, to describe this state. The exponents of these functions were obtained from Hartree–Fock (HF) calculations on the ^2S and ^2P Rydberg states of the gallium atom.

The final basis sets thus consisted of a (16*s*12*p*6*d*) primitive set contracted to [7*s*5*p*2*d*] for gallium and a (14*s*11*p*6*d*) primitive set contracted to [5*s*4*p*2*d*] for krypton.

B. Calculations on the gallium atom

The *reference orbitals* for the configuration interaction (CI) calculations on the gallium atom were obtained from a Hartree–Fock (HF) calculation of the ^2P state of the atom with the configuration

$$^2P: \ \ldots 4s^2 4p \ . \tag{1}$$

(The 4*p* set includes all three cartesian components and all are equivalent.) The Rydberg 5*s* orbital was obtained from an IVO calculation[9] using the $4s^2$ core. In terms of these orbitals the *reference configurations* for the other states of interest are

$$^2S: \ \ldots 4s^2 5s \tag{2}$$

$$^1S: \ \ldots 4s^2 \ . \tag{3}$$

Both polarization CI (POL–CI)[10] and full CI calculations have been carried out on the above states of the gallium atom. The POL–CI calculations include all single and double excitations relative to configurations (1)–(3) with the restrictions that

(1) all core orbitals (1*s*–3*s*, 2*p*–3*p*, and 3*d*) remain doubly occupied and

(2a) no more than one electron occupy the Rydberg 5*s* *and* virtual orbitals (^2P and ^1S states) or

(2b) no more than one electron occupy the Rydberg 5*s* orbital *and* no more than one electron occupy the virtual orbitals (^2S state).

The POL–CI calculations involve 22 space and 33 space/spin configurations for the ^2P state, 27 space and 43 space/spin configurations for the ^2S state, and 15 space and 15 space/spin configurations for the ^1S state.[11] Note that the POL–CI wavefunction accounts for the $4s^2 - 4p^2$ near-degeneracy effect.

The less restrictive condition, (2b), for the ^2S state is necessary to obtain a comparable description of this state. Only by relaxing condition (2a) can configurations such as

$$\ldots [4p_\pi 4p_\pi] 5s$$

be included in the POL–CI wavefunction of the ^2S state. These configurations account for the first-order difference between the 4*p* orbital obtained from the calcula-

tions on the ^2P state and the p orbital needed to describe the angular correlation of the $4s^2$ pair ($4s$–$4p$ near degeneracy) and are quite important. Comparable configurations are automatically included in the POL–CI wavefunctions of the ^2P and ^1S states.

The full CI calculations include all excitations (single–triple) relative to the configurations (1)–(3) with only restriction (1) above. The full CI calculations include 157 space and 263 space/spin configurations for the ^2P state, 204 space and 306 space/spin configurations for the ^2S state, and 42 space and 42 space/spin configurations for the ^1S state.[11] Within the frozen core approximation the accuracy of the results obtained from the full CI calculations is only limited by the completeness of the basis set.

C. Hartree–Fock calculations on GaKr

The *reference orbitals* for the POL–CI calculations[10] on the GaKr molecule were obtained from an HF calculation on the $1\,^2\Sigma^+$ state with configuration[12]

$$\cdots 13\sigma^2 14\sigma^2 15\sigma^2 16\sigma 7\pi^4 \ . \tag{1a}$$

In the POL–CI calculations the core orbitals (12 σ orbitals, 12 π orbitals, and 4 δ orbitals) are always required to be doubly occupied and so it is convenient to renumber the *valence orbitals* so that (1a) becomes

$$1\,^2\Sigma^+\!: \ 1\sigma^2 2\sigma^2 3\sigma^2 4\sigma 1\pi^4 \ . \tag{1b}$$

As $R\to\infty$ the above orbitals become

$1\sigma \to 4s_{Kr}$ $\quad 1\pi \to 4p\pi_{Kr}$

$2\sigma \to 4p\sigma_{Kr}$

$3\sigma \to 4s_{Ga}$

$4\sigma \to 4p\sigma_{Ga}$

The IVO method,[9] with a $1\sigma^2 2\sigma^2 3\sigma^2 1\pi^4$ core, was used to generate the *valence* 2π and *Rydberg* 5σ orbitals. As $R\to\infty$ these orbitals become the $4p\pi$ and $5s$ orbitals of the gallium atom. The virtual orbitals were also obtained from the IVO calculations.

In terms of the orbitals defined in this way the *reference configurations* for the other states of interest are[12]

$$1\,^2\Pi\!: \quad 1\sigma^2 2\sigma^2 3\sigma^2 1\pi^4 2\pi \tag{2}$$

$$2\,^2\Sigma^+\!: \quad 1\sigma^2 2\sigma^2 3\sigma^2 5\sigma 1\pi^4 \tag{3}$$

$$1\,^1\Sigma^+\!: \quad 1\sigma^2 2\sigma^2 3\sigma^2 1\pi^4 \ . \tag{4}$$

D. Polarization configuration interaction calculations on GaKr

The POL–CI calculations[10] on GaKr included all single and double excitations relative to the reference configurations given above with the restrictions that

(1) all core orbitals remain doubly occupied and

(2a) no more than one electron occupy the Rydberg 5σ *and* virtual orbitals ($^2\Pi$ and $^1\Sigma^+$ states) or

(2b) no more than one electron occupy the Rydberg 5σ orbital *and* no more than one electron occupy the virtual orbitals ($^2\Sigma^+$ states).

For the $^2\Sigma^+$ states the calculations considered both states simultaneously. This procedure results in 556 space and 1565 space/spin configurations for the $^2\Pi$ states, 764 space and 2314 space/spin configurations for the $^2\Sigma^+$ states, and 368 space and 558 space/spin configurations for the $^1\Sigma^+$ state.

The less restrictive condition, (2b), is necessary for the $^2\Sigma^+$ states to allow for configurations such as

$$1\sigma^2 2\sigma^2[2\pi n\pi]5\sigma 1\pi^4 \ ,$$

which are important in the description of the Rydberg $^2\Sigma^+$ state. As was the analogous case in the gallium atom, these configurations are necessary to allow for the inclusion of the $4s^2 - 4p^2$ near-degeneracy effect in the molecular wavefunction.

E. Inclusion of spin-orbit coupling in GaKr

As in our earlier calculations on excimer systems,[3,13] we have adopted a simple model[3] for including the effects of spin-orbit coupling on the calculated potential energy curves and wavefunctions. The experimental spin-orbit parameters for the open-shell atom, gallium in the present case, are used to determine the matrix elements of the spin-orbit Hamiltonian, H_{so}, coupling the molecular states at $R=\infty$. These matrix elements are then assumed to be independent of R and are added to the diagonal matrix of the electronic energies

$$H^{\Omega}(R) = \delta_{ij} E_i(R) + H_{so} \ . \tag{5}$$

The energies and wavefunctions with spin-orbit corrections are obtained by diagonalizing $H^{\Omega}(R)$. This procedure is reasonable only so long as (1) the molecular states retain the identity of the atomic states from which they arise and (2) the atomic contributions to the molecular spin-orbit interactions are dominant.

For GaKr the spin-orbit interaction affects only those states which arise from the $Ga(^2P) + Kr(^1S)$ limit. The Hamiltonian matrix for the $\Omega = 1/2$ states arising from this limit is

$$H^{\Omega=1/2} = \begin{bmatrix} E(1\,^2\Sigma^+) & \sqrt{2}\,\lambda_{Ga} \\ \sqrt{2}\lambda_{Ga} & E(1\,^2\Pi) - \lambda_{Ga} \end{bmatrix} \tag{6a}$$

and for the $\Omega = 3/2$ state is

$$H^{\Omega=3/2} = E(1\,^2\Pi) + \lambda_{Ga} \ . \tag{6b}$$

In (6) λ_{Ga} is one third of the $^2P_{3/2}$–$^2P_{1/2}$ splitting in the gallium atom. The energies of the $2\,^2\Sigma^+$ and $1\,^1\Sigma^+$ states of GaKr are unaffected by the inclusion of spin-orbit coupling. The wavefunctions for the states obtained by diagonalizing (6) may be written in the form

$$|I\,1/2\rangle = \cos\theta\,|1\,^2\Pi_+\beta\rangle + \sin\theta\,|1\,^2\Sigma^+\alpha\rangle \tag{7a}$$

$$|II\,1/2\rangle = -\sin\theta\,|1\,^2\Pi_+\beta\rangle + \cos\theta\,|1\,^2\Sigma^+\alpha\rangle \tag{7b}$$

$$|I\,3/2\rangle = |1\,^2\Pi_+\alpha\rangle \ , \tag{7c}$$

where θ is the spin rotation angle. The wavefunctions for the $|III\,1/2\rangle$ and $|I0\rangle$ states are

$$|III\,1/2\rangle = |2\,^2\Sigma^+\alpha\rangle \tag{7d}$$

$$|I0\rangle = |1\,^1\Sigma^+\rangle \ . \tag{7e}$$

TABLE I. Calculations on the ^2P, ^2S, and ^1S (ion) states of Ga and the ^1S state of krypton. For Ga the experimental results have been corrected for spin-orbit effects (see the text). Units are as indicated.

Atom state	^2P	Ga(Ga$^+$) ^2S	^1S	Kr ^1S
Total energies (hartree)[a]				
POL–CI	0.22523	0.12130	0.01912	0.97377[b]
Full CI	0.23484	0.12750	0.02305	···
Excitation energies (eV)				
POL–CI	0.00	2.83	5.61	···
Full CI	0.00	2.92	5.76	···
Expt'l[c]	0.000	3.005	5.930	···
Transition moment (ea_0)				
POL–CI	···	−1.298[d]	···	···
Lifetimes (nsec)				
POL–CI	···	8.2[e]	···	···
Expt'l	···	6.8[f]	···	···

[a]For Ga the energies are relative to −1923 hartree; for Kr the energies are relative to −2751 hartree.
[b]For Kr the POL–CI wavefunction is equivalent to the HF wavefunction.
[c]Reference 4.
[d]This is the matrix element $1/3[\langle {}^2S|x|{}^2P_x\rangle + \langle {}^2S|y|{}^2P_y\rangle + \langle {}^2S|z|{}^2P_z\rangle]$.
[e]Using the experimental excitation energy we obtain 6.9 nsec.
[f]Reference 14.

With the definitions (7) the dipole transition moments coupling the $III\,1/2$ state with all of the lower states are

$$\mu_z(III\,1/2 - I\,1/2) = \sin\theta\,\langle 2\,{}^2\Sigma^+|z|1\,{}^2\Sigma^+\rangle \quad (8a)$$

$$\mu_x(III\,1/2 - I\,1/2) = \cos\theta\,\langle 2\,{}^2\Sigma^+|x|1\,{}^2\Pi_x\rangle/\sqrt{2} \quad (8b)$$

$$\mu_z(III\,1/2 - II\,1/2) = -\cos\theta\,\langle 2\,{}^2\Sigma^+|z|1\,{}^2\Sigma^+\rangle \quad (9a)$$

$$\mu_x(III\,1/2 - II\,1/2) = \sin\theta\,\langle 2\,{}^2\Sigma^+|x|1\,{}^2\Pi_x\rangle/\sqrt{2} \quad (9b)$$

$$\mu_x(III\,1/2 - I\,3/2) = -\langle 2\,{}^2\Sigma^+|x|1\,{}^2\Pi_x\rangle/\sqrt{2}\ . \quad (10)$$

Since the $\langle 2\,{}^2\Sigma^+|x|1\,{}^2\Pi_x\rangle$ transition moment is expected to be comparable in magnitude to the $\langle 2\,{}^2\Sigma^+|z|1\,{}^2\Sigma^+\rangle$ moment (at $R=\infty$ they are identical), it is clear from (8) and (9) that the transitions from the $III\,1/2$ state of the $I\,1/2$ and $II\,1/2$ states can have both large parallel and perpendicular components.

IV. RESULTS FOR Ga, GaKr, AND GaKr$^+$

A. Electronic states of Ga, without spin-orbit corrections

The results of the calculations on the Ga atom are summarized in Table I. The computed ^2S–^2P excitation energy is 2.83 eV (POL–CI) and 2.92 eV (full CI). Averaging the multiplet energies for the $^2P_{1/2}$ and $^2P_{3/2}$ states of gallium from Moore,[4] the experimental ^2S–^2P splitting is calculated to be 3.005 eV, just 0.08–0.17 eV larger than the computed spacing. The errors in the calculated ionization potentials, 5.61 eV (POL–CI) and 5.76 eV (full CI), are somewhat larger, being 0.32 and 0.17 eV, respectively.

The lifetime of the ^2S state of gallium has been determined by Norton and Gallagher[14] to be 6.8 ± 0.3 nsec. For the model used here the lifetime of the ^2S state is independent of the spin-orbit corrections. From the POL–CI wavefunctions we calculate a lifetime of 8.2 nsec for the ^2S state (6.9 nsec, if the experimental excitation energy is used instead of the calculated excitation energy).

B. Electronic states of GaKr and GaKr$^+$, without spin-orbit corrections

The energies of the $1\,{}^2\Pi$, $1,2\,{}^2\Sigma^+$, and $1\,{}^1\Sigma^+$ states of GaKr and GaKr$^+$ obtained from the POL–CI calculations are listed in Table II and the resulting potential energy curves are plotted in Fig. 2. As is usual in such calculations,[3,8,13] the energies of the $1\,{}^2\Pi$ and $1\,{}^2\Sigma^+$ states, both of which arise from the ^2P limit, are not exactly equal at $R = 15.0\,a_0$ (the largest value of R considered). The difference, 0.00114 hartree (0.031 eV), is attributable to the inequivalence of the $4p\sigma$ and $4p\pi$ orbitals and to core polarization effects (the core orbitals were obtained from HF calculations on the $1\,{}^2\Sigma^+$ state which does not have the full rotational symmetry of the atom). In the plots the asymptotic energies of the states have been adjusted to give the experimental atomic energy splittings.

In line with the discussion in Sec. II, the potential energy curve for the $1\,{}^2\Pi$ state is found to be less repulsive than that of the $1\,{}^2\Sigma^+$ state, thus making the $1\,{}^2\Pi$ state the ground state of the system. In fact, we find that the $1\,{}^2\Pi$ curve is slightly bound, $D_e \sim 0.04$ eV (see Table III). Although spurious minima have been found in previous calculations on excimer systems[8,13] and attributed to basis set limitations, the well in the $1\,{}^2\Pi$ curve is substantially larger than has been observed heretofore. We thus suspect that the minimum in the $1\,{}^2\Pi$ curve is not just a result of calculational limitations. The depth of the well in the real $1\,{}^2\Pi$ curve is, of course, expected to be significantly larger than that calculated here since the POL–CI method is not designed to account for the attractive van der Waals' interaction.

As predicted in Sec. II both the $2\,{}^2\Sigma^+$ and $1\,{}^1\Sigma^+$ curves

TABLE II. Energies obtained from the POL–CI calculations on the low-lying electronic states of GaKr and GaKr$^+$. Distances are in bohr; energies are in hartree. Energies are relative to −4674 hartree.

R	GaKr $1\,{}^2\Pi$	GaKr $1\,{}^2\Sigma^+$	GaKr $2\,{}^2\Sigma^+$	GaKr$^+$ $1\,{}^1\Sigma^+$
3.75	−1.02539	−0.96866	−0.93379	−0.85625
4.00	−1.07933	−1.01870	−0.99238	−0.90683
4.50	−1.14810	−1.08688	−1.06440	−0.96758
5.00	−1.17924	−1.13516	−1.08855	−0.99287
5.50	−1.19225	−1.16324	−1.09545	−1.00070
6.00	−1.19772	−1.17938	−1.09677	−1.00202
6.50	−1.19984	−1.18865	−1.09631	−1.00107
7.00	−1.20045	−1.19396	−1.09545	−0.99950
8.00	−1.20020	−1.19865	−1.09420	−0.99670
10.00	−1.19933	−1.20029	−1.09406	−0.99402
15.00	−1.19878	−1.20012	−1.09502	−0.99304

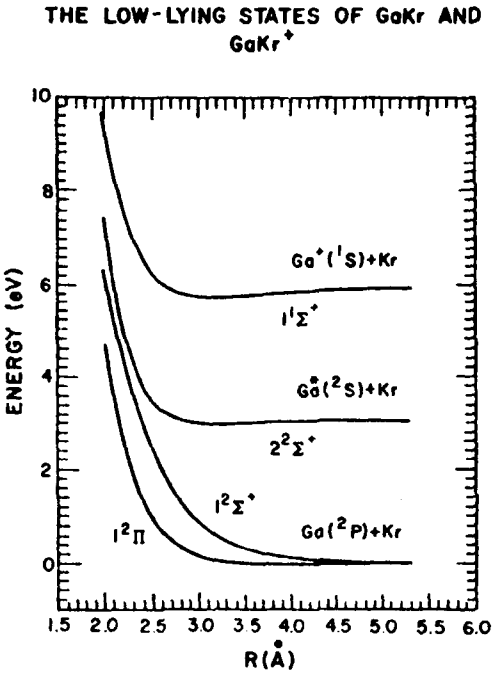


FIG. 2. Calculated potential energy curves for the states of GaKr and GaKr$^+$ arising from the $Ga(^2P, ^2S) + Kr(^1S)$ and $Ga^+(^1S) + Kr(^1S)$ separated atom limits. The curves have been uniformly shifted to correct for the errors in the gallium atom excitation energies.

are found to be bound with calculated dissociation energies of 0.047 and 0.24 eV, respectively. Again, inclusion of the attractive van der Waals' interaction would be expected to significantly increase the calculated well depths. There is a slight hump in the calculated potential curve of the $2\,^2\Sigma^+$ state, ~0.026 eV, since the long-range interaction of the excited Ga and Kr atoms is repulsive. This hump could, however, disappear when the van der Waals' attraction is added to the calculated curve.

The dipole transition moments radiatively coupling the states of GaKr are given in Table IV and plotted in Fig. 3. At $R = \infty$ the transition moment coupling the $1\,^2\Pi$ and $1\,^2\Sigma^+$ states is identically zero and those coupling the ($1\,^2\Pi$, $1\,^2\Sigma^+$) states with the $2\,^2\Sigma^+$ state are approximately equal to the ^2S–^2P atomic transition moment (after properly accounting for differences in degeneracy factors). As R decreases rather minor changes occur until $R \sim 2.5$ Å. For $R < 2.5$ Å substantial changes are noted in all of the transition moments, although the change in the $2\,^2\Sigma^+$–$1\,^2\Pi$ moment is less dramatic than for the $2\,^2\Sigma^+$–$1\,^2\Sigma^+$ and $1\,^2\Sigma^+$–$1\,^2\Pi$ moments. The erratic behavior of the transition moments for $R < 2.5$ Å is one manifestation of the strong interaction of the $1\,^2\Sigma^+$ and $2\,^2\Sigma^+$ states at short R.

Although, as noted above, substantial changes are found in the transition moments for $R < 2.5$ Å, such behavior can be expected to have little effect on the observable properties of the system. At $R = 2.5$ Å the energy of the $2\,^2\Sigma^+$ state is >0.5 eV above its asymptote so that the region $R < 2.5$ Å would be thermally inaccessible.

TABLE III. Spectroscopic constants for the bound states of $^{69}Ga\,^{84}Kr$ and $^{69}Ga\,^{84}Kr^+$. Units are as indicated.

	GaKr		GaKr$^+$
	$1\,^2\Pi$	$2\,^2\Sigma^+$	$1\,^2\Sigma^+$
T_e, eV	0.00	2.82	5.40
R_e, Å	3.78	3.17	3.14
D_e, eV	0.041	0.047	0.24
ω'_e, cm^{-1}	36	66	83
B_e	0.0312	0.0442	0.0452

TABLE IV. Dipole transition moments coupling the low-lying states of GaKr obtained from the POL–CI calculations.

R	$1\,^2\Sigma^+$–$1\,^2\Pi^a$	$2\,^2\Sigma^+$–$1\,^2\Pi^a$	$2\,^2\Sigma^+$–$1\,^2\Sigma^+$
3.75	−0.9037	−0.7877	0.4095
4.00	−0.8462	−0.8680	−0.3055
4.50	−0.1365	−1.2090	−2.0676
5.00	0.0992	−1.2349	−1.4402
5.50	0.1257	−1.2469	−1.2456
6.00	0.1166	−1.2554	−1.1893
6.50	0.0984	−1.2619	−1.1800
7.00	0.0789	−1.2677	−1.1902
8.00	0.0476	−1.2777	−1.2310
10.00	0.0163	−1.2898	−1.2976
15.00	0.0018	−1.2917	−1.3101

[a]The matrix element given is $\langle \pi\,^2\Sigma^+ | x | 1\,^2\Pi_x \rangle$.

DIPOLE TRANSITION MOMENTS AMONG THE LOW-LYING STATES OF GaKr

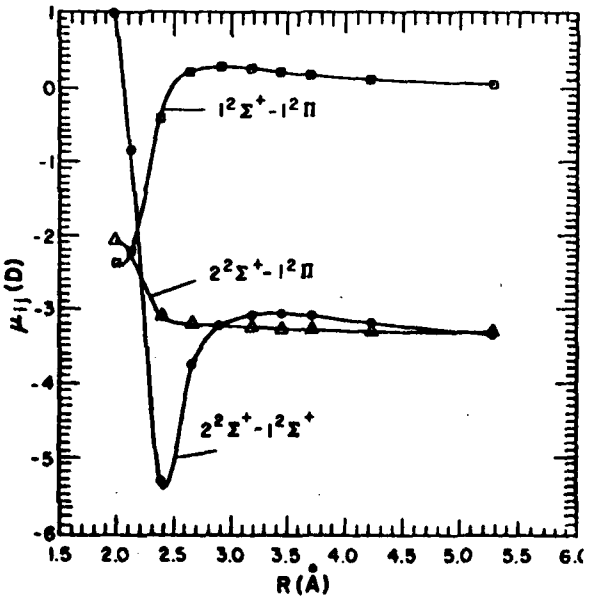


FIG. 3. Calculated dipole transition moments for the $1\,^2\Sigma^+$–$1\,^2\Pi$, $2\,^2\Sigma^+$–$1\,^2\Pi$, and $2\,^2\Sigma^+$–$1\,^2\Sigma^+$ transitions in GaKr.

THE LOW-LYING STATES OF GaKr AND GaKr$^+$ WITH SPIN-ORBIT CORRECTIONS

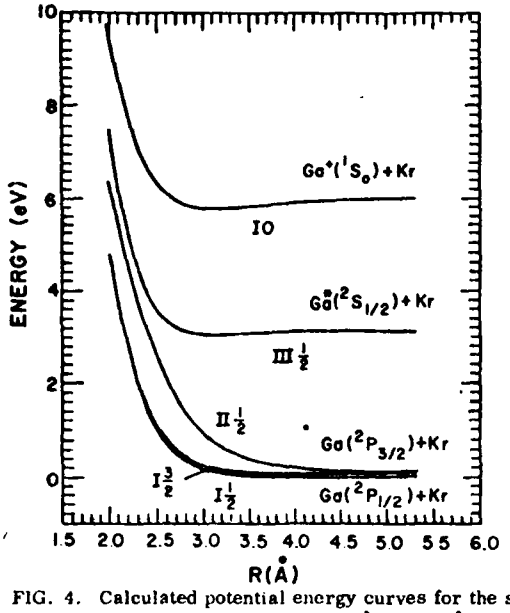


FIG. 4. Calculated potential energy curves for the states of GaKr and GaKr$^+$ arising from the Ga($^2P_{1/2,3/2}$, $^2S_{1/2}$)+Kr($^1S_0$) and Ga$^+$($^1S_0$)+Kr($^1S_0$) separated atom limits. The curves have been uniformly shifted to correct for the errors in the gallium excitation energies.

C. Electronic states of GaKr and GaKr$^+$, with spin-orbit corrections

The energies of the states of GaKr and GaKr$^+$ with spin-orbit corrections are given in Table V and the resulting potential energy curves are plotted in Fig. 4. In calculating the spin-orbit corrected energies we shifted the $1\,^2\Pi$ energies to agree with the $1\,^2\Sigma^+$ energy at the largest value of R considered ($R = 15.0\,a_0$). Again, in plotting the potential energy curves, the asymptotic energies of the states have been adjusted to give the correct atomic energy splittings.

As predicted in Sec. II, at short R the curves for the

TABLE V. Calculated energies of the I 1/2, II 1/2, and I 3/2 states of GaKr with spin-orbit corrections. Distances are in bohr; energies are in hartree. Energies are relative to the energy of the $1\,^2\Sigma^+$ state at $R = 15\,a_0$.

R	I 1/2	I 3/2	II 1/2
3.75	0.17228	0.17484	0.23152
4.00	0.11835	0.12091	0.18147
4.50	0.04957	0.05213	0.11329
5.00	0.01841	0.02099	0.06502
5.50	0.00538	0.00799	0.03698
6.00	−0.00015	0.00252	0.02089
6.50	−0.00234	0.00040	0.01170
7.00	−0.00307	−0.00022	0.00650
8.00	−0.00316	0.00003	0.00215
10.00	−0.00280	0.00090	0.00103
15.00	−0.00251	0.00126	0.00126

TABLE VI. Calculated dipole transition moments coupling the III 1/2 and I 1/2, II 1/2 and I 3/2 states of GaKr with spin-orbit corrections. Distances are in bohr; moments are in atomic units.

	III 1/2–I 1/2		III 1/2–II 1/2		III 1/2–I 3/2
R	z	(x,y)	z	(x,y)	(x,y)
4.00	−0.0086	0.6135	−0.3054	−0.0173	−0.6138
4.50	−0.0576	0.8545	−2.0667	−0.0238	−0.8549
5.00	−0.0549	0.8726	−1.4392	−0.0333	−0.8732
5.50	−0.0701	0.8803	−1.2437	−0.0496	−0.8817
6.00	−0.1007	0.8845	−1.1850	−0.0752	−0.8877
6.50	−0.1504	0.8850	−1.1704	−0.1138	−0.8923
7.00	−0.2249	0.8803	−1.1688	−0.1694	−0.8964
8.00	−0.4405	0.8436	−1.1495	−0.3233	−0.9035
10.00	−0.7254	0.7562	−1.0759	−0.5098	−0.9120
15.00	−0.7564	0.7457	−1.0697	−0.5273	−0.9134

$I$1/2 and $I$3/2 states become nearly degenerate and just represent the curves for the two spin-orbit components of the $1\,^2\Pi$ state. Also, at short R the $II$1/2 curve closely approximates the curve for the $1\,^2\Sigma^+$ state. The potential energy curves for the $2\,^2\Sigma^+$ ($III$1/2) and $1\,^1\Sigma^+$(I0) states are unchanged by the inclusion of spin-orbit corrections.

The dipole transition moments radiatively coupling the states of GaKr with spin-orbit corrections are given in Table VI and are plotted in Fig. 5. It should be noted that both the z component of the $III$1/2–$I$1/2 transition and the x component of the $III$1/2–$II$1/2 transition are now found to vary significantly even for $R > 2.5$ Å. This is due to the changing nature of the $I$1/2 and $II$1/2 states

DIPOLE TRANSITION MOMENTS AMONG THE LOW-LYING STATES OF GaKr WITH SPIN-ORBIT CORRECTIONS

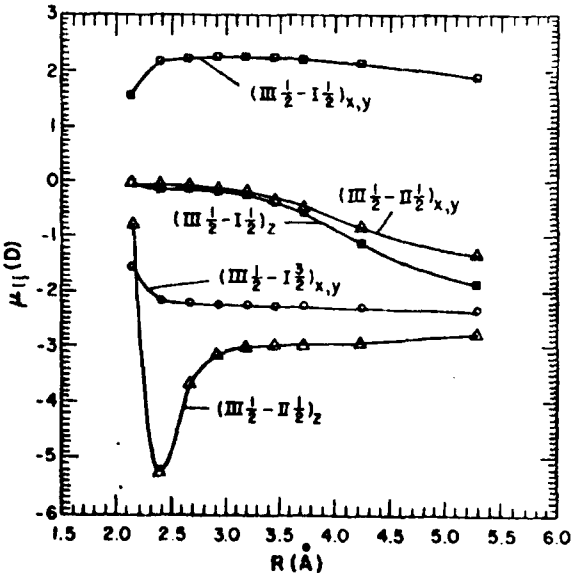


FIG. 5. Calculated dipole transition moments for the III 1/2–$I$1/2, III 1/2–$I$3/2, and III 1/2–II 1/2 transitions in GaKr.

TABLE VII. Excitation energies and ionization potentials for the Ga, In, and Tl atoms, in eV. Taken from Ref. 4.

	Group IIIA atoms		
State	Ga	In	Tl
$^2P_{1/2}$	0.000	0.000	0.000
$^2P_{3/2}$	0.102	0.274	0.966
$^2S_{1/2}$	3.073	3.022	3.282
$^1S_0$	5.998	5.786	6.108

as the atomic spin-orbit coupling is quenched by molecular formation.

V. EXTRAPOLATION TO InKr AND TlKr

To provide information on the InKr and TlKr molecules, the latter being the most experimentally accessible of the Group IIIA-krypton molecules, the potential curves for InKr and TlKr have been estimated from the GaKr curves. The excitation energies and ionization potentials for the series Ga, In, and Tl are given in Table VII. As can be seen, this series does not form a steady progression: In has a lower ionization potential and excitation energies than Ga, as expected, but Tl has a higher ionization potential and excitation energies. This anomalous behavior in Tl is due in part to the filling of the 4*f* shell ("lanthanide contraction") and to the larger spin-orbit interactions.[15] These effects are only partially accounted for in the present models.

To simulate InKr and TlKr, the experimental spin-orbit parameters for In and Tl, see Table VII, were used to couple the curves calculated for GaKr. The potential energy curves for InKr and TlKr obtained in this way are expected to be qualitatively correct; the curves are plotted in Figs. 6 and 7. As before, the plotted

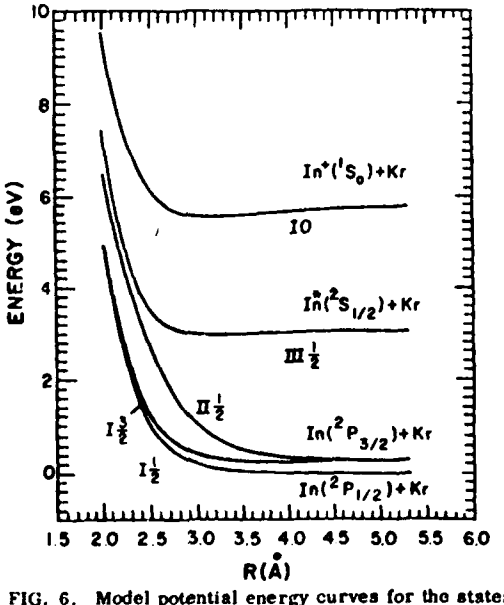


FIG. 6. Model potential energy curves for the states of InKr and InKr+ arising from the $In(^2P_{1/2,3/2}, ^2S_{1/2}) + Kr(^1S_0)$ and $In^+(^1S_0) + Kr(^1S_0)$ separated atom limits, see the text.

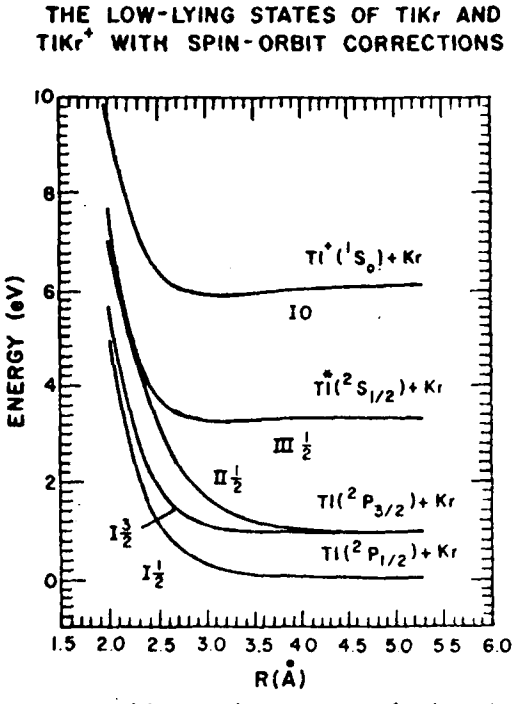


FIG. 7. Model potential energy curves for the states of TlKr and TlKr+ arising from the $Tl(^2P_{1/2,3/2}, ^2S_{1/2}) + Kr(^1S_0)$ and $In^+(^1S_0) + Kr(^1S_0)$ separated atom limits, see the text.

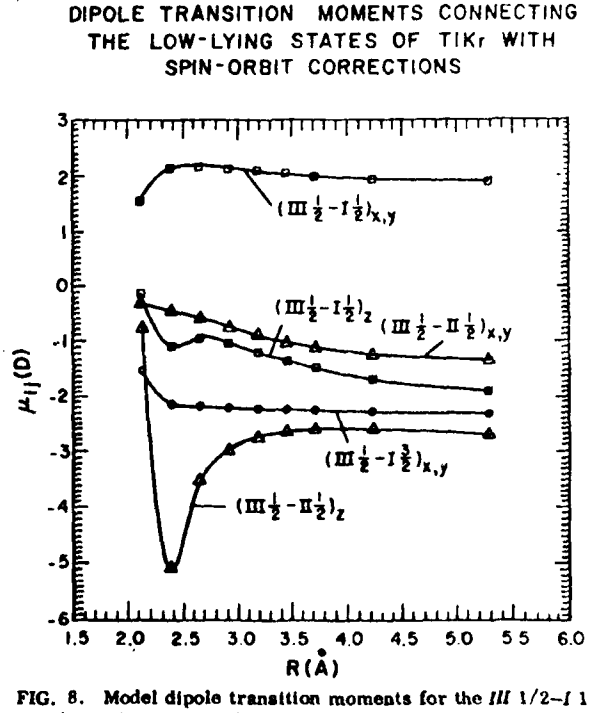


FIG. 8. Model dipole transition moments for the *III* 1/2–*I* 1/2, *III* 1/2–*I* 3/2, and *III* 1/2–*II* 1/2 transitions in TlKr.

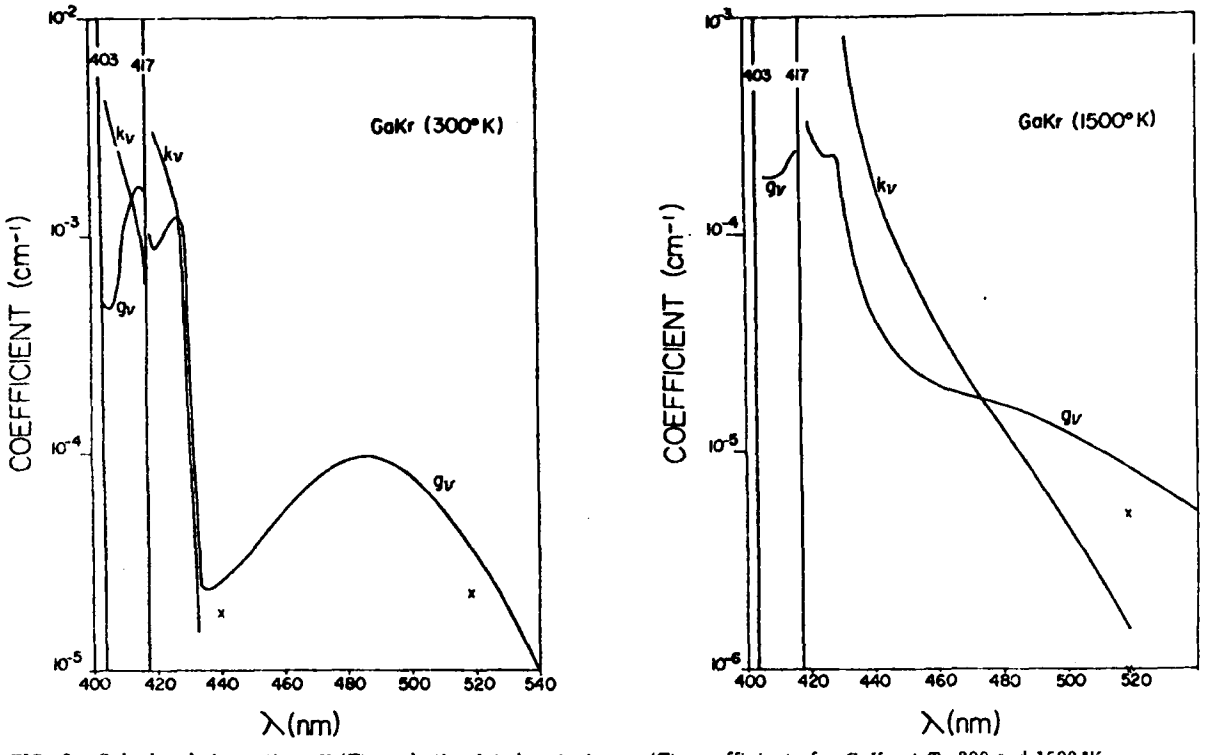


FIG. 9. Calculated absorption, $K_\nu(T)$, and stimulated emission, $g_\nu(T)$, coefficients for GaKr at $T = 300$ and 1500 °K.

curves have been shifted to give the correct atomic excitation energies. The effect of the increasing spin-orbit interaction in the sequence Ga < In < Tl is evident in Figs. 4, 6, and 7.

Using the transition moments obtained from the calculations on GaKr and the wavefunctions obtained from the TlKr simulation, the transition moments of TlKr have been estimated. The moments so obtained are plotted in Fig. 8.

VI. ABSORPTION AND STIMULATED EMISSION COEFFICIENTS FOR GaKr, InKr, AND TlKr

The interest in the Group IIIA-rare gas systems arises from the possibility of their use as visible laser systems. In order to judge their usefulness as lasers it is convenient to calculate the pure absorption, $K_\nu(T)$, and stimulated emission, $g_\nu(T)$, coefficients for the perturbed atomic transitions. Obtaining quantum mechanical results for these quantities would require a complex calculation which is not justified by the extrapolations used to obtain the InKr and TlKr curves. Consequently, we have used instead the method of Gallagher and coworkers,[16] which is based on the classical Franck–Condon principle. In this approximation

$$g_\nu(T)_J = [M^*][X](\lambda^2/8\pi)A_0(J)(\lambda/\lambda_0)I'_J(\nu T)$$

$$K_\nu(T) = [M][X](g^*/g)g_\nu(T), \exp[h(\nu-\nu_0)/kT]$$

and

$$I'_J(\nu, T) = 4\pi R(\nu)^2\left(\frac{\nu}{\nu_0}\right)^4 \frac{1}{d\nu/dR(\nu)} \frac{D[R(\nu)]}{D(\infty)} \frac{g_m}{g_A} \times \exp\{-V^*[R(\nu)]/kT\} .$$

In these equations, J refers to the bands associated with the $^2S_{1/2}$–^2P_J transition, ν_0 and λ_0 are the frequency and wavelength of the atomic transition with a transition rate of $A_0(J)$, $g^* = 2$ for the $^2S_{1/2}$ state and $g = 2J+1$ for the ^2P_J states, $D(R)$ is the transition dipole moment at R, g_m and g_A are the statistical weights of the excited molecular and parent atomic state, and $V^*(R)$ is the excited state potential curve relative to the energy of the excited atomic state. [M], [M*], and [X] are the concentrations of ground and excited metal atoms and of rare gas atoms at the temperature T.

In order to obtain $g_\nu(T)$ and $K_\nu(T)$, the calculated curves were first fit with cubic splines. The spline fit was then used to calculate $d\nu/dR$, and these quantities, along with the atomic transition rates,[14] were used to calculate the absorption and stimulated emission coefficients for pressure and excitation conditions relevant to the experimental studies. We have calculated the coefficients for two different types of conditions. The high temperature results correspond to the case where the concentration of the metal is obtained from the vapor pressure of the metal itself, while the low temperature results correspond to obtaining the required concentration of the metal from vaporization of MI_3. This latter condition has been suggested by Gallagher[2] as a possible means of obtaining high concentrations of the metal at low temperatures. In both cases the densities used are

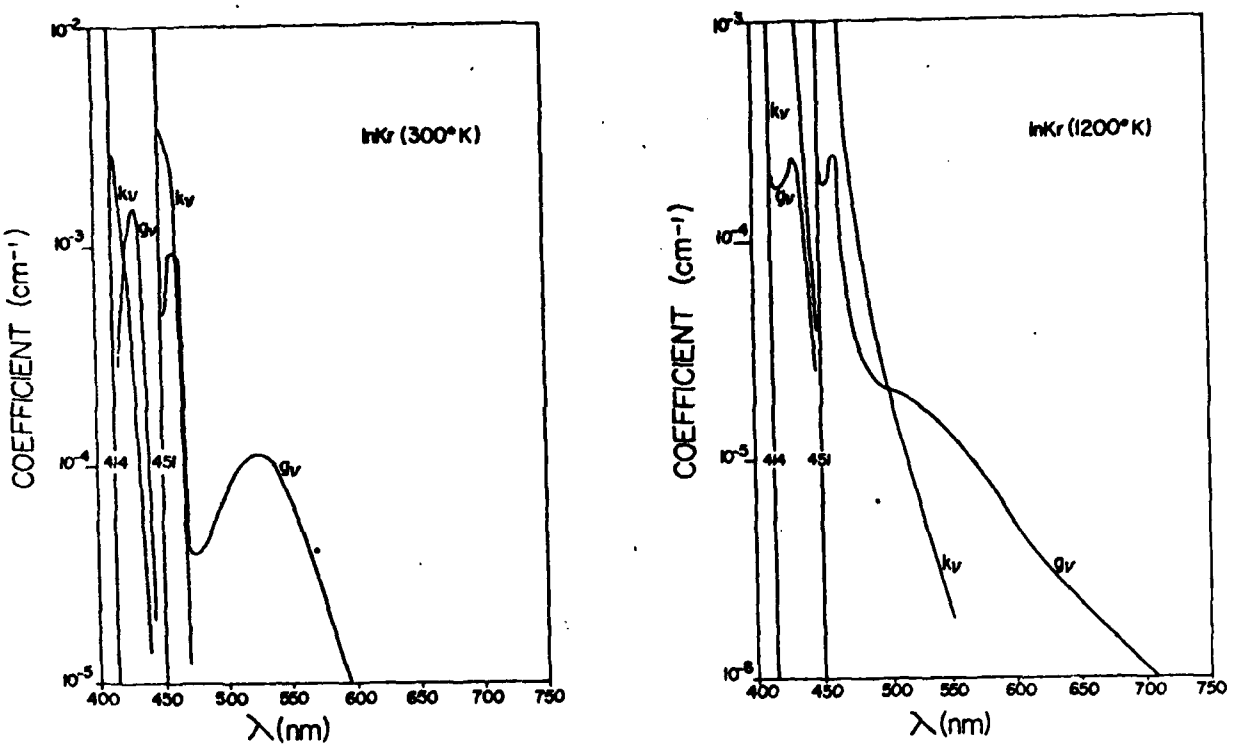


FIG. 10. Model absorption, $K_\nu(T)$, and stimulated emission, $g_\nu(T)$, coefficients for InKr at $T = 300$ and 1200 °K.

$10^{20}/\text{cm}^3$ for [Kr], $10^{16}/\text{cm}^3 = 3[\text{M}(^2P_{1/2})] = 1.5[\text{M}(^2P_{3/2})]$, and $2\times10^{14}/\text{cm}^3$ for $[\text{M}(^2S_{1/2})]$. Since the blue wings of the $^2S_{1/2}-{}^2P_{1/2,3/2}$ bands are due to transitions occurring at large R, where our curves are not expected to be especially accurate, we have not calculated g_ν and K_ν in these regions.

The resulting absorption and stimulated emission coefficients for GaKr, InKr, and TlKr are given in Figs.

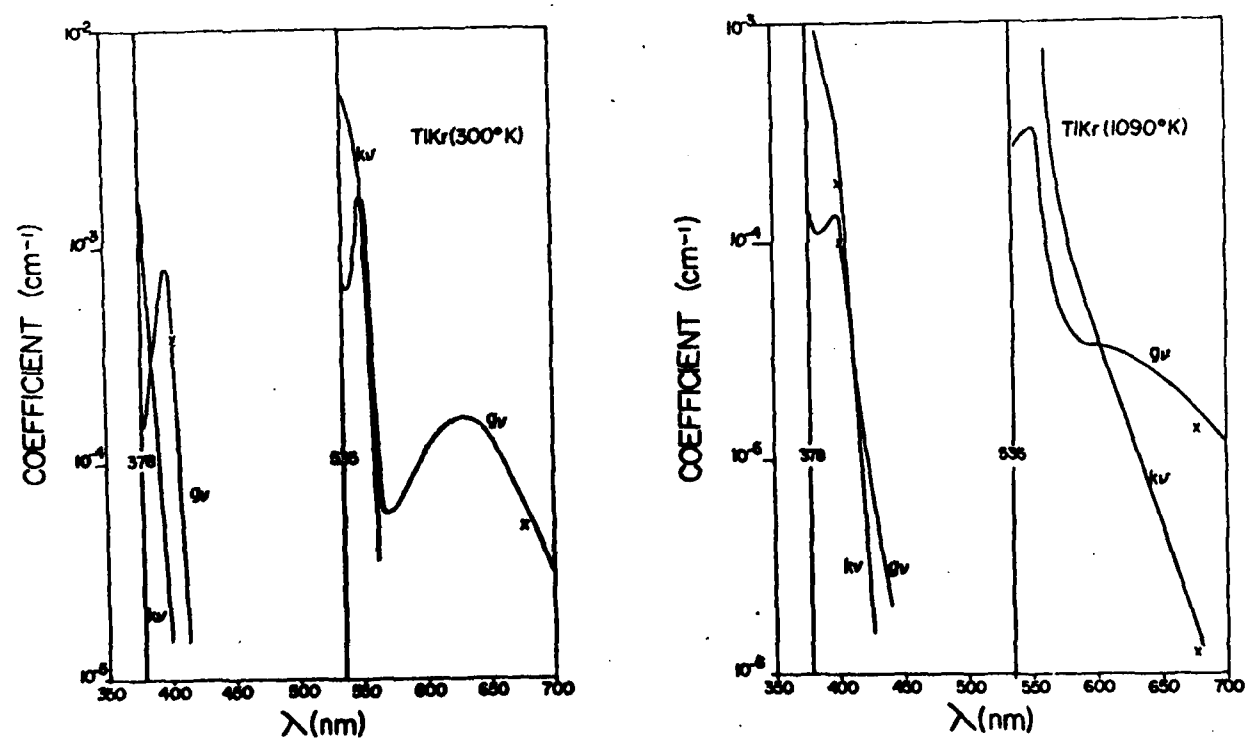


FIG. 11. Model absorption, $K_\nu(T)$, and stimulated emission, $g_\nu(T)$, coefficients for TlKr at $T = 300$ and 1090 °K.

9–11. In these figures the solid lines are the total coefficients calculated by assuming that the transition moment, $D(R)$, is a constant. The X's are the coefficients obtained by using the spatially averaged values of $D(R)$ calculated from Table VI and the appropriate spin-orbit parameters. For GaKr at 1500 °K (Fig. 10) and for $403.4 < \lambda < 417.3$ nm, $K_\nu > 10^{-3}$ cm^{-1}. The dip in g_ν close to the line center is due to the positive value of $V^*(R \sim 8a_0) - V^*(R = \infty)$.

The gain coefficient, which is approximately equal to $g_\nu - K_\nu$ can be estimated from these curves. For the frequencies at which gain occurs, the use of $D(R)$ decreases g_ν, K_ν, and the gain by 30–40% for GaKr and by 9–15% for TlKr. For TlKr the errors introduced by the extrapolation procedure are likely to be larger than those produced by using $D(\infty)$ rather than $D(R)$. The maximum gain occurs approximately at the minimum in the $III\,1/2$ curve. Because g_ν depends exponentially on V^*, changes in the well depth of the excited state would have a significant effect on g_ν. If the well depth were larger for TlKr, as Gallagher and co-workers predict,[1] g_ν and the gain would be larger. Likewise, since the excited state in TlXe is predicted[1] to be more bound than in TlKr, the gain should be larger for TlXe than for TlKr.

[1]B. Cheron, R. Scheps, and A. Gallagher, J. Chem. Phys. 65, 326 (1976).
[2]A. Gallagher, private communication.
[3]P. J. Hay, T. H. Dunning, Jr. and R. C. Raffenetti, J. Chem. Phys. 65, 2679 (1976).
[4]C. E. Moore, *Atomic Energy Levels*, National Bureau of Standards, Circular No. 467, Vols. I–III. (U.S. GPO, Washington, D. C., 1949).
[5]T. H. Dunning, Jr., J. Chem. Phys. 66, 1382 (1977).
[6]R. C. Raffenetti, J. Chem. Phys. 58, 4452 (1973).
[7]T. H. Dunning, Jr. and P. J. Hay in *Methods of Electronic Structure Theory*, edited by H. F. Schaefer, III (Plenum, New York, 1977), Chap. 1.
[8]T. H. Dunning, Jr. and P. J. Hay, J. Chem. Phys. 66, 3767 (1977).
[9]W. J. Hunt and W. A. Goddard, III, Chem. Phys. Lett. 3, 414 (1969).
[10]P. J. Hay and T. H. Dunning, Jr., J. Chem. Phys. 64, 5077 (1976); T. H. Dunning, Jr., J. Chem. Phys. 65, 3854 (1976). The POL–CI wavefunction is closely related to the first-order wavefunction of Schaefer, Harris, and co-workers, see, e.g., H. F. Schaefer, III, R. A. Klemm, and F. E. Harris, J. Chem. Phys. 51, 4643 (1969).
[11]The CI program cannot make full use of the atomic or diatomic symmetry. The numbers of configurations quoted in the text are for C_{2v} symmetry.
[12]For the states of interest here the generalized valence bond wavefunctions would also include the s–p near-degeneracy configurations. As noted in the text these configurations are included in the POL–CI wavefunctions.
[13]P. J. Hay and T. H. Dunning, Jr., J. Chem. Phys. 66, 1306 (1977).
[14]M. Norton and A. Gallagher, Phys. Rev. A 3, 915 (1971).
[15]P. Bagus, Y. S. Lee, and K. S. Pitzer, Chem. Phys. Lett. 33, 408 (1975).
[16]R. Scheps, Ch. Ottinger, G. York, and A. Gallagher, J. Chem. Phys. 63, 2581 (1975).

Appendix B

```
;  <WATTS>QQQQ&34;10 *  WED 15-NOV-78 8:55AM

operator sinh,cosh,re,ef,ra,ab,rg,dd;
array ee(3),ff(3),bb(3);
for all x let df(sinh(x),x) = cosh(x);
for all x let df(cosh(x),x) = sinh(x);
let ee(1)=ex,ee(2)=ey,ee(3)=ez,ff(1)=fx,ff(2)=fy,ff(3)=fz,bb(1)=bx,bb(2)=by,BB(3)=BZ
let df(re(ee,ff),ex) =  ef(ex,fx)/re(ee,ff);
let df(re(ee,ff),ey) =  ef(ey,fy)/re(ee,ff);
let df(re(ee,ff),ez) =  ef(ez,fz)/re(ee,ff);
let df(re(ee,ff),fx) = -ef(ex,fx)/re(ee,ff);
let df(re(ee,ff),fy) = -ef(ey,fy)/re(ee,ff);
let df(re(ee,ff),fz) = -ef(ez,fz)/re(ee,ff);
let df(re(ee,ff),bx) = 0;
let df(re(ee,ff),by) = 0;
let df(re(ee,ff),bz) = 0;
let df(ef(ex,fx),ex) =  1;
let df(ef(ex,fx),fx) = -1;
let df(ef(ey,fy),ey) =  1;
let df(ef(ey,fy),fy) = -1;
let df(ef(ez,fz),ez) =  1;
let df(ef(ez,fz),fz) = -1;
for all x,y let df(ef(x,y),bx) = 0;
for all x,y let df(ef(x,y),by) = 0;
for all x,y let df(ef(x,y),bz) = 0;
let df(ra(ee,ff,bb),ex) =  (a/c)*ab(ex,fx,bx)/ra(ee,ff,bb);
let df(ra(ee,ff,bb),ey) =  (a/c)*ab(ey,fy,by)/ra(ee,ff,bb);
let df(ra(ee,ff,bb),ez) =  (a/c)*ab(ez,fz,bz)/ra(ee,ff,bb);
let df(ra(ee,ff,bb),fx) =  (b/c)*ab(ex,fx,bx)/ra(ee,ff,bb);
let df(ra(ee,ff,bb),fy) =  (b/c)*ab(ey,fy,by)/ra(ee,ff,bb);
let df(ra(ee,ff,bb),fz) =  (b/c)*ab(ez,fz,bz)/ra(ee,ff,bb);
let df(ra(ee,ff,bb),bx) = -ab(ex,fx,bx)/ra(ee,ff,bb);
let df(ra(ee,ff,bb),by) = -ab(ey,fy,by)/ra(ee,ff,bb);
let df(ra(ee,ff,bb),bz) = -ab(ez,fz,bz)/ra(ee,ff,bb);
let df(ab(ex,fx,bx),ex) = a/c;
let df(ab(ey,fy,by),ey) = a/c;
let df(ab(ez,fz,bz),ez) = a/c;
let df(ab(ex,fx,bx),fx) = b/c;
let df(ab(ey,fy,by),fy) = b/c;
let df(ab(ez,fz,bz),fz) = b/c;
let df(ab(ex,fx,bx),bx) = -1;
let df(ab(ey,fy,by),by) = -1;
let df(ab(ez,fz,bz),bz) = -1;
let df(ab(ex,fx,bx),ey) = 0;
let df(ab(ex,fx,bx),ez) = 0;
let df(ab(ex,fx,bx),fy) = 0;
let df(ab(ex,fx,bx),fz) = 0;
let df(ab(ex,fx,bx),by) = 0;
let df(ab(ex,fx,bx),bz) = 0;
let df(ab(ey,fy,by),ex) = 0;
let df(ab(ey,fy,by),ez) = 0;
let df(ab(ey,fy,by),fx) = 0;
let df(ab(ey,fy,by),fz) = 0;
```

```
; <WATTS>GQQD.Q8;10   WED 15-NOV-78 8:55AM

let df(ab(ey,fy,by),ax) = 0;
let df(ab(ey,fy,by),bz) = 0;
let df(ab(ez,fz,bz),ex) = 0;
let df(ab(ez,fz,bz),ey) = 0;
let df(ab(ez,fz,bz),fx) = 0;
let df(ab(ez,fz,bz),fy) = 0;
let df(ab(ez,fz,bz),ax) = 0;
let df(ab(ez,fz,bz),by) = 0;
let df(rq(ee,ff,bb),ex) = (a/d)*dq(ex,fx,bx)/rq(ee,ff,bb);
let df(rq(ee,ff,bb),ey) = (a/d)*dq(ey,fy,by)/rq(ee,ff,bb);
let df(rq(ee,ff,bb),ez) = (a/d)*dq(ez,fz,bz)/rq(ee,ff,bb);
let df(rq(ee,ff,bb),fx) = (b/d)*dq(ex,fx,bx)/rq(ee,ff,bb);
let df(rq(ee,ff,bb),fy) = (b/d)*dq(ey,fy,by)/rq(ee,ff,bb);
let df(rq(ee,ff,bb),fz) = (b/d)*dq(ez,fz,bz)/rq(ee,ff,bb);
let df(rq(ee,ff,bb),ax) =(cc/d)*dq(ex,fx,bx)/rq(ee,ff,bb);
let df(rq(ee,ff,bb),ay) =(cc/d)*dq(ey,fy,by)/rq(ee,ff,bb);
let df(rq(ee,ff,bb),bz) =(cc/d)*dq(ez,fz,bz)/rq(ee,ff,bb);
let df(dq(ex,fx,bx),ex) =  a/d;
let df(dq(ey,fy,by),ey) =  a/d;
let df(dq(ez,fz,bz),ez) =  a/d;
let df(dq(ex,fx,bx),fx) =  b/d;
let df(dq(ey,fy,by),fy) =  b/d;
let df(dq(ez,fz,bz),fz) =  b/d;
let df(dq(ex,fx,bx),bx) = cc/d;
let df(dq(ey,fy,by),by) = cc/d;
let df(dq(ez,fz,bz),bz) = cc/d;
let df(dq(ex,fx,bx),ey) = 0;
let df(dq(ex,fx,bx),ez) = 0;
let df(dq(ex,fx,bx),fy) = 0;
let df(dq(ex,fx,bx),fz) = 0;
let df(dq(ex,fx,bx),by) = 0;
let df(dq(ex,fx,bx),bz) = 0;
let df(dq(ey,fy,by),ex) = 0;
let df(dq(ey,fy,by),ez) = 0;
let df(dq(ey,fy,by),fx) = 0;
let df(dq(ey,fy,by),fz) = 0;
let df(dq(ey,fy,by),bx) = 0;
let df(dq(ey,fy,by),bz) = 0;
let df(dq(ez,fz,bz),ex) = 0;
let df(dq(ez,fz,bz),ey) = 0;
let df(dq(ez,fz,bz),fx) = 0;
let df(dq(ez,fz,bz),fy) = 0;
let df(dq(ez,fz,bz),bx) = 0;
let df(dq(ez,fz,bz),by) = 0;
let rtd*rtd = d;
let rtc*rtc = c;
out sam;
procedure ocent(d1);
begin d2 := sub(sinh(z)=zz+zz**3/6+zz**5/120+zz**7/5040+zz**9/362881,
        cosh(z)=1+zz**2/2+zz**4/24+zz**6/720+zz**8/40320+zz**10/3628800
    **,01);
d3 := sub(zz=2*c*r*rob,d2);
```

```
;  <RATTS>QQ22.34;10   WED 15-NOV-79 8:35AM

d4 := sub(abx=0,aby=0,abz=0,rab=0,d3);
return d4;
end;
comment**********************************************************;
q1 := e**(-a*b*r*(ee,ff)**2/c);
q2 := e**(-c*cc*rg(ee,ff,bb)**2/d);
q3 := e**(-d*rg(ee,ff,bb)**2);
q4 := (2*b*rtd)/(rtc*c*rg(ee,ff,bb));
q5 := sinh(2*d*r*rg(ee,ff,bb));
q10 := q1*q2*q3*q4*q5;
q20 := (3*(DF(Q10,RZ,2)/(4*cc**2)+q10)/(2*cc))-q1*q3*q4*r**2*q5)/2;
let d=c*cc,rtd=rtc*cc;
qq20 := q20$
let cc=0;
q20 := qq20;
q21 := df(q20,fx)/(2*b)$
q24 := df(q20,fy)/(2*b)$
q27 := df(q20,fz)/(2*b);
q31 := df(q21,ex)/(2*a);
q34 := df(q24,ey)/(2*a)$
q37 := df(q27,ez)/(2*a)$
comment**********************************************************;
let rg(ee,ff)=rab,rg(ee,ff,bb)=rab,ef(ex,fx)=efx,ef(ey,fy)=efy,ef(ez,fz)=efz,
        AB(EX,FX,BX)=ABX,AB(EY,FY,BY)=ABY,AB(EZ,FZ,BZ)=ABZ;

let rg(ee,ff,bb)=rab,dg(ex,fx,bx)=abx,dg(ey,fy,by)=aby,dg(ez,fz,bz)=abz;
let d=c,rtd=rtc;
q11 := q20;
let cc=0;
let sinh(2*c*r*rab)=sinh(z),cosh(2*c*r*rab)=cosh(z);
q12 := sub(efx=0,efy=0,efz=0,ref=0,q11);
sub(sinh(z)=zz+zz**3/6+zz**5/120,cosh(z)=1+zz**2/2+zz**4/24+zz**6/720,q12);
saveas q16;
sub(zz=2*c*r*rab,q16);
saveas q17;
q18 := sub(abx=0,aby=0,abz=0,rab=0,q17);
q28 := sub(efx=0,efy=0,efz=0,ref=0,q27);
q29 := ocont(q28);
q32 := sub(efx=0,efy=0,efz=0,ref=0,q31);
q33 := ocont(q32);
q35 := sub(efx=0,efy=0,efz=0,ref=0,q34);
q36 := ocont(q35);
q38 := sub(efx=0,efy=0,efz=0,ref=0,q37);
q39 := ocont(q38);
shut san;
off nat;
factor rab;
out qfn;
write "comment (s,s)";
q20 := q11;
q12 := q12;
q18 := q18;
write "comment  (s,ox)";
```

```
;  <WATTS>QQQQ.24;10   WED 15-NOV-78 8:55AM

q21 := q21;
write "comment  (s,py)";
q24 := q24;
write "comment  (s,pz)";
q27 := q27;
q28 := q28;
q29 := q29;
write "comment  (px,px)";
q31 := q31;
q32 := q32;
q33 := q33;
write "comment  (py,py)";
q34 := q34;
q35 := q35;
q36 := q36;
write "comment  (pz,pz)";
q37 := q37;
q38 := q38;
q39 := q39;
write "end";
shut qfn;
on nat;
```

APPENDIX C

```
0001                SUBROUTINE JKPOT(H,DJ,DK,ISNX,IUNIT,HO,DKO,CLOSED)
0002                IMPLICIT REAL*8(A-H,O-Z)
0003                LOGICAL CLOSED
0004                DIMENSION H(1275),DJ(1275),DK(1275),HO(1),DKO(1)
0005                INTEGER*2 ISNX(50),IA,IB,IC,ID,IMU,IF
0006                COMMON/INTS/NINTS,LAST,IA(500),IB(500),IC(500),ID(500),
                   /            IMU(500),IF(500),V(500)
                    C
                    C   WILL ADD 2J-K TO H
                    C   IF CLOSED = FALSE, WILL ALSO ADD -K FOR OPEN SHELL TO HO
                    C   DJ AND DK ALWAYS CONTAIN TOTAL (CLOSED + OPEN) DENSITY
                    C     MATRICES
                    C   DKO CONTAINS "DK" DENSITY MATRIX FOR OPEN SHELL ORBITALS
                    C     ONLY
                    C   TOTAL COULOMB + EXCHANGE INTERACTION  IS
                    C     TR(D8*H)+ TR(DO*HO)
0007                11    READ(IUNIT)NINTS,LAST,IA,IB,IC,ID,IMU,IF,V
0008                      IF(NINTS.EQ.0) GO TO 30
0009                      DO 29 M=1,NINTS
0010                      I=IA(M)
0011                      J=IB(M)
0012                      K=IC(M)
0013                      L=ID(M)
0014                      MU=IMU(M)
0015                      VAL=V(M)
0016                      GO TO (12,13,14,15,16,17,18,19,20,21,22,23,24,25),MU
0017                12    II=ISNX(I)+I
0018                      H(II)=H(II)+VAL*(DJ(II)+DK(II))
0019                      IF(CLOSED) GO TO 29
0020                      HO(II)=HO(II)+VAL*DKO(II)
0021                      GO TO 29
0022                13    IJ=ISNX(I)+J
0023                      II=ISNX(I)+I
0024                      JJ=ISNX(J)+J
0025                      H(IJ)=H(IJ)+VAL*(DJ(IJ)+DJ(IJ)+DK(IJ))
0026                      H(II)=H(II)+VAL*DK(JJ)
0027                      H(JJ)=H(JJ)+VAL*DK(II)
0028                      IF(CLOSED) GO TO 29
0029                      HO(IJ)=HO(IJ)+VAL*DKO(IJ)
0030                      HO(II)=HO(II)+VAL*DKO(JJ)
0031                      HO(JJ)=HO(JJ)+VAL*DKO(II)
0032                      GO TO 29
0033                14    II=ISNX(I)+I
0034                      KK=ISNX(K)+K
0035                      IK=ISNX(I)+K
0036                      H(II)=H(II)+VAL*DJ(KK)
0037                      H(KK)=H(KK)+VAL*DJ(II)
0038                      H(IK)=H(IK)+DK(IK)*VAL
0039                      IF(CLOSED) GO TO 29
0040                      HO(IK)=HO(IK)+VAL*DKO(IK)
0041                      GO TO 29
0042                15    IL=ISNX(I)+L
0043                      II=ISNX(I)+I
0044                      H(IL)=H(IL)+VAL*(DJ(II)+DK(II))
0045                      H(II)=H(II)+(VAL+VAL)*(DJ(IL)+DK(IL))
0046                      IF(CLOSED) GO TO 29
0047                      HO(IL)=HO(IL)+VAL*DKO(II)
0048                      HO(II)=HO(II)+VAL*DKO(IL)*2.D0
0049                      GO TO 29
```

```
0050          16    KL=ISNX(K)+L
0051                IL=ISNX(I)+I
0052                IL=ISNX(I)+L
0053                IK=ISNX(I)+K
0054                H(KL)=H(KL)+VAL*DJ(II)
0055                H(II)=H(II)+(VAL+VAL)*DJ(KL)
0056                H(IL)=H(IL)+VAL*DK(IK)
0057                H(IK)=H(IK)+VAL*DK(IL)
0058                IF(CLOSED) GO TO 29
0059                HO(IL)=HO(IL)+VAL*DKO(IK)
0060                HO(IK)=HO(IK)+VAL*DKO(IL)
0061                GO TO 29
0062          17    JJ=ISNX(J)+J
0063                IJ=ISNX(I)+J
0064                H(JJ)=H(JJ)+(VAL+VAL)*(DJ(IJ)+DK(IJ))
0065                H(IJ)=H(IJ)+VAL*(DJ(JJ)+DK(JJ))
0066                IF(CLOSED) GO TO 29
0067                HO(JJ)=HO(JJ)+VAL*DKO(IJ)*2.DO
0068                HO(IJ)=HO(IJ)+VAL*DKO(JJ)
0069                GO TO 29
0070          18    JK=ISNX(J)+K
0071                GO TO 27
0072          19    JK=ISNX(K)+J
0073          27    KK=ISNX(K)+K
0074                IJ=ISNX(I)+J
0075                IK=ISNX(I)+K
0076                H(KK)=H(KK)+(VAL+VAL)*DJ(IJ)
0077                H(IJ)=H(IJ)+VAL*DJ(KK)
0078                H(JK)=H(JK)+VAL*DK(IK)
0079                H(IK)=H(IK)+VAL*DK(JK)
0080                IF(CLOSED) GO TO 29
0081                HO(JK)=HO(JK)+VAL*DKO(IK)
0082                HO(IK)=HO(IK)+VAL*DKO(JK)
0083                GO TO 29
0084          20    IJ=ISNX(I)+J
0085                JL=ISNX(J)+L
0086                JJ=ISNX(J)+J
0087                IL=ISNX(I)+L
0088                H(JL)=H(JL)+VAL*(DJ(IJ)+DJ(IJ)+DK(IJ))
0089                H(IJ)=H(IJ)+VAL*(DJ(JL)+DJ(JL)+DK(JL))
0090                H(JJ)=H(JJ)+(VAL+VAL)*DK(IL)
0091                H(IL)=H(IL)+VAL*DK(JJ)
0092                IF(CLOSED) GO TO 29
0093                HO(JL)=HO(JL)+VAL*DKO(IJ)
0094                HO(IJ)=HO(IJ)+VAL*DKO(JL)
0095                HO(JJ)=HO(JJ)+VAL*DKO(IL)*2.DO
0096                HO(IL)=HO(IL)+VAL*DKO(JJ)
0097                GO TO 29
0098          21    IL=ISNX(I)+L
0099                IJ=ISNX(I)+J
0100                II=ISNX(I)+I
0101                JL=ISNX(J)+L
0102                H(IL)=H(IL)+VAL*(DJ(IJ)+DJ(IJ)+DK(IJ))
0103                H(IJ)=H(IJ)+VAL*(DJ(IL)+DJ(IL)+DK(IL))
0104                H(II)=H(II)+(VAL+VAL)*DK(JL)
0105                H(JL)=H(JL)+VAL*DK(II)
0106                IF(CLOSED) GO TO 29
0107                HO(IL)=HO(IL)+VAL*DKO(IJ)
0108                HO(IJ)=HO(IJ)+VAL*DKO(IL)
```

```
0109                  HC(II)=HC(II)+VAL*DKD(JL)*2.D0
0110                  HC(JL)=HC(JL)+VAL*DKC(II)
0111                  GO TO 29
0112            22    IJ=ISNX(I)+J
0113                  KJ=ISNX(K)+J
0114                  JJ=ISNX(J)+J
0115                  IK=ISNX(I)+K
0116                  H(KJ)=H(KJ)+VAL*(DJ(IJ)+DJ(IJ)+DK(IJ))
0117                  H(IJ)=H(IJ)+VAL*(DJ(KJ)+DJ(KJ)+DK(KJ))
0118                  H(JJ)=H(JJ)+(VAL+VAL)*DK(IK)
0119                  H(IK)=H(IK)+VAL*DK(JJ)
0120                  IF(CLOSED) GO TO 29
0121                  HD(KJ)=HC(KJ)+VAL*DKC(IJ)
0122                  HD(IJ)=HD(IJ)+VAL*DKD(KJ)
0123                  HC(JJ)=HD(JJ)+VAL*DKC(IK)*2.D0
0124                  HD(IK)=HD(IK)+VAL*DKC(JJ)
0125                  GO TO 29
0126            23    JL=ISNX(J)+L
0127                  JK=ISNX(J)+K
0128                  GO TO 26
0129            24    JL=ISNX(J)+L
0130                  JK=ISNX(K)+J
0131                  GO TO 26
0132            25    JL=ISNX(L)+J
0133                  JK=ISNX(K)+J
0134            26    KL=ISNX(K)+L
0135                  IJ=ISNX(I)+J
0136                  IL=ISNX(I)+L
0137                  IK=ISNX(I)+K
0138                  H(JL)=H(JL)+VAL*DK(IK)
0139                  H(IK)=H(IK)+VAL*DK(JL)
0140                  H(IL)=H(IL)+VAL*DK(JK)
0141                  H(JK)=H(JK)+VAL*DK(IL)
0142                  IF(CLOSED) GO TO 290
0143                  HD(JL)=HD(JL)+VAL*DKC(IK)
0144                  HD(IK)=HD(IK)+VAL*DKC(JL)
0145                  HD(IL)=HC(IL)+VAL*DKC(JK)
0146                  HD(JK)=HD(JK)+VAL*DKD(IL)
0147             290  VAL=VAL+VAL
0148                  H(KL)=H(KL)+VAL*DJ(IJ)
0149                  H(IJ)=H(IJ)+VAL*DJ(KL)
0150            29    CONTINUE
0151            30    IF(LAST.EQ.0) GO TO 11
0152                  RETURN
0153                  END
```

```
0001                REAL FUNCTION QINT*8(K)
0002                IMPLICIT REAL*8(A-H,O-Z)
0003                EXTERNAL QS1,QS2,QS3,QS4,QS5,QS6,QS7,QS8,QS9
0004                LOGICAL INTC
0005                COMMON/INT/C,CK,RAB,CKC,IITYPE
0006                COMMON/SCLR/EXI,EXJ,ITYPE,JTYPE,ICNTR,JCNTR,R,CUT,CUT2,CUT3,
                   U ALPHAD,ALPHAQ,AP(4)
0007                COMMON/FCT/FCTCK(15),IFCT
0008                DATA SQ3/1.732050807568877D0/
0009                DATA PI/3.141592653589793D0/
0010                DATA NAPP,STEPM/4,200.D0/
0011                DATA EPR/1.D-16/
0012                DATA NGG/6/
                   C******* WARNING INTERVALS FOR GQ32 DECREASED************
                   C*********INCREASED INTEGRATION PARAM
                   C       PROGRAM USES INTEGRAL EXPRESSIONS OBTAINED FORM REDUCE
                   C   ITYPE=1,S
                   C          2,PX
                   C          3,PY
                   C          4,PZ,Z ASSUMED ALONG DIRECTION OF BOND
                   C   ICNTR=1, FOR A THE CLOSED SHELL ATOM ON WHICH EFFECTIVE POTENTIAL
                   C             IS CENTERED
                   C          2 FOR ATOM B, NON-CLOSED SHELL
                   C WARNING Q1 AND Q4  CODE DOES NOT CONTAIN EFX,EFY ETC TERMS
                   C WHICH WOULD BE NEEDED FOR MOLECULES
0013                IF(IFCT.EQ.0) GO TO 100
0014                CK=CUT2
0015                CKC=CUT3
0016                GO TO 99
0017             100  CK=CUT
0018              99  QINT=0.000
0019                IF(ITYPE.LE.4.AND.JTYPE.LE.4.OR.K.GT.8.OR.K.LE.0) GO TO 101
0020                WRITE(6,600) ITYPE,JTYPE,K
0021             600  FORMAT(1H ,25H**********ERROR********** /
                   U 1H ,6HITYPE=,I5,5X,6HJTYPE=,I5/
                   U 1H ,24HONLY TYPE .LE. 4 ALLOWED /
                   U 1H ,2HK=,I5,5X,16HK MUST BE 1 TO 8/
                   1 1H ,20H********************* //)
0022                RETURN
0023             101  IJTYPE=ITYPE+JTYPE
0024                IJCNTR=ICNTR+JCNTR
0025                GO TO (111,112,113,111,112,113,117,118),K
0026             111  IF((ITYPE.NE.1.OR.JTYPE.NE.4).AND.(JTYPE.NE.1.OR.ITYPE.NE.4)
                   U .AND.(ITYPE.NE.JTYPE)) GO TO 115
0027                IF(K.EQ.1.AND.(((IJTYPE.EQ.2.OR.IJTYPE.EQ.8).AND.IJCNTR.EQ.2)
                   U .OR.((IJTYPE.EQ.4.OR.IJTYPE.EQ.6).AND.IJCNTR.EQ.2)))GO TO 115
0028                IF(K.EQ.4.AND.((ITYPE.EQ.1.OR.JTYPE.EQ.1).AND.IJCNTR.EQ.2))GOTO115
0029                GO TO 120
0030             112  IF((JTYPE.NE.2.OR.(ITYPE.NE.1.AND.ITYPE.NE.4))
                   U .AND.(ITYPE.NE.2.OR.(JTYPE.NE.1.AND.JTYPE.NE.4))) GO TO 115
0031                IF(K.EQ.2.AND.(ITYPE.EQ.4.OR.JTYPE.EQ.4).AND.IJCNTR.EQ.2)
                   U GO TO 115
0032                IF(K.EQ.5.AND.(ITYPE.EQ.1.OR.JTYPE.EQ.1).AND.IJCNTR.EQ.2)
                   U GO TO 115
0033                GO TO 120
0034             113  IF((JTYPE.NE.3.OR.(ITYPE.NE.1.AND.ITYPE.NE.4))
                   U .AND.(ITYPE.NE.3.OR.(JTYPE.NE.1.AND.JTYPE.NE.4))) GO TO 115
0035                IF(K.EQ.3.AND.(ITYPE.EQ.4.OR.JTYPE.EQ.4).AND.IJCNTR.EQ.2)
                   U GO TO 115
```

```
0036                IF(K.EQ.6.AND.(ITYPE.EQ.1.OR.JTYPE.EQ.1).AND.IJCNTR.EQ.2)
                   U GO TO 115
0037                GO TO 120
0038          117   IF((ITYPE.NE.JTYPE).OR.(ITYPE.NE.2.AND.ITYPE.NE.3)) GO TO 115
0039                GO TO 120
0040          118   IF((ITYPE.NE.2.OR.JTYPE.NE.3)
                   U .AND.(ITYPE.NE.3.OR.JTYPE.NE.2)) GO TO 115
0041                GO TO 120
0042          115   RETURN
0043          120   C=EXI+EXJ
0044                HM3=0.0D0
0045                HM4=0.0D0
0046                HM5=0.0D0
0047                EXP0=1.D0
0048                IF(ICNTR.EQ.JCNTR) GO TO 130
0049                EXPREF=DEXP(-EXI*EXJ*R**2/C)
0050                IF(EXPREF.EQ.0.0D0) RETURN
0051                IF(ICNTR.EQ.1) GO TO 125
0052                RAB=EXI*R/C
0053                GO TO 150
0054          125   RAB=EXJ*R/C
0055                GO TO 150
0056          130   EXPREF=1.D0
0057                IF(ICNTR.EQ.1) GO TO 135
0058                RAB=R
0059                GO TO 150
0060          135   RAB=0.0D0
0061          150   EXP5M=1.D0
0062                EXP5=1.D0
0063                C2=C*C
0064                C3=C2*C
0065                RAB2=RAB*RAB
0066                RAB3=RAB2*RAB
0067                IF(IJTYPE.EQ.2) GO TO 190
0068                AEX=0.0D0
0069                AEY=0.0D0
0070                AEZ=RAB
0071                EFX=0.0D0
0072                EFY=0.0D0
0073                EFZ=0.0D0
0074                IF(ICNTR.NE.JCNTR) EFZ=R
0075                RAB4=RAB2*RAB2
0076                RAB5=RAB2*RAB3
0077                RAB6=RAB3*RAB3
0078                RAB7=RAB3*RAB4
0079                C4=C2*C2
0080                AEX2=0.0D0
0081                AEY2=0.0D0
0082                AEZ2=AEZ*AEZ
0083                AEZ3=AEZ*AEZ2
0084                IF(ITYPE.NE.1)A=EXJ
0085                IF(JTYPE.NE.1) A=EXI
0086                IF(ITYPE.EQ.1.OR.JTYPE.EQ.1) GO TO 190
           C DEFINITION OF A AND B IMPORTANT IF EXX,EFY.NE./
0087                IF(ICNTR.NE.2) GO TO 188
0088                A=EXI
0089                B=EXJ
0090                GO TO 199
0091          188   A=EXJ
```

```
0092                  B=EXI
0093             189  C5=C4*C
0094                  RAB8=RAB4*RAB4
0095                  RAB9=RAB8*RAB
0096                  ABX3=0.000
0097                  ABY3=0.000
0098                  ABX4=C.000
0099                  ABY4=C.000
0100                  ABZ4=ABZ2*ABZ2
0101                  EFX2=C.000
0102                  EFY2=0.000
0103                  EFZ2=EFZ*EFZ
0104                  GO TO 191
0105             190  IF(ICNTR+ITYPE.EQ.2.OR.JCNTR+JTYPE.EQ.2) EFZ=-EFZ
0106             191  REXP=DSQRT(-DLOG(1.D-16)/C)
                 C***BETTER INT LIMITS**********************************
0107                  RMIN=DMAX1(RAE-REXP,0.0D0)
0108                  RMAX=RAB+REXP
0109                  HEXP=(RAB+REXP-RMIN)/STEPM
0110                  HIN=HEXP*2.**(NAPP-1)
0111                  INTC=(C.LE..1D0.AND.RAB.LE.10.000).OR.
                     U (C.GE.1.D3.AND.RAB.LT..1D0)
0112                  IF(K.GE.4) GO TO 400
                 C        IF(INTC) GO TO 195
                 C******WARNING: ALL INTEGRALS DONE BY GAUSSIAN QUAD****************
                 C** REMOVE C FOR ITROMB
                 C******WARNING: GAUSSIAN QUAD CHANGED TO 4 INTERVALS
0113                  CALL NDQG32(RMIN,RMAX,DS1,HM1,NGQ)
0114                  IF(RAB.GE.1.D-10) CALL NDQG32(RMIN,RMAX,DS2,HM2,NGQ)
0115                  GO TO 199
0116             195  CALL ITROMB(DS1,HIN,ERR   ,NAPP,HM1,IERR,RMIN)
0117                  IF(RAE.GE.1.D-10)
                     .CALL ITROMB(DS2,HIN,ERR   ,NAPP,HM2,IERR,RMIN)
0118             199  CONTINUE
0119             601  FORMAT(1H ,4HHM1=,E27.16,5X,4HHM2=,E27.16)
0120             608  FORMAT(1H ,5HRMIN=,E15.6,5X,5HRMAX=,E15.6)
0121                  IF(IJTYPE.NE.2) GO TO 200
                 C  G1
0122                  QINT=PI*(2.*HM2/(C*RAE)-HM1/(C2*RAB2))/EXP0
0123                  QINT=QINT*EXPREF
0124                  RETURN
0125             200  CONTINUE
0126                  IF((ITYPE.EQ.1.OR.JTYPE.EQ.1).AND.K.NE.1) GO TO 205
                 C        IF(INTC) GO TO 204
0127                  IF(RAB.GE.1.D-10) CALL NDQG32(RMIN,RMAX,DS3,HM3,NGQ)
0128                  GO TO 205
0129             204  IF(RAB.GE.1.D-10)
                     .CALL ITROMB(DS3,HIN,ERR   ,NAPP,HM3,IERR,RMIN)
0130             205  CONTINUE
0131             602  FORMAT(1H ,4HHM3=,E27.16)
0132                  IF(ITYPE.NE.1.AND.JTYPE.NE.1) GO TO 300
0133                  GO TO (201,202,203),K
                 C  G1=  < S / Q1 / PZ >
0134             201  IF(RAB.LT.1.D-10) GO TO 211
0135                  Q1= (-PI*(6*RAB*HM2*ABZ2*C-2*HM2*RAB3*(2*ABZ*EFZ*C*A-(2*C2*
                     .  ABZ2-C))+2*ABZ*HM1*RAB2*EFZ*A-(HM1*(RAB2*(2*ABZ2*C-1)+3*ABZ2)
                     .  +4*RAB2*C2*HM3*ABZ2)))/(2*EXP0*EXP5*C3*RAB5)
0136                  QINT=Q1*EXPREF
0137                  RETURN
```

```
0138          211  CINT=4.*PI*HM1/3.
0139               RETURN
0140          202  IF(RAE.LT.1.D-10) GO TO 212
             C Q2= < S / Q2 / PX >
0141               Q2= ((((2*C*RAB2*ABX2-RAB2+3*ABX2)-2*ABX*EFX*A*RAB2)*HM1+2*((C
                  . -2*C2*ABX2)*RAB3-3*RAB*C*ABX2+2*ABX*EFX*C*A*RAB3)*HM2+4*HM3*
                  . C2*RAB2*ABX2)*PI)/(2*EXPC*EXP5*C3*RAB5)
0142               CINT=C2*EXPREF
0143               RETURN
0144          212  QINT=4.*PI*HM1/3.
0145               RETURN
0146          203  IF(RAB.LT.1.D-10) GO TO 213
             C Q3= < S / Q3 / PY >
0147               Q3= (-PI*(6*RAB*HM2*AEY2*C+2*ABY*((HM1*RAB2-2*HM2*RAB3*C)*EFY*A
                  . -(((2*ABY2*C-1)*RAB2+3*ABY2)*HM1-2*(2*C2*ABY2-C)*HM2*RAB3+4*
                  . HM3*C2*RAB2*ABY2))*EXPSM)/(2*C3*RAB5)*EXPREF
0148               QINT=C3
0149               RETURN
0150          213  CINT=4.*PI*HM1/3.
0151               RETURN
0152          300  CONTINUE
0153               IF(IJTYPE.NE.8) GO TO 302
             C     IF(INTC) GO TO 301
0154               IF(RAE.GE.1.D-10) CALL NDGG32(RMIN,RMAX,DS4,HM4,NGQ)
0155               GO TO 302
0156          301  IF(RAE.GE.1.D-10)
                  /CALL ITROMB(DS4,HIN,ERR  ,NAPP,HM4,IERR,RMIN)
0157          302  CONTINUE
0158          603  FORMAT(1H ,4HHM4=,F27.16)
0159               GO TO (310,320,330),K
0160          310  GO TO (115,312,313,314),ITYPE
             C < PX(E) / Q1(F) / PX(E) >
0161          312  Q1= (-ABZ*PI*(6*HM2*RAB3*C-(3*HM1*RAB2+4*C2*HM3*RAB4)))/(4*
                  . EXP0*EXP5*RAE7*C4)
0162               QINT=Q1*EXPREF
0163               RETURN
             C < PY(E) / Q1(F) / PY(E) >
0164          313  Q1= (-ABZ*PI*(6*HM2*RAB3*C-(3*HM1*RAB2+4*C2*HM3*RAB4)))/(4*
                  . EXP0*EXP5*RAB7*C4)
0165               QINT=Q1*EXPREF
0166               RETURN
             C < PZ / Q1 / PZ >
0167          314  Q1= (PI*(30*RAB*HM2*AEZ3*C-2*HM2*(ABZ*(9*RAB3*C+4*C2*RAB5+4*
                  . RAB5*B*EFZ2*C*A)-2*(3*RAB3*(2*C2*ABZ3+ABZ2*EFZ*(B-A)*C)+2*C2*
                  . RAB5*ABZ2*EFZ*(B-A)+2*C3*RAE5*ABZ3-RAB5*EFZ*(B-A)*C))+ABZ*(
                  . HM1*(9*RAB2+4*RAB4*(B*EFZ2*A+C))+12*C2*HM3*RAB4)-(HM1*(6*RAB2
                  . *(ABZ2*EFZ*(B-A)+2*AEZ3*C)+4*C2*RAB4*ABZ3+4*ABZ2*EFZ*RAB4*(B-
                  . A)*C-2*EFZ*RAB4*(B-A)+15*ABZ3)+24*RAE2*C2*HM3*ABZ3-8*RAB3*C3*
                  . HM4*ABZ3+8*C2*HM3*ABZ2*EFZ*RAB4*(B-A)+16*HM3*C3*RAB4*ABZ3)))/
                  . (4*EXP0*EXP5*RAB7*C4)
0168               QINT=Q1*EXPREF
0169               RETURN
             C < PX / Q2 / PZ >
0170          320  IF(ITYPE.EQ.2) GO TO 321
0171               A=EXI
0172               B=EXJ
0173               IF(ICNTR.EQ.1) EFZ=-B
0174               GO TO 322
0175          321  B=EXI
```

```
0176                    A=EXJ
0177                    IF(JCNTR.EQ.1) EFZ=-R
0178          322       Q2= (-PI*(2*B*(HM1*((2*C*RAB4*AEX2+3*RAB2*ABX2-RAB4)-2*ABX*EFX
                       . *A*RAB4)-2*HM2*(((3*RAB3*ABX2-RAB5)*C+2*C2*RAB5*ABX2)-2*ABX*
                       . EFX*C*A*RAB5)+4*HM3*(2*RAE4*AEX2)*EFZ+HM1*ABZ*((2*(6*RAB2*
                       . ABX2-RAB4)*C+4*C2*RAE4*AEX2-3*RAB2+15*ABX2)-2*(2*C*RAB4+3*
                       . RAB2)*ABX*EFX*A)-2*HM2*ABZ*((2*(6*RAB3*ABX2-RAB5)*C2-3*C*RAB3
                       . +4*C3*RAB5*ABX2)-2*(3*C*RAB3+2*C2*RAB5)*ABX*EFX*A+15*RAB*C*
                       . ABX2)+4*HM3*ABZ*(((6*RAB2*ABX2-RAB4)*C2+4*C3*RAE4*ABX2)-2*ABX
                       . *EFX*A*C2*RAB4)-8*HM4*ABZ*C3*RAB3*ABX2))/(4*EXP0*EXP5*RAB7*C4
                       . )
0179                    QINT=Q2*EXPREF
0180                    RETURN
                      C < PY / Q3 / PZ >
0181          330       IF(ITYPE.EQ.3) GO TO 331
0182                    A=EXJ
0183                    B=EXI
0184                    IF(JCNTR.EQ.1) EFZ=-R
0185                    GO TO 332
0186          331       A=EXI
0187                    B=EXJ
0188                    IF(ICNTR.EQ.1) EFZ=-R
0189          332       Q3= (PI*EXP5M*(30*RAB*AB7*HM2*ABY2*C+2*ABY*B*EFY*(2*EFZ*(HM1*
                       . RAB4-2*HM2*RAB5*C)*A-AEZ*(3*HM1*RAB2+2*HM1*RAB4*C-4*HM2*C2*
                       . RAB5-6*HM2*RAB3*C+4*HM3*C2*RAB4))+(2*EFZ*(HM1*((2*ABY2*C-1)*
                       . RAB4+3*RAB2*ABY2)-2*HM2*(2*C2*RAB5*ABY2+3*RAB3*ABY2*C-RAB5*C)
                       . +4*HM3*C2*RAB4*ABY2)*A-AEZ*(4*HM1*C2*RAB4*ABY2+12*HM1*RAE2*
                       . ABY2*C-3*HM1*RAB2-2*HM1*RAB4*C+15*HM1*ABY2-24*HM2*C2*RAB3*
                       . ABY2+4*HM2*C2*RAE5-8*HM2*C3*RAB5*ABY2+6*HM2*RAB3*C+24*HM3*C2*
                       . RAB2*ABY2-4*HM3*C2*RAB4+16*HM3*C3*RAE4*ABY2-8*HM4*C3*RAB3*
                       . ABY2))))/(4*C4*EXP0*RAE7)
0190                    QINT=Q3*EXPREF
0191                    RETURN
0192          400       CONTINUE
                      C       IF(INTC) GO TO 401
0193                    CALL NDQG32(RMIN,RMAX,DS5,HM1,NGQ)
0194                    IF(RAB.LT.1.D-10) GO TO 449
0195                    CALL NDQG32(RMIN,RMAX,DS6,HM2,NGQ)
0196                    CALL NDQG32(RMIN,RMAX,DS7,HM3,NGQ)
0197                    GO TO 449
0198          401       CALL ITROMB(DS5,HIN,ERR   ,NAPP,HM1,IERR,RMIN)
0199                    IF(RAB.LT.1.D-10) GO TO 449
0200                    CALL ITROMB(DS6,HIN,ERR   ,NAPP,HM2,IERR,RMIN)
0201                    CALL ITROMB(DS7,HIN,ERR   ,NAPP,HM3,IERR,RMIN)
0202          449       CONTINUE
0203          605       FORMAT(1H ,4HHM2=,E27.16,5X,4HHM2=,E27.16,5X,4HHM3=,E27.16)
0204                    IF(IJTYPE.NE.2) GO TO 450
                      C < S / 24 / S >
0205                    QINT=PI*(2.*HM3+3.*HM1/(2.*C2*RAB2)-3.*HM2/(C*RAB))/(C*RAB*EXP0)
0206                    QINT=QINT*EXPREF
0207                    RETURN
0208          450       CONTINUE
0209                    IF(((ITYPE.EQ.1.OR.JTYPE.EQ.1).AND.K.NE.4).OR.K.EQ.7) GO TO 452
                      C       IF(INTC) GO TO 451
0210                    IF(RAB.GE.1.D-10) CALL NDQG32(RMIN,RMAX,DS8,HM4,NGQ)
0211                    GO TO 452
0212          451       IF(RAB.GE.1.D-10)
                       /CALL ITROMB(DS8,HIN,ERR   ,NAPP,HM4,IERR,RMIN)
0213          452       CONTINUE
```

```
0214          606  FORMAT(1H ,4HHM4=,E27.16)
0215               IF(ITYPE.NE.1.AND.JTYPE.NE.1) GO TO 500
0216               GO TO (115,115,115,454,455,456),K
              C < S / Q4 / PZ >
0217          454  Q4= (PI*(90*RAB*HM2*AEZ3*C-6*HM2*(6*ABZ2*RAB3*EFZ*C*A-(2*C2*(3
                  . *RAB3*ABZ3-RAB5*ABZ)-9*RAB3*ABZ*C+2*RAB5*EFZ*C*A))+6*ABZ2*EFZ
                  . *(3*HM1*RAB2+4*RAB4*C2*HM3)*A-(18*HM1*RAB2*ABZ3*C-27*HM1*RAB2
                  . *ABZ+6*HM1*RAB4*EFZ*A-6*HM1*RAE4*ABZ*C+45*HM1*ABZ3+72*RAB2*C2
                  . *HM3*ABZ3-40*RAB4*C2*HM3*ABZ+24*RAB4*HM3*C3*ABZ3+8*C2*HM3*
                  . RAB6*EFZ*A-24*RAE3*C3*HM4*AEZ3+8*RAB5*C3*HM4*ABZ-8*HM3*RAB6*
                  . C3*ABZ)))/(8*EXP5*C4*EXP0*RAB7)
0218               CINT=Q4*EXPREF
0219               RETURN
              C < S / Q5 / PX >
0220          455  Q5= (ABZ*PI*(30*RAE*HM2*C*ABX2+(2*ABX*EFX*(3*HM1*RAB2-6*HM2*C*
                  . RAB3+4*HM3*C2*RAB4)*A-3*(2*C*RAB2*ABX2-RAB2+5*ABX2)*HM1-6*(C-
                  . 2*C2*ABX2)*HM2*RAB3-4*((6*RAB2*ABX2-RAB4)*C2+2*C3*RAB4*ABX2)*
                  . HM3+8*HM4*C3*RAB3*ABX2)))/(4*EXP0*EXP5*RAB7*C4)
0221               QINT=Q5*EXPREF*SQ3
0222               RETURN
              C < S / Q6 / PY >
0223          456  Q6= (ABZ*PI*EXP5M*(30*RAB*ABY2*HM2*C+2*ABY*EFY*(4*C2*RAB4*HM3+
                  . 3*RAB2*HM1-6*RAB3*HM2*C)*A-(4*C2*(6*RAB2*ABY2*HM3-3*RAB3*ABY2
                  . *HM2-RAB4*HM3)-8*C3*ABY2*(RAB3*HM4-RAB4*HM3)+3*RAB2*HM1*(2*
                  . ABY2*C-1)+6*RAB3*HM2*C+15*ABY2*HM1)))/(4*C4*EXP0*RAB7)
0224               QINT=Q6*EXPREF*SQ3
0225               RETURN
0226          500  CONTINUE
0227               IF(JJTYPE.NE.8) GO TO 502
              C        IF(INTC) GO TO 501
0228               IF(RAB.GE.1.D-10) CALL NDGG32(RMIN,RMAX,DS9,HM5,NGQ)
0229               GO TO 502
0230          501  IF(RAE.GE.1.D-10)
                  /CALL ITROM3(DS9,HIN,ERR  ,NAPP,HM5,IERR,RMIN)
0231          502  CONTINUE
0232          607  FORMAT(1H ,4HHM5=,E27.16)
0233               GO TO (115,115,115,540,550,560,570,580),K
0234          540  GO TO (115,542,543,544),ITYPE
              C < PX / Q4 / PX >
0235          542  IF(RAB.LT.1.D-10) GO TO 532
0236               Q4= (PI*(18*HM2*(5*ABZ2*RAB3-RAB5)*C-3*ABZ2*(15*HM1*RAB2+24*
                  . RAB4*C2*HM3-8*RAB5*C3*HM4)+(9*HM1*RAB4+16*C2*HM3*RAB6-8*C3*
                  . HM4*RAB7)))/(16*EXP5*EXP0*C5*RAB9)
0237               QINT=Q4*EXPREF
0238               RETURN
0239          532  CINT=-4.*PI*HM1/15.
0240               RETURN
              C < PY / Q4 / PY >
0241          543  IF(RAB.LT.1.D-10) GO TO 532
0242               Q4= (PI*(18*HM2*(5*ABZ2*RAB3-RAB5)*C-3*ABZ2*(15*HM1*RAB2+24*
                  . RAB4*C2*HM3-8*RAB5*C3*HM4)+(9*HM1*RAB4+16*C2*HM3*RAB6-8*C3*
                  . HM4*RAB7)))/(16*EXP5*EXP0*C5*RAB9)
0243               CINT=Q4*EXPREF
0244               RETURN
              C < PZ / Q4 / PZ >
0245          544  IF(RAB.LT.1.D-10) GO TO 545
0246               Q4= (-PI*(630*RAB*HM2*C*AEZ4-6*HM2*(2*ABZ2*(18*C2*RAB5+45*RAB3
                  . *C+6*RAB5*3*C*A*EFZ2+2*C3*RAB7)-(4*C2*(15*RAB3*ABZ4+3*RAB5*
                  . ABZ3*EFZ*(B-A)-RAB7*EFZ*ABZ*(B-A))+30*RAB3*ABZ3*EFZ*(B-A)*C+3
```

```
     .  *PAB5*(4*C3*ABZ4-6*EFZ*ABZ*(B-A)*C+3*C)+4*RAB7*B*C*A*EFZ2))+2
     .  *ABZ2*(135*HM1*RAB2+18*HM1*RAB4*B*A*EFZ2+54*HM1*RAB4*C+6*HM1*
     .  C2*RAB6+222*RAB4*C2*HM3+24*C2*HM3*RAB6*B*A*EFZ2-84*RAB5*C3*
     .  HM4+80*HM3*RAB6*C3+8*HM3*C4*RAB8+8*RAB6*C4*HM5-16*HM4*C4*RAB7
     .  )-(3*HM1*(30*RAB2*(ABZ3*EFZ*(B-A)+2*C*ABZ4)+3*RAB4*(4*C2*ABZ4
     .  +4*ABZ3*EFZ*(B-A)*C-6*EFZ*ABZ*(B-A)+3)-4*RAB6*(EFZ*ABZ*(B-A)*
     .  C-B*A*EFZ2)+105*ABZ4)+540*RAB2*C2*HM3*ABZ4+48*RAB4*(3*C2*HM3*
     .  ABZ3*EFZ*(B-A)+6*HM3*C3*ABZ4+C4*HM5*ABZ4)-8*C2*HM3*(5*RAB6*(2
     .  *EFZ*ABZ*(B-A)-1)-2*RAB8*E*A*EFZ2)-240*RAB3*C3*HM4*ABZ4-48*
     .  RAB5*HM4*(C3*ABZ3*EFZ*(B-A)+2*C4*ABZ4)+16*HM3*(3*RAB6*(C3*
     .  ABZ3*EFZ*(B-A)+C4*ABZ4)-C3*EFZ*ABZ*RAB8*(B-A))+8*C3*HM4*RAB7*
     .  (2*EFZ*ABZ*(B-A)-1)))))/(16*EXP5*EXP0*C5*RAB9)
0247                 QINT=C4*EXPREF
0248                 RETURN
0249          545    QINT=8.*PI*HM1/15.
0250                 RETURN
               C < PX / Q5 / PZ >
0251          550    IF(RAB.LT.1.D-10) GO TO 551
0252                 IF(ITYPE.EQ.2) GO TO 553
0253                 A=EXI
0254                 B=EXJ
0255                 IF(ICNTR.EQ.1) EFZ=-R
0256                 GO TO 552
0257          553    B=EXI
0258                 A=EXJ
0259                 IF(JCNTR.EQ.1) EFZ=-R
0260          552    Q5= (-PI*(ABZ2*(210*RAB*HM2*C*ABX2+(2*ABX*EFX*(8*RAB6*HM3*C3+6
     .  *HM1*C*RAB4+15*HM1*RAB2-30*HM2*C*RAB3-12*HM2*C2*RAB5+24*HM3*
     .  C2*RAB4-8*HM4*C3*RAB5)*A+8*RAB6*HM3*(C3-2*C4*ABX2)-3*HM1*(2*(
     .  10*RAB2*ABX2-RAB4)*C+4*C2*RAB4*ABX2-5*RAB2+35*ABX2)+6*HM2*(2*
     .  (10*RAB3*ABX2-RAB5)*C2-5*C*RAB3+4*C3*RAB5*ABX2)-12*HM3*((15*
     .  RAB2*ABX2-2*RAB4)*C2+8*C3*RAB4*ABX2)+8*HM4*((10*RAB3*ABX2-
     .  RAB5)*C3+4*C4*RAB5*ABX2)-16*HM5*C4*RAB4*ABX2))+2*ABZ*B*EFZ*(2
     .  *ABX*EFX*(4*RAB6*HM3*C2+3*(HM1*RAB4-2*HM2*C*RAB5))*A+(4*RAB6*
     .  HM3*(C2-2*C3*ABX2)-(3*HM1*(2*C*RAB4*ABX2+5*RAB2*ABX2-RAB4)-6*
     .  HM2*((5*RAB3*ABX2-RAB5)*C+2*C2*RAB5*ABX2)+24*HM3*C2*RAB4*ABX2
     .  -8*HM4*C3*RAB5*AFX2)))-(2*AEX*EFX*(4*RAB6*HM3*C2+3*HM1*RAB4-6
     .  *HM2*C*RAB5)*A+4*RAB6*HM3*(C2-2*C3*ABX2)-3*HM1*(2*C*RAB4*ABX2
     .  +5*RAB2*ABX2-RAB4)+6*HM2*((5*RAB3*ABX2-RAB5)*C+2*C2*RAB5*ABX2
     .  )-24*HM3*C2*RAB4*ABX2+8*HM4*C3*RAB5*ABX2)))/(8*EXP0*EXP5*C5*
     .  RAB9)
0261                 QINT=C5*EXPREF*SC3
0262                 RETURN
0263          551    QINT=4.*PI*HM1/15.*SQ3
0264                 RETURN
               C < PY / Q6 / PZ >
0265          560    IF(RAB.LT.1.D-10) GO TO 551
0266                 IF(ITYPE.EQ.3) GO TO 561
0267                 A=EXJ
0268                 B=EXI
0269                 IF(JCNTR.EQ.1) EFZ=-R
0270                 GO TO 562
0271          561    A=EXI
0272                 B=EXJ
0273                 IF(ICNTR.EQ.1) EFZ=-R
0274          562    Q6= (-PI*EXP5M*(210*RAB*ABY2*ABZ2*HM2*C+2*ABY*B*EFY*(2*ABZ*EFZ
     .  *A*(4*C2*RAB6*HM3+3*RAB4*HM1-6*RAB5*HM2*C)-(4*C2*(6*RAB4*ABZ2
     .  *HM3-3*RAB5*ABZ2*HM2-RAB6*HM3)-8*C3*ABZ2*(RAB5*HM4-RAB6*HM3)+
     .  15*RAB2*ABZ2*HM1-30*RAB3*ABZ2*HM2*C+3*RAB4*HM1*(2*ABZ2*C-1)+6
```

```
     .  *RAB5*HM2*C))+2*ABZ*EFZ*(4*C2*(6*RAB4*ABY2*HM3-3*RAB5*ABY2*
     .  HM2-RAB6*HM3)-8*C3*AEY2*(RAE5*HM4-RAB6*HM3)+15*RAB2*ABY2*HM1-
     .  30*RAB3*ABY2*HM2*C+3*RAB4*HM1*(2*ABY2*C-1)+6*RAE5*HM2*C)*A-(
     .  16*(RAB4*HM5-2*RAB5*HM4+RAB6*HM3)*C4*ABY2*ABZ2+4*C2*(45*RAB2*
     .  ABY2*ABZ2*HM3-30*RAB3*ABY2*ABZ2*HM2+3*RAB4*(ABY2*(ABZ2*HM1-2*
     .  HM3)-2*ABZ2*HM3)+3*RAB5*(ABY2+ABZ2)*HM2+RAB6*HM3)-8*C3*(10*
     .  RAB3*ABY2*ABZ2*HM4-12*RAE4*ABY2*ABZ2*HM3+RAB5*(ABY2*(3*ABZ2*
     .  HM2-HM4)-ABZ2*HM4)+ABY2*RAB6*HM3+ABZ2*RAB6*HM3)+15*RAB2*(ABY2
     .  *(4*ABZ2*C-1)-ABZ2)*HM1+30*RAB3*(ABY2+ABZ2)*HM2*C-3*RAB4*(2*
     .  ABY2*C+2*ABZ2*C-1)*HM1-6*RAB5*HM2*C+105*ABY2*ABZ2*HM1)))/(8*
     .  C5*RAB9*EXP0)
0275          GINT=C6*EXPREF*SQ3
0276          RETURN
0277     570  GO TO (115,572,573),ITYPE
0278     572  IF(RAB.LT.1.D-10) GO TO 574
0279          ANS5=-8*(((5*ABX2-ABY2)*RAB5+10*RAB3*(ABX2*ABY2-ABX4))*C3+4*C4
     .  *RAB5*(ABX2*ABY2-ABX4))*HM4+16*HM5*C4*RAB4*(ABX2*ABY2-ABX4)
0280          ANS4=2*ABX*EFX*(8*(15*HM1*RAE2*ABY2+6*HM1*RAE4*AEY2*C+6*HM1*
     .  RAB4-12*HM2*C2*RAB5*ABY2-30*HM2*RAB3*ABY2*C-12*HM2*RAB5*C+24*
     .  HM3*C2*RAB4*ABY2+8*HM3*C2*RAB6+8*HM3*C3*ABY2*RAB6-8*HM4*C3*
     .  RAB5*ABY2)-3*HM1*(5*RAB2*ABY2+2*RAB4*(ABY2*C+1))*A+6*HM2*(2*
     .  C2*RAB5*ABY2+5*RAB3*ABY2*C+2*RAB5*C)*A-8*HM3*A*(C2*(3*RAB4*
     .  ABY2+RAB6)+C3*ABY2*RAB6)+8*HM4*C3*RAB5*ABY2*A)-2*EFX*ABX3*(8*
     .  (15*HM1*RAB2+6*HM1*RAB4*C-12*HM2*C2*RAB5-30*HM2*RAB3*C+24*HM3
     .  *C2*RAB4+8*HM3*C3*RAE6-8*HM4*C3*RAB5)-3*HM1*A*(5*RAB2+2*RAB4*
     .  C)+6*HM2*A*(2*C2*RAB5+5*RAB3*C)-8*HM3*A*(3*C2*RAB4+C3*RAB6)+8
     .  *HM4*C3*RAB5*A)+4*(3*HM1*RAB4-6*HM2*RAB5*C+4*HM3*C2*RAB6)*8*
     .  EFX2*(ABX2-ABY2)*A+3*(5*((4*ABY2*C+5)*ABX2-4*ABX4*C-ABY2)*
     .  RAB2+2*(4*ABX2*C-1)*RAB4+4*C2*RAB4*(ABX2*ABY2-ABX4)+35*ABX2*
     .  ABY2-35*ABX4)*HM1-6*(4*(5*RAB3*(ABX2*ABY2-ABX4)+2*RAB5*ABX2)*
     .  C2+5*(5*ABX2-ABY2)*RAB3*C+4*C3*RAB5*(ABX2*ABY2-ABX4)-2*RAB5*C
     .  )*HM2+4*((6*(5*ABX2-ABY2)*RAB4+45*RAB2*(ABX2*ABY2-ABX4)-2*
     .  RAB6)*C2+8*(3*RAE4*(ABX2*ABY2-ABX4)+ABX2*RAB6)*C3+4*C4*RAB6*(
     .  ABX2*ABY2-ABX4))*HM3+ANS5
0281          ANS3=-ANS4
0282          ANS2=210*RAB*HM2*(ABX2*ABY2-AEX4)*C+ANS3
0283          ANS1=PI*ANS2*EXP5M
0284          Q7= ANS1/(8*C5*RAE9)*EXPREF
0285          QINT=Q7*SQ3/2.D0
0286          RETURN
0287     573  IF(RAB.LT.1.D-10) GO TO 575
0288          ANS4=-(2*ABY*EFY*(15*HM1*RAB2*ABX2+6*HM1*RAB4*ABX2*C+6*HM1*
     .  RAB4-12*HM2*C2*RAB5*ABX2-30*HM2*RAB3*ABX2*C-12*HM2*RAB5*C+24*
     .  HM3*C2*RAB4*ABX2+8*HM3*C2*RAB6+8*HM3*C3*ABX2*RAE6-8*HM4*C3*
     .  RAB5*ABX2)*(8-A)-2*B*(EFY*ABY3*(15*HM1*RAB2+6*HM1*RAB4*C-12*
     .  HM2*C2*RAB5-30*HM2*RAB3*C+24*HM3*C2*RAB4+8*HM3*C3*RAB6-8*HM4*
     .  C3*RAB5)+6*HM1*RAB4*EFY2*(ABX2-ABY2)*A-12*HM2*RAE5*EFY2*(ABX2
     .  -ABY2)*C*A+8*HM3*C2*EFY2*RAB6*(ABX2-ABY2)*A)+2*EFY*ABY3*A*(15
     .  *HM1*RAB2+6*HM1*RAB4*C-12*HM2*C2*RAB5-30*HM2*RAE3*C+24*HM3*C2
     .  *RAB4+8*HM3*C3*RAB6-8*HM4*C3*RAB5)+3*HM1*(4*C2*RAB4*ABX2*ABY2
     .  -4*C2*RAB4*ABY4+20*RAB2*AEX2*ABY2*C-5*RAB2*ABX2-20*RAB2*ABY4*
     .  C+25*RAB2*ABY2+8*RAB4*ABY2*C-2*RAB4+35*ABX2*ABY2-35*ABY4)-6*
     .  HM2*(4*C2*(5*RAB3*(ABX2*ABY2-ABY4)+2*RAB5*ABY2)+4*C3*RAB5*(
     .  ABX2*ABY2-ABY4)-5*RAE3*C*(AEX2-5*ABY2)-2*RAB5*C)+4*HM3*(C2*(
     .  45*RAB2*(ABX2*ABY2-ABY4)-6*RAB4*(ABX2-5*ABY2)-2*RAB6)+8*C3*(3
     .  *RAB4*(ABX2*ABY2-ABY4)+AEY2*RAB6)+4*C4*RAB6*(ABX2*ABY2-ABY4))
     .  -8*HM4*(C3*(10*RAB3*(ABX2*ABY2-ABY4)-RAB5*(ABX2-5*ABY2))+4*C4
     .  *RAB5*(ABX2*ABY2-ABY4))+16*HM5*C4*RAB4*(ABX2*ABY2-ABY4))
0289          ANS3=210*RAB*HM2*(ABX2*ABY2-ABY4)*C+ANS4
```

```
0290                ANS2=PI*ANS3*EXP5M
0291                ANS1=-ANS2
0292                Q7= ANS1/(8*C5*RAE9)*EXPREF
0293                QINT=Q7*SQ3/2.DO
0294                RETURN
0295          575   HM1=-HM1
0296          574   QINT=2.*PI*HM1/15.*SQ3
0297                RETURN
                C < PX / Q8 / PY >
0298          580   IF(RAB.LT.1.D-10) GO TO 581
0299                IF(ITYPE.EQ.2) GO TO 583
0300                A=EXJ
0301                B=EXI
0302                IF(JCNTR.EQ.1) EFZ=-R
0303                GO TO 582
0304          583   A=EXI
0305                B=EXJ
0306                IF(ICNTR.EQ.1) EFZ=-R
0307          582   Q8= (-PI*(210*RAB*ABY*HM2*C*ABX2+ABY*(2*AEX*EFX*B*(8*RAB6*HM3*
                    . (EFY*A*C2-C3)+6*EFY*(HM1*RAB4-2*HM2*C*RAB5)*A-3*(2*C*RAB4+5*
                    . RAB2)*HM1+6*(5*C*RAE3+2*(2*RAB5)*HM2-24*HM3*C2*RAB4+8*HM4*C3*
                    . RAB5)-8*RAB6*HM3*(EFY*(C2-2*C3*ABX2)*A-(C3-2*C4*ABX2))+2*EFY*
                    . (3*HM1*((5*RAB2*ABX2-RAB4)+2*C*RAB4*ABX2)-6*HM2*((5*RAB3*ABX2
                    . -RAB5)*C+2*C2*RAE5*AEX2)+24*HM3*C2*RAB4*ABX2-8*HM4*C3*RAB5*
                    . ABX2)*A-3*HM1*(2*(10*RAB2*ABX2-RAB4)*C+4*C2*RAB4*ABX2-5*RAB2+
                    . 35*AEX2)+6*HM2*(2*(10*RAE3*ABX2-RAB5)*C2-5*C*RAB3+4*C3*RAB5*
                    . ABX2)-12*HM3*((15*RAE2*AEX2-2*RAB4)*C2+9*C3*RAB4*ABX2)+8*HM4*
                    . ((10*RAB3*ABX2-RAB5)*C3+4*C4*RAB5*ABX2)-16*HM5*C4*RAB4*ABX2)+
                    . (8*ABX*EFX*B*RAB6*HM3*C2+6*AEX*EFX*B*HM1*RAB4-12*ABX*EFX*B*
                    . HM2*C*RAB5-4*RAB6*HM3*C2+8*RAB6*HM3*C3*ABX2+6*HM1*C*RAB4*ABX2
                    . +15*HM1*RAB2*ABX2-3*HM1*RAB4-30*HM2*C*RAB3*ABX2+6*HM2*C*RAB5-
                    . 12*HM2*C2*RAB5*AEX2+24*HM3*C2*RAB4*ABX2-8*HM4*C3*RAB5*ABX2)))
                    . /(8*RAB9*EXP0*EXP5*C5)
0308                CINT=Q8*FXPREF*SQ3
0309                RETURN
0310          581   CINT=2.*PI*HM1/15.*SQ3
0311                RETURN
0312                END
```

```
0001                  REAL FUNCTION DS1*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/C,DK,RAB,DKC
0004                  COMMON/POT/POTCK(15),IPOT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARND/
0006                  IF(R.EQ.0.0D0) GO TO 100
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C       GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0007              97  CUT=DK*R
0008                  CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+.5D0*CUT))
0009                  POTCK(1)=PEACH
0010                  GO TO 99
0011              98  CUT=(DK*R)**3
0012                  POTCK(1)=DALGAR
0013                  IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0014              99  IF(RAB.LT.1.D-10) GO TO 200
0015                  TGRAB=2.*C*RAB*R
0016                  IF(TGRAB.LT.1.D0) GO TO 199
0017                  EXPR1=DEXP(-C*(RAB-R)**2)
0018                  EXPR2=DEXP(-C*(RAB+R)**2)
0019                  DS1=CUT*(EXPR1-EXPR2)/(R*R*2.)
0020                  RETURN
0021             199  DS1=CUT*DSINH(TGRAB)*DEXP(-C*(R*R+RAB*RAB))
                     U /(R*R)
0022                  RETURN
0023             200  DS1=CUT*R*DEXP(-C*R*R)
0024                  RETURN
0025             100  DS1=0.0D0
0026                  RETURN
0027                  END
```

```
0001                  REAL FUNCTION DS2*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/C,CK,RAB,DKQ
0004                  COMMON/POT/POTCK(15),IPOT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNO/
0006                  IF(R.EQ.0.0D0) GO TO 100
0007                  TGRAB=2.*C*RAB*R
                C***  TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C     GO TO 98
                C***  PEACH'S CUTOFF FUNCTION  ***
0008                97  CUT=DK*R
0009                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+.5D0*CUT))
0010                    POTCK(2)=PEACH
0011                    GO TO 99
0012                98  CUT=(CK*R)**3
0013                    POTCK(2)=DALGAR
0014                    IF(CUT.GT.1.D-10) CUT=1.DC-DEXP(-CUT)
0015                99  IF(TGRAB.LT.1.D0) GO TO 199
0016                    EXPRM=DEXP(-C*(RAB-R)**2)
0017                    EXPRP=DEXP(-C*(RAB+R)**2)
0018                    DS2=CUT*(EXPRM+EXPRP)/(R**2.)
0019                    RETURN
0020                199 DS2=CUT*DCOSH(TGRAB)*CEXP(-C*(R*R+RAB*RAB))/R
0021                    RETURN
0022                100 DS2=0.0D0
0023                    RETURN
0024                    END
```

```
0001                REAL FUNCTION DS3*8(R)
0002                IMPLICIT REAL*8(A-H,O-Z)
0003                COMMON/INT/C,DK,RAB,DKQ
0004                COMMON/PUT/POTCK(15),IPCT
0005                DATA PEACH,DALGAR/ 5HPEACH,8HDALGARND/
0006                IF(R.EQ.0.0D0) GO TO 100
0007                TGRAB=2.*C*RAB*R
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C       GO TO 99
                C*** PEACH'S CUTOFF FUNCTION  ***
0008              97    CUT=DK*R
0009                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+.5D0*CUT))
0010                    POTCK(3)=PEACH
0011                    GO TO 99
0012              98    CUT=(DK*R)**3
0013                    POTCK(3)=DALGAR
0014                    IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0015              99    IF(TGRAB.LT.1.D0) GO TO 199
0016                    EXPRM=DEXP(-C*(RAB-R)**2)
0017                    EXPRP=DEXP(-C*(RAB+R)**2)
0018                    DS3=CUT*(EXPRM-EXPRP)/2.
0019                    RETURN
0020             199    DS3=CUT*DSINH(TGRAB)*DEXP(-C*(R*R+RAB*RAB))
0021                    RETURN
0022             100    DS3=0.0D0
0023                    RETURN
0024                    END
```

```
0001                  REAL FUNCTION DS4*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/C,DK,RAB,DKO
0004                  COMMON/POT/POTCK(15),IPCT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNO/
0006                  IF(R.EQ.0.000) GO TO 100
0007                  TGRAB=2.*C*RAB*R
                C***  TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C     GO TO 98
                C***  PEACH'S CUTOFF FUNCTION  ***
0008                97 CUT=DK*R
0009                  CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+.5D0*CUT))
0010                  POTCK(4)=PEACH
0011                  GO TO 99
0012                98 CUT=(DK*R)**3
0013                  POTCK(4)=DALGAR
0014                  IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0015                99 IF(TGRAB.LT.1.D0) GO TO 199
0016                  EXPRM=DEXP(-C*(RAB-R)**2)
0017                  EXPRP=DEXP(-C*(RAB+R)**2)
0018                  DS4=CUT*(EXPRM+EXPRP)*R/2.
0019                  RETURN
0020                199 DS4=CUT*DCOSH(TGRAB)*DEXP(-C*(R*R+RAB*RAB))*R
0021                  RETURN
0022                100 DS4=0.0D0
0023                  RETURN
0024                  END
```

```
0001                  REAL FUNCTION DSS*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/C,DK,RAB,DKQ
0004                  COMMON/PCT/PCTCK(15),IPCT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNC/
0006                  IF(R.EQ.0.0D0) GO TO 100
                C***  TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C     GO TO 98
                C***  PEACH'S CUTOFF FUNCTION  ***
0007              97  CUT=DKQ*R
0008                  CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*(.5D0+CUT/6.D0)))
0009                  POTCK(5)=PEACH
0010                  GO TO 99
0011              98  CUT=(DK*R)**4
0012                  POTCK(5)=DALGAR
0013                  IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0014              99  IF(RAB.LT.1.D-10) GO TO 200
0015                  TGRAB=2.*C*RAB*R
0016                  IF(TGRAB.LT.1.D0) GO TO 199
0017                  EXPRM=DEXP(-C*(RAB-R)**2)
0018                  EXPRP=DEXP(-C*(RAB+R)**2)
0019                  DSS=CUT*(EXPRM-EXPRP)/(R**4*2.)
0020                  RETURN
0021             199  DSS=CUT*DSINH(TGRAB)*DEXP(-C*(R*R+RAB*RAB))/
                     U R**4
0022                  RETURN
0023             200  DSS=CUT*R*DEXP(-C*R*R)
0024                  RETURN
0025             100  DSS=0.0D0
0026                  RETURN
0027                  END
```

```
0001                  REAL FUNCTION DS6*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/C,DK,RAB,DKQ
0004                  COMMON/PUT/POTCK(15),IFCT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARND/
0006                  IF(R.EQ.0.0D0) GO TO 100
0007                  TGRAB=2.*C*RAB*R
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C       GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0008                97  CUT=DKQ*R
0009                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*(.5D0+CUT/6.D0)))
0010                    POTCK(6)=PEACH
0011                    GO TO 99
0012                98  CUT=(DK*R)**4
0013                    IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0014                    POTCK(6)=DALGAR
0015                99  IF(TGRAB.LT.1.D0) GO TO 199
0016                    EXPRM=DEXP(-C*(RAB-R)**2)
0017                    EXPRP=DEXP(-C*(RAB+R)**2)
0018                    DS6=CUT*(EXPRM+EXPRP)/(R**3*2.)
0019                    RETURN
0020               199  DS6=CUT*DCOSH(TGRAB)*DEXP(-C*(R*R+RAB*RAB))
                   U /R**3
0021                    RETURN
0022               100  DS6=0.0D0
0023                    RETURN
0024                    END
```

```
0001                REAL FUNCTION DS7*8(R)
0002                IMPLICIT REAL*8(A-H,C-Z)
0003                COMMON/INT/C,DK,RAB,DKQ
0004                COMMON/POT/POTCK(15), IPCT
0005                DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNC/
0006                IF(R.EQ.0.0D0) GO TO 100
0007                TGRAB=2.*C*RAB*R
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C       GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0008              97    CUT=DKQ*R
0009                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*(.500+CUT/6.D0)))
0010                    POTCK(7)=PEACH
0011                    GO TO 99
0012              98    CUT=(DK*R)**4
0013                    POTCK(7)=DALGAR
0014                    IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0015              99    IF(TGRAB.LT.1.D0) GO TO 199
0016                    EXPRM=DEXP(-C*(RAB-R)**2)
0017                    EXPRP=DEXP(-C*(RAB+R)**2)
0018                    DS7=CUT*(EXPRM-EXPRP)/(R*R*2.)
0019                    RETURN
0020             199    DS7=CUT*DSINH(TGRAB)*DEXP(-C*(R*R+RAB*RAB))
                       U /(R*R)
0021                    RETURN
0022             100    DS7=0.000
0023                    RETURN
0024                    END
```

```
0001                REAL FUNCTION DS8*8(R)
0002                IMPLICIT REAL*8(A-H,O-Z)
0003                COMMON/INT/C,CK,RAB,CKC
0004                COMMON/POT/POTCK(15),IPOT
0005                DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNO/
0006                IF(R.EQ.0.0D0) GO TO 100
0007                TGRAB=2.*C*RAB*R
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C       GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0008              97    CUT=CKC*R
0009                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*(.5D0+CUT/6.D0)))
0010                    POTCK(8)=PEACH
0011                    GO TO 99
0012              98    CUT=(CK*R)**4
0013                    POTCK(8)=DALGAR
0014                    IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0015              99    IF(TGRAB.LT.1.D0) GO TO 199
0016                    EXPRM=DEXP(-C*(RAB-R)**2)
0017                    EXPRP=DEXP(-C*(RAB+R)**2)
0018                    DS8=CUT*(EXPRM+EXPRP)/(R*2.)
0019                    RETURN
0020             199    DS8=CUT*DCOSH(TGRAB)*DEXP(-C*(R*R+RAB*RAB))/R
0021                    RETURN
0022             100    DS8=0.0D0
0023                    RETURN
0024                    END
```

```
0001                  REAL FUNCTION DS9*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/C,DK,RAB,DKG
0004                  COMMON/POT/POTCK(15),IPOT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNO/
0006                  IF(R.EQ.0.0D0) GO TO 100
0007                  TGRAB=2.*C*RAB*R
                C***  TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C     GO TO 98
                C***  PEACH'S CUTOFF FUNCTION  ***
0008                97  CUT=DKG*R
0009                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*(.5D0+CUT/6.D0)))
0010                    POTCK(9)=PEACH
0011                    GO TO 99
0012                98  CUT=(DK*R)**4
0013                    POTCK(9)=DALGAR
0014                    IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0015                99  IF(TGRAB.LT.1.D0) GO TO 199
0016                    EXPRM=DEXP(-C*(RAB-R)**2)
0017                    EXPRP=DEXP(-C*(RAB+R)**2)
0018                    DS9=CUT*(EXPRM-EXPRP)/2.
0019                    RETURN
0020               199  DS9=CUT*DSINH(TGRAB)*DEXP(-C*(R*R+RAB*RAB))
0021                    RETURN
0022               100  DS9=0.0D0
0023                    RETURN
0024                    END
```

```
0001                  SUBROUTINE ITROMB(FCN,HIN,ERT,NAPP,VINT,IERR,X0)
0002                  IMPLICIT REAL*8 (A-H,O-Z)
0003                  DIMENSION AUX(10)
0004                  COMMON/ERR/NERR
                C     REQUIRES EXTERNAL STATEMENT FOR FCN
                C EVALUATES INT FORM X0 TO INF OF (FCN(X) DX)
                C     BY TRAPEZOIDAL RULE WITH ROMBERG EXTRAPOLATION
                C     PROG MAY RESULT IN ERROR IN LANDS ON SUCCESSIVE NODES OF FCN(X)
                C     HIN IN INTERVAL AT WHICH 1ST INTEGRATION DONE
                C     ERT IS MAX ERROR DESIRED
                C     NAPP IS MAX NO TIMES INTEGRAL EVALUATED BY TRAP. MUST BE .LE.10
0005                  IERR=0
0006                  NERR=0
0007                  R0=0.0D0
0008                  V=0.0D0
0009                  HS=HIN
0010                  H=HS
0011                  X=X0
0012                  A=.5D0*FCN(X)
0013                  DO 250 N=1,NAPP
0014                  NINT=0
0015               50 X=X+HS
0016                  NINT=NINT+1
0017                  T=FCN(X)
0018                  IF(NERR.NE.0) GO TO 400
0019                  IF(DABS(T).LE.ERT*DABS(A/NINT)) GO TO 100
0020                  A=A+T
0021                  GO TO 50
0022              100 A=A+T
0023                  IF(R0.NE.0.0D0) GO TO 150
0024                  NINT=NINT+1
0025                  X=X+HS
0026                  T=FCN(X)
0027                  IF(NERR.NE.0) GO TO 400
0028                  IF(DABS(T).LE.ERT*DABS(A/NINT)) GO TO 101
0029                  A=A+T
0030                  GO TO 50
0031              150 IF(X.LT.R0) GO TO 50
0032              101 AUX(N)=H*A+0.5D0*V
0033               20 FORMAT(1H ,2HN=,I5,5X,7HAUX(N)=D27.16)
0034                  IF(DABS(AUX(N)-V).LE.ERT*DABS(V)) GO TO 251
0035                  V=AUX(N)
0036                  IF(R0.EQ.0.0D0) R0=X
0037                  IF(N.EQ.NAPP) GO TO 250
0038                  HS=H
0039                  H=.5D0*H
0040                  X=X0+H
0041                  A=FCN(X)
0042              250 CONTINUE
0043               10 FORMAT(1H ,12HMAX VALUE X=,D13.6)
0044                  D=1.D0
0045                  NK=NAPP-1
0046                  DO 300 K=1,NK
0047                  D=4.*D
0048                  NJ=NAPP-K
0049                  Y=AUX(NJ+1)
0050                  DO 300 J=1,NJ
0051                  NA=NAPP-(K+J)+1
0052                  AA=AUX(NA)
```

```
0053                AUX(NA)=Y+(Y-AN)/(D-1.)
0054                IF(DABS(AUX(NA)-Y).LE.ERT*DABS(Y)) GO TO 301
0055                Y=AN
0056          300   CONTINUE
0057                IERR=1
                C   IMPLIES NAPP NOT LARGE ENOUGH
0058                SERR=AUX(NA)-AUX(NA+1)
0059                WRITE(6,500) SERR,AUX(NA)
0060          500   FORMAT(1H ,5X,6HERROR=,D27.16,5X,5HVINT=,D27.16)
0061          301   VINT=AUX(NA)
0062          600   FORMAT(1H ,10X,19HITRCMB CONV WITH K=,I5,5X,3HNK=,I5)
0063                RETURN
0064          251   VINT=AUX(N)
0065                RETURN
0066          400   IERR=NERR
0067                IF(NERR.NE.3) GO TO 401
0068                TR0=FCN(X-HS)
0069                NERR=0
0070          501   FORMAT(1H ,10HFCN AT R0=,D27.16)
0071                A=A+T
0072                GO TO 101
0073          401   VINT=H*A+.5*V
0074                WRITE(6,602) NERR
0075          602   FORMAT(1H0,13HITRCMB ERROR=,I5//)
0076                RETURN
0077                END
```

```
0001                  SUBROUTINE NDQG32(XMIN,XMAX,FCN,VINT,N)
0002                  IMPLICIT REAL*8(A-H,C-Z)
0003                  EXTERNAL FCN
                  C   DOES 32 POINT GAUSSIAN QUADRATURE N TIMES ON INTERVAL
                  C   XMIN TO XMAX IN EQUAL SUBINTERVALS
0004                  VINT=0.0D0
0005                  STEP=(XMAX-XMIN)/N
0006                  XMINI=XMIN
0007                  DO 100 IN=1,N
0008                  XMAXI=XMINI+STEP
0009                  CALL DQG32(XMINI,XMAXI,FCN,VINTI)
0010                  VINT=VINT+VINTI
0011              100 XMINI=XMAXI
0012                  RETURN
0013              600 FORMAT(1H ,10X,6HVINTI=,E27.16)
0014                  END
```

```
0001                  REAL FUNCTION RINT*8(IDUM)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  REAL*8 K,K2,K3,K4,K5,K6
0004                  REAL*4 SEC
0005                  LOGICAL INTC,LA
0006                  EXTERNAL GS1,GS2,GS3,GS4,GS5,GS6
0007                  COMMON/INT/C,DK,RAB,DKO,IITYPE
0008                  COMMON/SCLR/EXI,EXJ,ITYPE,JTYPE,ICNTR,JCNTR,R,K,CUT2,CUT3,
                     U ALPHAO,ALPHAC,AP(4),LA(6),RINTG(6)
0009                  COMMON/POT/PCTCK(15),IPCT
0010                  DATA PI,RTPI/3.14159265358979D0,1.77245385090551D0/
0011                  DATA NAPP,STEPM/3,1000.D0/
0012                  DATA NGQ/6/
                     C**********WARNING: INTERVALS FOR GAUSSIAN QUAD DECREASED***************
0013                  PHI(X)=DERF(X)
0014                  SINH(X)=DSINH(X)
0015                  COSH(X)=DCOSH(X)
                     C  PROGRAM USES INTEGRAL EXPRESSIONS OBTAINED FROM REDUCE
                     C  ITYPE=1,S
                     C         2,PX
                     C         3,PY
                     C         4,PZ: Z ASSUMED TO BE ALONG DIRECTION OF BOND
                     C  ICNTR=1 FOR A, THE CLOSED SHELL ATOM ON WHICH EFFECTIVE POTENTIAL
                     C           IS CENTERED
                     C         2 FOR ATOM B, NON-CLOSED SHELL
                     C
0016                  IITYPE=ITYPE
0017                  DO 98 I=1,6
0018                98 RINTG(I)=0.0D0
0019                  R1=0.0D0
0020                  R2=0.0D0
0021                  R3=0.0D0
0022                  R4=0.0D0
0023                  R5=0.0D0
0024                  R6=0.0D0
0025                  IF(IPCT .EQ.0) GO TO 100
0026                  DK=CUT2
0027                  DKO=CUT3
0028                  GO TO 99
0029               100 DK=K
0030                99 RINT=0.0D0
0031                  IF(ITYPE.LE.4.AND.JTYPE.LE.4) GO TO 101
0032                  WRITE(6,600) ITYPE,JTYPE
0033               600 FORMAT(1H ,25H**********ERROR********** /
                     U 1H ,6HITYPE=,I5,5X,6HJTYPE=,I5/
                     U 1H ,24HONLY TYPE .LE.4 ALLOWED /
                     U 1H ,20H**********  //)
0034                  RETURN
0035               101 IF(ITYPE.EQ.JTYPE.OR.(ITYPE.EQ.1.AND.JTYPE.EQ.4).OR.(ITYPE.EQ.4
                     U .AND.JTYPE.EQ.1)) GO TO 102
0036                  RETURN
0037               102 C=EXI+EXJ
0038                  EXPO=1.D0
0039                  IF(ICNTR.EQ.JCNTR) GO TO 110
0040                  EXPREF=DEXP(-EXI*EXJ*R**2/C)
0041                  IF(EXPREF.EQ.0.0D0) RETURN
0042                  IF(ICNTR.EQ.1) GO TO 105
0043                  RAB=EXI*R/C
0044                  GO TO 150
```

```
0045          105  RAB=EXJ*R/C
0046               GO TO 150
0047          110  EXPREF=1.D0
0048               IF(ICNTR.EQ.1) GO TO 115
0049               RAB=R
0050               GO TO 150
0051          115  RAB=0.0D0
0052          150  RTC=DSQRT(C)
0053               IF((ITYPE.NE.1.OR.JTYPE.NE.1).AND.ITYPE.NE.JTYPE.AND
                  U RAB.LT.1.D-10) GO TO 200
              C DEF
0054               EXP1   =  DEXP(K**2/(4*C))
0055               EXP2   =  DEXP(K*RAB)
0056               EXP5M=DEXP(-C*RAB**2)
0057               RAM    =  RAB - K/(2*C)
0058               RAP    =  RAB + K/(2*C)
0059               IF(LA(3)) GO TO 114
0060               PHIMC=DERFC(-RTC*RAM)
0061               PHIPC=DERFC(RTC*RAP)
0062               PHI0C=PHIPC
0063          114  EXP22=EXP2**2
0064               C2=C*C
0065               C3=C2*C
0066               C4=C2*C2
0067               C5=C4*C
0068               RAB2=RAB*RAB
0069               RAB3=RAB*RAB2
0070               K2=K*K
0071               K3=K2*K
0072               K4=K2*K2
0073               RAP2=RAP**2
0074               RAP3=RAP2*RAP
0075               RAM2=RAM**2
0076               RAM3=RAM2*RAM
0077               REXP=DSQRT(-DLOG(1.D-10)/C)
0078               RMIN=DMAX1(RAB-REXP,0.0D0)
0079               RMAX=RAB+REXP
0080               INTC=(C.LT..2D0.OR.(C.LE..2D0.AND.RAB.GT.10.D0)
                  U .OR.(C.GE.1.D3.AND.RAB.LT..1D0))
              C        IF(INTC) GO TO 120
              C*****WARNING: ALL INTEGRALS DONE BY DGG32
              C*****REMOVE C TO GET ITROMB************
0081               P3=3.
0082               P5=5.
0083               IF(RAB.GT.1.D-10) GO TO 116
0084               P3=2.
0085               P5=4.
0086          116  RMAX1=DMIN1(RMAX,1.D10**(1./P3))
0087               RMAX2=DMIN1(RMAX,1.D10**(1./P5))
0088               IF(.NOT.LA(1)) CALL NDQG32(RMIN,RMAX1,GS1,HM1,NGC)
0089               IF(.NOT.LA(2)) CALL NDQG32(RMIN,RMAX2,GS2,HM21,NGQ)
0090               GO TO 130
0091          120  HEXP=(RAB+REXP-RMIN)/STEPM
0092               HDEN1=(1.D8)**(1./3.)/STEPM
0093               HDEN2=1.D8**.2/STEPM
0094               HIN1=DMIN1(HEXP,HDEN1)*2**(NAPP-1)
0095               HIN2=DMIN1(HEXP,HDEN2)*2**(NAPP-1)
0096               IF(.NOT.LA(1)) CALL ITROMB(GS1,HIN1,1.D-8,NAPP,HM1,IERR,RMIN)
0097               IF(.NOT.LA(2)) CALL ITROMB(GS2,HIN2,1.D-8,NAPP,HM21,IERR,RMIN)
```

```
0098          130  CONTINUE
0099          617  FORMAT(1H ,4HHMU=,E15.6,5X,5HHM21=,E15.6)
0100               IF(ITYPE.NE.1.OR.JTYPE.NE.1) GO TO 200
              C < S / R / S > INTEGRALS
0101               IF(RAB.LT.1.D-10) GO TO 160
              C DEF R1 ETC
0102               IF(.NOT.LA(1)) R1=(2*PI*EXPREF*HM1)/(RAB*C)
0103               IF(.NOT.LA(2)) R2=(2*PI*EXPREF*HM21)/(RAB*C)
0104               IF(LA(3)) GO TO 131
0105               R3= (RTPI*PI*(K*(PHIPC*EXP22-PHIMC)+2*RAB*(PHIPC*EXP22+PHIMC)*
                  . C)*EXP1)/(4*RAB*RTC*EXP2*C2)*EXPREF
0106          131  IF(LA(4)) GO TO 132
0107               R4= (-PI*(4*K*RAB*RTC*RTPI*EXP1*(PHIPC*EXP22+PHIMC)*C-8*RAB*
                  . EXP2*EXP5M*C2+PHIPC*RTC*RTPI*EXP1*(4*C2*RAB2+K2+2*C)*EXP22-
                  . RTC*RTPI*EXP1*PHIMC*(4*C2*RAB2+K2+2*C)))/(8*RAB*EXP2*C4)*EXPREF
0108          132  IF(LA(5)) GO TO 133
0109               R5= (-PI*(2*K*(8*RAB*EXP2*EXP5M*C2-3*EXP1*RTC*RTPI*(2*C2*RAB2+
                  . C)*(PHIPC*EXP22-PHIMC))-6*RAB*EXP1*RTC*RTPI*(2*C2+K2*C)*(
                  . PHIPC*EXP22+PHIMC)-EXP1*RTC*RTPI*(PHIPC*EXP22*(8*C3*RAB3+K3)+
                  . PHIMC*(8*C3*RAB3-K3))))/(16*RAB*EXP2*C5)*EXPREF
0110          133  IF(LA(6)) GO TO 500
0111               R6= (-RTPI*PI*EXP1*(PHIPC*EXP22-PHIMC))/(2*RAB*RTC*EXP0*EXP2*C
                  . )
0112               R6=R6*EXPREF
0113               GO TO 500
0114          160  IF(.NOT.LA(1)) R1=(2*PI*HM1)/(      EXP0*C)
0115               IF(.NOT.LA(2)) R2=(2*PI*HM21)/(      EXP0*C)
              C DEF R3 ETC
0116               IF(LA(3)) GO TO 161
0117               R3= (-PI*(2*K*RTC-RTPI*(K2+2*C)*EXP1*PHI0C))/(2*RTC*C2)
0118          161  IF(LA(4)) GO TO 162
0119               R4= (-PI*(6*K*RTC*RTPI*EXP1*PHI0C*C+(RTC*RTPI*EXP1*PHI0C*K3-8*
                  . C2-2*K2*C)))/(4*C4)
0120          162  IF(LA(5)) GO TO 163
0121               R5= (-PI*(20*K*C2-RTC*EXP1*RTPI*PHI0C*(12*C2+12*K2*C+K4)+2*K3*
                  . C))/(8*C5)
0122          163  IF(LA(6)) GO TO 500
0123               R6= (-PI*(K*RTPI*EXP1*PHI0C-2*RTC))/(RTC*C)
0124               GO TO 500
0125          200  IF(RAB.NE.0.OR.ITYPE.EQ.JTYPE) GO TO 201
0126               RINT=0.000
0127               RETURN
0128          201  C6=C3*C3
0129               RAB4=RAB2*RAB2
0130               RAB5=RAB4*RAB
0131               IF(RAB.LT.1.D-10) GO TO 300
              C        IF(INTC) GO TO 170
0132               RMAX1=DMIN1(RMAX,1.D10**.5)
0133               RMAX2=DMIN1(RMAX,1.D10**.25)
0134               IF(.NOT.LA(1)) CALL NDQG32(RMIN,RMAX1,GS3,HM2,NGQ)
0135               IF(.NOT.LA(2)) CALL NDQG32(RMIN,RMAX2,GS4,HM22,NGQ)
0136               GO TO 171
0137          170  HDEN1=1.D4/STEPM
0138               HDEN2=1.D2/STEPM
0139               HIN1=DMIN1(HEXP,HDEN1)*2**(NAPP-1)
0140               HIN2=DMIN1(HEXP,HDEN2)*2**(NAPP-1)
0141               IF(.NOT.LA(1)) CALL ITROMB(GS3,HIN1,1.D-8,NAPP,HM2,IERR,RMIN)
0142               IF(.NOT.LA(2)) CALL ITROMB(GS4,HIN2,1.D-8, NAPP,HM22, IERR,RMIN)
0143          171  CONTINUE
```

```
0144          618  FORMAT(1H ,4HHM2=,E15.6,5X,5HHM22=,E15.6)
0145               IF(ITYPE.NE.1.AND.JTYPE.NE.1) GO TO 300
              C < S / R / P > INTEGRALS
0146               ABX=RAB
0147               EFX=R
0148               IF(ICNTR.EQ.JCNTR) GO TO 202
0149               IF(ICNTR+ITYPE.NE.2.AND.JCNTR+JTYPE.NE.2) EFX=-R
0150               GO TO 203
0151          202  EFX=0.000
0152          203  IF(ITYPE.EQ.4) B=EXJ
0153               IF(JTYPE.EQ.4) B=EXI
0154               IF(ICNTR.EQ.JCNTR) GO TO 210
              C DEF R1 ETC
0155               IF(.NOT.LA(1)) R1 =(PI*(2*RAB*C*ABX*HM2-2*B*RAB2*EFX*HM1-2
                  . *C*RAB2*ABX*HM1-ABX*HM1))/(      EXPC*RAB3*C2)
0156               R1=R1*EXPREF
              C DEF R2 ETC
0157               HM1=HM21
0158               HM2=HM22
0159               IF(.NOT.LA(2)) R2 =(PI*(2*RAB*C*ABX*HM2-2*B*RAB2*EFX*HM1-2
                  . *C*RAB2*ABX*HM1-ABX*HM1))/(      EXP0*RAB3*C2)
0160               R2=R2*EXPREF
0161               IF(LA(3)) GO TO 204
0162               R3= (-PI*(K*(PHIPC*RTPI*EXP1*(2*B*EFX*RAB2-ABX*(2*RAB2*C-1))*
                  . EXP22-RTPI*EXP1*PHIMC*(2*B*EFX*RAB2-ABX*(2*RAB2*C-1))+4*EXP2*
                  . RAB*RTC*ABX*EXP5M)-PHIPC*RTPI*EXP1*(RAB*ABX*K2-4*B*EFX*RAB3*C
                  . )*EXP22-RTPI*EXP1*PHIMC*(RAB*ABX*K2-4*B*EFX*RAB3*C)))/(8*EXP2
                  . *RTC*C3*RAB3)*EXPREF
0163          204  IF(LA(4)) GO TO 205
0164               R4= (-PI*(2*K*RTC*RTPI*EXP1*(RAB*ABX*(PHIPC*EXP22+PHIMC)*C-2*
                  . PHIPC*(2*B*EFX*C-ABX*C2)*RAB3*EXP22-2*PHIMC*(2*B*EFX*C-ABX*C2
                  . )*RAB3)+RAB*ABX*(PHIPC*RTC*RTPI*EXP1*K3*EXP22+(RTC*RTPI*EXP1*
                  . PHIMC*K3-4*EXP2*(2*C2+K2*C)*EXP5M))-PHIPC*RTC*RTPI*EXP1*(2*B*
                  . EFX*(4*C2*RAB4+K2*RAB2+2*RAB2*C)-ABX*((4*RAB2*C-1)*K2+4*C2*
                  . RAB2-2*C))*EXP22+(RTC*RTPI*EXP1*PHIMC*(2*B*EFX*(4*C2*RAB4+K2*
                  . RAB2+2*RAB2*C)-ABX*((4*RAB2*C-1)*K2+4*C2*RAB2-2*C))+16*EXP2*B
                  . *EFX*EXP5M*C2*RAB3)))/(16*EXP2*C5*RAB3)*EXPREF
0165          205  IF(LA(5)) GO TO 206
0166               R5= (-PI*(2*K*(12*RAB*EXP2*ABX*EXP5M*C2+2*B*EFX*(3*EXP1*RTC*
                  . RTPI*(PHIPC*EXP22-PHIMC)*(2*C2*RAB4+RAB2*C)-8*EXP2*EXP5M*C2*
                  . RAB3)-EXP1*RTC*RTPI*ABX*(PHIPC*EXP22-PHIMC)*(12*C2*RAB2+4*C3*
                  . RAB4-3*C)+8*EXP2*ABX*EXP5M*C3*RAB3)-RAB*ABX*(EXP1*RTC*RTPI*(
                  . PHIPC*EXP22+PHIMC)*(6*K2*C+K4)-4*EXP2*EXP5M*K3*C)+2*B*EXP1*
                  . RTC*RTPI*EFX*(PHIPC*EXP22*(12*C2*RAB3+8*C3*RAB5+6*K2*RAB3*C+
                  . K3*RAB2)+PHIMC*(12*C2*RAB3+8*C3*RAB5+6*K2*RAB3*C-K3*RAB2))-
                  . EXP1*RTC*RTPI*ABX*(PHIPC*EXP22*(12*C2*K2*RAB3+16*C3*RAB3+K3*(
                  . 6*RAB2*C-1))+PHIMC*(12*C2*K2*RAB3+16*C3*RAB3-K3*(6*RAB2*C-1))
                  . ))/(32*EXP2*C6*RAB3)*EXPREF
0167          206  IF(LA(6)) GO TO 500
0168               R6= (-PI*(K*RTPI*EXP1*RAB*ABX*(PHIPC*EXP22+PHIMC)-PHIPC*RTPI*
                  . EXP1*EXP22*(2*B*EFX*RAB2+ABX)+RTPI*EXP1*PHIMC*(2*B*EFX*RAB2+
                  . ABX)-4*EXP2*RAB*RTC*ABX*EXP5M))/(4*EXP2*RTC*C2*RAB3)*EXPREF
0169               GO TO 500
0170          210  IF(.NOT.LA(1)) R1 =((2*RAB*C*HM2-2*C*RAB2*HM1-HM1)*ABX*PI
                  . )/(      RAB3*C2)
0171               HM1=HM21
0172               HM2=HM22
              C DEF R2 ETC
0173               IF(.NOT.LA(2)) R2 =((2*RAB*C*HM2-2*C*RAB2*HM1-HM1)*ABX*PI
```

```
                     . )/(      RAB3*C2)
0174                  IF(LA(3)) GO TO 211
0175                  R3= (PI*ABX*(K*(PHIPC*RTPI*EXP1*(2*RAB2*C-1)*EXP22-RTPI*EXP1*
                     . PHIMC*(2*RAB2*C-1)-4*EXP2*RAB*RTC*EXP5M)+PHIPC*RTPI*EXP1*RAB*
                     . K2*EXP22+RTPI*EXP1*RAB*PHIMC*K2))/(8*EXP2*RTC*C3*RAB3)
0176            211   IF(LA(4)) GO TO 212
0177                  R4= (-PI*ABX*(2*K*RTC*RTPI*EXP1*(RAB*(PHIPC*EXP22+PHIMC)*C+2*
                     . PHIPC*C2*RAB3*EXP22+2*PHIMC*C2*RAB3)+RAB*(PHIPC*RTC*RTPI*EXP1
                     . *K3*EXP22+(RTC*RTPI*EXP1*PHIMC*K3-4*EXP2*(2*C2+K2*C)*EXP5M))+
                     . PHIPC*RTC*RTPI*EXP1*((4*RAB2*C-1)*K2+4*C2*RAB2-2*C)*EXP22-RTC
                     . *RTPI*EXP1*PHIMC*((4*RAB2*C-1)*K2+4*C2*RAB2-2*C)))/(16*EXP2*
                     . C5*RAB3)
0178            212   IF(LA(5)) GO TO 213
0179                  R5= (-PI*ABX*(2*K*(12*RAB*EXP2*EXP5M*C2-EXP1*RTC*RTPI*(PHIPC*
                     . EXP22-PHIMC)*(12*C2*RAB2+4*C3*RAB-3*C)+8*EXP2*EXP5M*C3*RAB3)
                     . -RAB*(EXP1*RTC*RTPI*(PHIPC*EXP22+PHIMC)*(6*K2*C+K4)-4*EXP2*
                     . EXP5M*K3*C)-EXP1*RTC*RTPI*(PHIPC*EXP22*(12*C2*K2*RAB3+16*C3*
                     . RAB3*K3*(6*RAB2*C-1))+PHIMC*(12*C2*K2*RAB3+16*C3*RAB3-K3*(6*
                     . RAB2*C-1)))))/(32*EXP2*C6*RAB3)
0180            213   IF(LA(6)) GO TO 500
0181                  R6= (-PI*ABX*(K*RTPI*EXP1*RAB*(PHIPC*EXP22+PHIMC)-PHIPC*RTPI*
                     . EXP1*EXP22+RTPI*EXP1*PHIMC-4*EXP2*RAB*RTC*EXP5M))/(4*EXP2*RTC
                     . *C2*RAB3)
0182                  GO TO 500
0183            300   C7=C6*C
0184                  RAB6=RAB3*RAB3
0185                  RAB7=RAB6*RAB
0186                  K5=K4*K
0187                  K6=K5*K
0188                  IF(RAB.LT.1.D-10) GO TO 309
0189                  IF(ITYPE.NE.4) GO TO 302
                C     IF(INTC) GO TO 301
0190                  RMAX1=DMIN1(RMAX,1.D10)
0191                  RMAX2=DMIN1(RMAX,1.D10**(1./3.))
0192                  IF(.NOT.LA(1)) CALL NDQG32(RMIN,RMAX1,GS5,HM3,NGQ)
0193                  IF(.NOT.LA(2)) CALL NDQG32(RMIN,RMAX2,GS6,HM6,NGQ)
0194                  GO TO 302
0195            301   HDEN1=1.D8/STEPM
0196                  HDEN2=(1.D8)**(1./3.)/STEPM
0197                  HIN1=DMIN1(HEXP,HDEN1)*2**(NAPP-1)
0198                  HIN2=DMIN1(HEXP,HDEN2)*2**(NAPP-1)
0199                  IF(.NOT.LA(1)) CALL ITRGMB(GS5,HIN1,1.D-8,NAPP,HM3,IERR,RMIN)
0200                  IF(.NOT.LA(2)) CALL ITRGMB(GS6,HIN2,1.D-8,NAPP,HM6,IERR,RMIN)
0201            302   CONTINUE
0202            619   FORMAT(1H ,4HHM3=,E15.6,5X,4HHM6=,E15.6)
               C < P / R / P > INTEGRALS
0203            309   HM23=HM6
0204                  ABX=0.0D0
0205                  EFX=0.0D0
0206                  IF(ITYPE.NE.4) GO TO 310
0207                  ABX=RAB
0208                  IF(ICNTR.NE.JCNTR) EFX=R
0209            310   ABX2=ABX*ABX
0210                  EFX2=EFX*EFX
0211                  IF(ICNTR.NE.2) GO TO 311
0212                  A=EXI
0213                  B=EXJ
0214                  GO TO 312
0215            311   A=EXJ
```

```
0216                D=EXI
0217          312   IF(RAB.LT.1.D-10) GO TO 350
0218                IF(ICNTR.EQ.JCNTR) GO TO 320
              C DEF R1
0219                IF(.NOT.LA(1)) R1 =(PI*(4      *RAB2*C2*ABX2*HM3-6*RAB*C*
                   . ABX2*HM2-2*A*(2*B*RAB4*EFX2*HM1-2*C*ABX*EFX*(
                   . RAB3*HM2-RAB4*HM1)+RAB2*ABX*EFX*HM1)-2*B*ABX*EFX*
                   . (2*C*(RAB3*HM2-RAB4*HM1)-RAB2*HM1)+2*C*(2*
                   . RAB2*ABX2*HM1+RAB3*HM2)-RAB2*HM1-8*RAB3*C2*ABX2*HM2
                   . +4*RAB4*C2*ABX2*HM1+3*ABX2*HM1))/(2*      EXP0
                   . *RAB5*C3)
0220                R1=R1*EXPREF
0221                HM1=HM21
0222                HM2=HM22
0223                HM3=HM23
              C DEF R2 ETC
0224                IF(.NOT.LA(2)) R2 =(PI*(4      *RAB2*C2*AEX2*HM3-6*RAB*C*
                   . ABX2*HM2-2*A*(2*B*RAB4*EFX2*HM1-2*C*ABX*EFX*(
                   . RAB3*HM2-RAB4*HM1)+RAB2*AEX*EFX*HM1)-2*B*ABX*EFX*
                   . (2*C*(RAB3*HM2-RAB4*HM1)-RAB2*HM1)+2*C*(2*
                   . RAB2*ABX2*HM1+RAB3*HM2)-RAB2*HM1-8*RAB3*C2*ABX2*HM2
                   . +4*RAB4*C2*ABX2*HM1+3*ABX2*HM1))/(2*      EXP0
                   . *RAB5*C3)
0225                R2=R2*EXPREF
0226                IF(LA(3)) GO TO 313
0227                R3= (PI*(K*(PHIPC*RTPI*EXP1*(2*B*(ABX*EFX*(RAB2-2*RAB4*C)-2*
                   . RAB4*EFX2*A)-2*AEX*EFX*(RAB2-2*RAB4*C)*A-(2*ABX2*C+1)*RAB2+4*
                   . RAB4*C+3*ABX2)*EXP22-RTPI*EXP1*PHIMC*(2*E*(AEX*EFX*(RAB2-2*
                   . RAB4*C)-2*RAB4*EFX2*A)-2*ABX*EFX*(RAB2-2*RAB4*C)*A-(2*ABX2*C+
                   . 1)*RAB2+4*RAB4*C+3*AEX2)+4*EXP2*RTC*(3*RAB*AEX2+2*B*ABX*EFX*
                   . RAB3-(2*ABX*EFX*A+1)*RAB3)*EXP5M)-PHIPC*RTPI*EXP1*(3*RAB*K2*
                   . ABX2+2*B*(ABX*EFX*K2*RAB3+4*RAB5*EFX2*C*A)-2*ABX*EFX*K2*RAB3*
                   . A-(2*ABX2*C+1)*K2*RAB3-4*C2*RAB5-K3*RAB2*ABX2)*EXP22-RTPI*
                   . EXP1*PHIMC*(3*RAB*K2*ABX2+(2*B*(ABX*EFX*K2*RAB3+4*RAB5*EFX2*C
                   . *A)-2*ABX*EFX*K2*RAB3*A-(2*AEX2*C+1)*K2*RAB3-4*C2*RAB5+K3*
                   . RAB2*ABX2))))/(16*EXP2*RTC*C4*RAB5)*EXPREF
0228          313   IF(LA(4)) GO TO 314
0229                ANS3=(RTC*RTPI*EXP1*PHIMC*(2*B*(ABX*EFX*((RAB2-4*RAB4*C)*K2-4*
                   . C2*RAB4+K3*RAB3+2*RAB2*C)-8*C2*EFX2*RAB6*A-2*K2*RAB4*EFX2*A-4
                   . *RAB4*EFX2*C*A)-2*ABX*EFX*((RAB2-4*RAB4*C)*K2-4*C2*RAB4+K3*
                   . RAB3+2*RAB2*C)*A+4*(K2*RAB4*ABX2-RAB2*ABX2+2*RAB4)*C2-((2*
                   . ABX2*C+1)*RAB2-6*RAB4*C-3*ABX2)*K2-(4*ABX2*C+1)*K3*RAB3+B*C3*
                   . RAB6+K4*RAB2*ABX2-2*RAB2*C+6*ABX2*C)-4*EXP2*(2*B*(ABX*EFX*(2*
                   . C2+K2*C)*RAB3+4*C2*RAB5*EFX2*A)-2*ABX*EFX*(2*C2+K2*C)*RAB3*A-
                   . (2*(K2*ABX2+1)*C2*RAB3+4*C3*RAB5+K2*RAB3*C))*EXP5M)
0230                ANS2=2*(*RTC*RTPI*EXP1*(3*RAB*(PHIPC*EXP22+PHIMC)*ABX2*C+PHIPC
                   . *(2*B*(ABX*EFX*(2*C2*RAB5+RAB3*C)+4*RAB5*EFX2*C*A)-2*ABX*EFX*
                   . (2*C2*RAB5+RAB3*C)*A-2*(RAB3*ABX2+3*RAB5)*C2-RAB3*C)*EXP22+
                   . PHIMC*(2*B*(ABX*EFX*(2*C2*RAB5+RAB3*C)+4*RAB5*EFX2*C*A)-2*ABX
                   . *EFX*(2*C2*RAB5+RAB3*C)*A-2*(RAB3*ABX2+3*RAB5)*C2-RAB3*C))+3*
                   . RAB*(PHIPC*RTC*RTPI*EXP1*K3*EXP22+(RTC*RTPI*EXP1*PHIMC*K3-4*
                   . EXP2*(2*C2+K2*C)*EXP5M))*ABX2-PHIPC*RTC*RTPI*EXP1*(2*B*(ABX*
                   . EFX*((RAB2-4*RAB4*C)*K2-4*C2*RAB4-K3*RAB3+2*RAB2*C)-8*C2*EFX2
                   . *RAB6*A-2*K2*RAB4*EFX2*A-4*RAB4*EFX2*C*A)-2*ABX*EFX*((RAB2-4*
                   . RAB4*C)*K2-4*C2*RAB4-K3*RAB3+2*RAB2*C)*A+4*(K2*RAB4*ABX2-RAB2
                   . *ABX2+2*RAB4)*C2-((2*ABX2*C+1)*RAB2-(*RAB4*C-3*AEX2)*K2+(4*
                   . ABX2*C+1)*K3*RAB3+B*C3*RAB6+K4*RAB2*ABX2-2*RAB2*C+6*ABX2*C)*
                   . EXP22+ANS3
0231                ANS1=PI*ANS2
```

```
0232                 A4= ANS1/(32*EXP2*C6*RAB5)*EXPREF
0233           314   IF(LA(5)) GO TO 315
0234                 ANS3=EXP1*RTC*RTP1*(PHIPC*EXP22*(2*ABX*EFX*(12*C2*K2*RAB5+16*
                    . C3*RAB5+6*K2*RAB3*C-K3*(RAB2-6*RAB4*C)+K4*RAB3)*A+12*C2*(2*K2
                    . *(RAB3*ABX2+RAB5)+K3*RAB4*ABX2)+8*C3*RAB5*(K2*AEX2+5)+16*C4*
                    . RAB7+6*K2*RAB3*C-K3*(RAB2-8*RAB4*C-3*ABX2)+K4*RAB3*(6*ABX2*C+
                    . 1)+K5*RAB2*ABX2)*PHIMC*(2*AEX*EFX*(12*C2*K2*RAB5+16*C3*RAB5+6
                    . *K2*RAB3*C+K3*(RAB2-6*RAB4*C)+K4*RAB3)*A+12*C2*(2*K2*(RAB3*
                    . ABX2+RAB5)-K3*RAB4*ABX2)+8*C3*RAB5*(K2*ABX2+5)+16*C4*RAB7+6*
                    . K2*(RAB3*C+K3*(RAB2-8*RAB4*C-3*ABX2)+K4*RAB3*(6*AFX2*C+1)-K5*
                    . RAB2*ABX2))-4*EXP2*EXP5M*K3*RAB3*(2*ABX*EFX*C*A+(4*C2*ABX2*C)
                    . )
0235                 ANS2=2*K*(36*RAB*EXP2*EXP5M*C2*ABX2-2*B*(EXP1*RTC*RTP1*(PHIPC*
                    . EXP22-PHIMC)*(ABX*EFX*(12*C2*RAB4+4*C3*RAB6-3*RAB2*C)+12*C2*
                    . EFX2*RAB6*A+6*RAB4*EFX2*C*A)-4*EXP2*EXP5M*(ABX*EFX*(3*C2*RAB3
                    . +2*C3*RAB5)+4*C2*RAB5*EFX2*A))+EXP1*RTC*RTP1*(PHIPC*EXP22-
                    . PHIMC)*(24*ABX*EFX*C2*RAB4*A+8*ABX*EFX*C3*RAB6*A-6*ABX*EFX*
                    . RAB2*C*A-12*C2*RAB2*ABX2+18*C2*RAB4+12*C3*RAB4*ABX2+16*C3*
                    . RAB6-3*RAB2*C+9*ABX2*C)-4*EXP2*EXP5M*(2*ABX*EFX*(3*C2*RAB3+2*
                    . C3*RAB5)*A+3*(C2*RAB3+2*C3*(RAB3*ABX2+RAB5)))-3*RAB*ABX2*(
                    . EXP1*RTC*RTP1*(PHIPC*EXP22+PHIMC)*(6*K2*C+K4)-4*EXP2*EXP5M*K3
                    . *C)-2*B*(EXP1*RTC*RTP1*(PHIPC*EXP22*(ABX*EFX*(12*C2*K2*RAB5+
                    . 16*C3*RAB5+6*K2*RAB3*C-K3*(RAB2-6*RAB4*C)+K4*RAB3)+24*C2*RAB5
                    . *EFX2*A+16*C3*EFX2*RAB7*A+12*K2*RAB5*EFX2*C*A+2*K3*RAB4*EFX2*
                    . A)+PHIMC*(ABX*EFX*(12*C2*K2*RAB5+16*C3*RAB5+6*K2*RAB3*C+K3*(
                    . RAB2-6*RAB4*C)+K4*RAB3)+24*C2*RAB5*EFX2*A+16*C3*EFX2*RAB7*A+
                    . 12*K2*RAB5*EFX2*C*A-2*K3*RAB4*EFX2*A))-4*EXP2*ABX*EFX*EXP5M*
                    . K3*RAB3*C)+ANS3
0236                 ANS1=PI*ANS2
0237                 R5= ANS1/(64*EXP2*C7*RAE5)*EXPREF
0238           315   IF(LA(6)) GO TO 500
0239                 R6= (PI*(K*RTP1*EXP1*(PHIPC*EXP22+PHIMC)*(2*B*ABX*EFX*RAB3+3*
                    . RAB*ABX2-2*ABX*EFX*RAB3*A-RAE3)-PHIPC*RTP1*EXP1*EXP22*(2*B*(
                    . ABX*EFX*RAB2-2*RAB4*EFX2*A)-2*ABX*EFX*RAB2*A+K2*RAB2*ABX2-
                    . RAB2+2*RAB4*C+3*ABX2)+RTP1*EXP1*PHIMC*(2*B*(ABX*EFX*RAB2-2*
                    . RAB4*EFX2*A)-2*ABX*EFX*RAB2*A+K2*RAB2*ABX2-RAB2+2*RAB4*C+3*
                    . ABX2)-4*EXP2*(((2*ABX2*C-1)-2*ABX*EFX*A)*RAB3+2*B*ABX*EFX*
                    . RAB3+7*RAB*ABX2)*RTC*EXP5M))/(8*EXP2*RTC*C3*RAB5)*EXPREF
0240                 GO TO 500
0241           320   IF(.NOT.LA(1)) R1 =((4       *RAB2*C2*ABX2*HM3-6*RAB*C*ABX2
                    . *HM2+2*C*(2*RAB2*ABX2*HM1+RAB3*HM2)-RAB2*HM1-8
                    . *RAB3*C2*ABX2*HM2+4*RAB4*C2*ABX2*HM1+3*ABX2*HM1)
                    . *PI)/(2*       RAB5*C3)
               C DEF
0242                 HM1=HM21
0243                 HM2=HM22
0244                 HM3=HM23
0245                 IF(.NOT.LA(2)) R2 =((4       *RAB2*C2*ABX2*HM3-6*RAB*C*ABX2
                    . *HM2+2*C*(2*RAB2*ABX2*HM1+RAB3*HM2)-RAB2*HM1-8
                    . *RAB3*C2*ABX2*HM2+4*RAB4*C2*ABX2*HM1+3*ABX2*HM1)
                    . *PI)/(2*       RAB5*C3)
0246                 IF(LA(3)) GO TO 321
0247                 R3= (-PI*(K*(PHIPC*RTP1*EXP1*((2*ABX2*C+1)*RAB2-4*RAB4*C-3*
                    . ABX2)*EXP22-RTP1*EXP1*PHIMC*((2*ABX2*C+1)*RAB2-4*RAB4*C-3*
                    . ABX2)-4*EXP2*RTC*(3*RAB*ABX2-RAB3)*EXP5M)+PHIPC*RTP1*EXP1*(3*
                    . RAB*K2*ABX2-(2*AEX2*C+1)*K2*RAB3-4*C2*RAB5-K3*RAB2*ABX2)*
                    . EXP22+RTP1*EXP1*PHIMC*(3*RAB*K2*ABX2-((2*ABX2*C+1)*K2*RAB3+4*
                    . C2*RAB5-K3*RAB2*ABX2))))/(16*EXP2*RTC*C4*RAB5)
0248           321   IF(LA(4)) GO TO 322
```

```
0249                  R4= (PI*(2*K*RTC*RTPI*EXP1*(3*RAB*(PHIPC*EXP22+PHIMC)*ABX2*C-
                     . PHIPC*(2*(RAB3*ABX2+3*RAB5)*C2+RAB3*C))+3*RAB*(PHIPC*RTC*RTPI*EXP1*K3*EXP22+
                     . (RTC*RTPI*EXP1*PHIMC*K3-4*EXP2*(2*C2+K2*C)*EXP5M))*ABX2-PHIPC
                     . *RTC*RTPI*EXP1*(4*(K2*RAB4*ABX2-RAB2*ABX2+2*RAB4)*C2-((2*ABX2
                     . *C+1)*RAB2-6*RAB4*C-3*ABX2)*K2+(4*ABX2*C+1)*K3*RAB3+8*C3*RAB6
                     . +K4*RAB2*ABX2-2*RAB2*C+6*ABX2*C)*EXP22+(RTC*RTPI*EXP1*PHIMC*(
                     . 3*(K2*RAB4*ABX2-RAB2*ABX2+2*RAB4)*C2-((2*ABX2*C+1)*RAB2-6*
                     . RAB4*C-3*ABX2)*K2-(4*ABX2*C+1)*K3*RAB3+8*C3*RAB6+K4*RAB2*ABX2
                     . -2*RAB2*C+6*ABX2*C)+4*EXP2*(2*(K2*ABX2+1)*C2*RAB3+4*C3*RAB5+
                     . K2*RAB3*C)*EXP5M)))/(32*EXP2*C6*RAB5)
0250            322   IF(LA(5)) GO TO 323
0251                  R5= (PI*(2*K*(36*RAB*EXP2*EXP5M*C2*ABX2-EXP1*RTC*RTPI*(12*C2*
                     . RAB2*ABX2-18*C2*RAB4-12*C3*RAB4*ABX2-16*C3*RAB6+3*RAB2*C-9*
                     . ABX2*C)*(PHIPC*EXP22-PHIMC)-12*EXP2*EXP5M*(C2*RAB3+2*C3*(RAB3
                     . *ABX2+RAB5)))-3*RAB*ABX2*(EXP1*RTC*RTPI*(PHIPC*EXP22+PHIMC)*(
                     . 6*K2*C+K4)-4*EXP2*EXP5M*K3*C)+EXP1*RTC*RTPI*(PHIPC*EXP22*(12*
                     . C2*(2*K2*(RAB3*ABX2+RAB5)+K3*RAB4*ABX2)+8*C3*RAB5*(K2*ABX2+5)
                     . +16*C4*RAB7+6*K2*RAB3*C-K3*(RAB2-8*RAB4*C-3*ABX2)+K4*RAB3*(6*
                     . ABX2*C+1)+K5*RAB2*ABX2)+PHIMC*(12*C2*(2*K2*(RAB3*ABX2+RAB5)-
                     . K3*RAB4*ABX2)+8*C3*RAB5*(K2*ABX2+5)+16*C4*RAB7+6*K2*RAB3*C+K3
                     . *(RAB2-8*RAB4*C-3*ABX2)+K4*RAB3*(6*ABX2*C+1)-K5*RAB2*ABX2))-4
                     . *EXP2*EXP5M*K3*RAB3*(4*C2*ABX2+C)))/(64*EXP2*C7*RAB5)
0252            323   IF(LA(6)) GO TO 500
0253                  R6= (PI*(K*RTPI*EXP1*(3*RAB*ABX2-RAB3)*(PHIPC*EXP22+PHIMC)-
                     . PHIPC*RTPI*EXP1*(K2*RAB2*ABX2-RAB2+2*RAB4*C+3*ABX2)*EXP22+
                     . RTPI*EXP1*PHIMC*(K2*RAB2*ABX2-RAB2+2*RAB4*C+3*ABX2)-4*EXP2*
                     . RTC*((2*ABX2*C-1)*RAB3+3*RAB*ABX2)*EXP5M))/(8*EXP2*RTC*C3*
                     . RAB5)
0254                  GO TO 500
0255            350   IF(.NOT.LA(1)) R1=4*PI/3*HM1
0256                  IF(.NOT.LA(2)) R2=4*PI/3*HM21
                C DEF
0257                  IF(LA(3)) GO TO 351
0258                  R3= (-PI*(20*K*RTC*C-RTPI*EXP1*PHI0C*(12*C2+12*K2*C+K4)+2*RTC*
                     . K3))/(24*RTC*C4)
0259            351   IF(LA(4)) GO TO 352
0260                  R4= (-PI*(60*K*RTC*RTPI*EXP1*PHI0C*C2+(RTC*RTPI*EXP1*PHI0C*(20
                     . *K3*C+K5)-36*C2*K2-64*C3-2*K4*C)))/(48*C6)
0261            352   IF(LA(5)) GO TO 353
0262                  R5= (-PI*(264*K*C3-RTC*EXP1*RTPI*PHI0C*(180*C2*K2+120*C3+30*K4
                     . *C+K6)+2*(28*C2*K3+K5*C)))/(96*C7)
0263            353   IF(LA(6)) GO TO 500
0264                  R6= (-PI*(6*K*RTPI*EXP1*PHI0C*C+RTPI*K3*EXP1*PHI0C-2*RTC*(K2+4
                     . *C)))/(12*RTC*C3)
0265            500   RINT=R1
0266                  RINTG(2)=R2
0267                  RINTG(3)=R3
0268                  RINTG(4)=R4
0269                  RINTG(5)=R5
0270                  RINTG(6)=R6
0271            616   FORMAT(1H ,2X,1HI,4X,2HRI/6(1H ,3X,E20.12/),1H ,5HRINT=,E20.12)
0272                  RETURN
0273                  END
```

```
0001                REAL FUNCTION GS1*8(R)
0002                IMPLICIT REAL*8(A-H,O-Z)
0003                COMMON/INT/GAM,DK,RAB,DKG,ITYPE
0004                COMMON/PUT/PUTCK(15),IPCI
0005                DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNO/
0006                IF(R.LT.1.D-10) GO TO 100
0007                R2=R*R
0008                R3=R2*R
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C       GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0009                97  CUT=DK*R
0010                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*.5D0))
0011                    CUT=CUT*CUT
0012                    PUTCK(10)=PEACH
0013                    GO TO 99
0014                98  CUT=DK**6*R3**2
0015                    PUTCK(10)=DALGAR
0016                    IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0017                99  IF(RAB.LT.1.D-10) GO TO 110
0018                    GS1=CUT*(DEXP(-GAM*(R-RAB)**2)
                       U -DEXP(-GAM*(R+RAB)**2))/(R3*2.)
0019                    RETURN
0020                110 IF(ITYPE.NE.1) GO TO 120
0021                    GS1=CUT*DEXP(-GAM*R2)*2.*GAM/R2
0022                    RETURN
0023                100 GS1=0.0D0
0024                    RETURN
0025                120 GS1=CUT*DEXP(-GAM*R2)
                C CHECK THIS EQUN
0026                    RETURN
0027                    END
```

```
0001                  REAL FUNCTION GS2*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/GAM,DK,RAB,DKQ,ITYPE
0004                  COMMON/POT/POTCK(15),IPOT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNO/
0006                  IF(R.LT.1.D-10) GO TO 100
0007                  R2=R*R
0008                  R4=R2*R2
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C     GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0009                97  CUT=DKQ*R
0010                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*(.5D0+CUT/6.D0)))
0011                    CUT=CUT*CUT
0012                    POTCK(11)=PEACH
0013                    GO TO 99
0014                98  CUT=DK**8*R4**2
0015                    POTCK(11)=DALGAR
0016                    IF(CUT.GT.1.D-10) CUT=1.DO-DEXP(-CUT)
0017                99  IF(RAB.LT.1.D-10) GO TO 110
0018                    GS2=CUT*(DEXP(-GAM*(R-RAB)**2)-
                       U DEXP(-GAM*(R+RAB)**2))/(R4*R*2.)
0019                    RETURN
0020               110  IF(ITYPE.NE.1) GO TO 120
0021                    GS2=CUT*DEXP(-GAM*R2)*2.*GAM/R4
0022                    RETURN
0023               100  GS2=0.000
0024                    RETURN
0025               120  GS2=CUT/R2*DEXP(-GAM*R2)
                C CHECK THIS EQUN
0026                    RETURN
0027                    END
```

```
0001                  REAL FUNCTION GS3*8(R)
0002                  IMPLICIT REAL*8(A-H,C-Z)
0003                  COMMON/INT/GAM,DK,RAB,DKQ,ITYPE
0004                  COMMON/PCT/PCTCK(15),IPCT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNC/
0006                  IF(R.LT.1.D-10.OR.RAB.LT.1.D-10) GO TO 100
0007                  R2=R*R
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C     GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0008                97  CUT=DK*R
0009                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*.5D0))
0010                    CUT=CUT*CUT
0011                    PCTCK(12)=PEACH
0012                    GO TO 99
0013                98  CUT=DK**6*R2**3
0014                    PCTCK(12)=DALGAR
0015                    IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0016                99  GS3=CUT*(DEXP(-GAM*(R-RAB)**2)
                       U +DEXP(-GAM*(R+RAB)**2))/(R2*2.)
0017                    RETURN
0018               100  GS3=0.0D0
0019                    RETURN
0020                    END
```

```
0001                  REAL FUNCTION GS4*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/GAM,DK,RAE,DKG,ITYPE
0004                  COMMON/POT/POTCK(15),IPOT
0005                  DATA PEACH,CALGAR/ 5HPEACH,8HCALGARNO/
0006                  IF(R.LT.1.D-10.OR.RAB.LT.1.D-10) GO TO 100
0007                  R2=R*R
0008                  R4=R2*R2
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C     GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0009                97  CUT=DKG*R
0010                    CUT=1.DO-DEXP(-CUT)*(1.DO+CUT*(1.DO+CUT*(.5DO+CUT/6.DO)))
0011                    CUT=CUT*CUT
0012                    POTCK(13)=PEACH
0013                    GO TO 99
0014                98  CUT=DK**8*R4**2
0015                    POTCK(13)=CALGAR
0016                    IF(CUT.GT.1.D-10) CUT=1.DO-DEXP(-CUT)
0017                99  GS4=CUT*(DEXP(-GAM*(R-RAB)**2)
                   U +DEXP(-GAM*(R+RAE)**2))/(R4*2.)
0018                    RETURN
0019               100  GS4=0.0D0
0020                    RETURN
0021                    END
```

```
0001                  REAL FUNCTION GSS*8(R)
0002                  IMPLICIT REAL*8(A-H,C-Z)
0003                  COMMON/INT/GAM,DK,RAB,DKQ,ITYPE
0004                  COMMON/PCT/PCTCK(15),IPCT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNC/
0006                  IF(R.LT.1.D-10) GO TO 100
0007                  R2=R*R
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C       GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0008                97  CUT=DK*R
0009                    CUT=1.D0-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*.5D0))
0010                    CUT=CUT*CUT
0011                    PCTCK(14)=PEACH
0012                    GO TO 99
0013                98  CUT=DK**6*R2**3
0014                    PCTCK(14)=DALGAR
0015                    IF(CUT.GT.1.D-10) CUT=1.D0-DEXP(-CUT)
0016                99  IF(RAB.LT.1.D-10) GO TO 110
0017                    GSS=CUT*(DEXP(-GAM*(R-RAB)**2)
                       U -DEXP(-GAM*(R+RAB)**2))/(R*2.)
0018                    RETURN
0019               110  GSS=CUT*DEXP(-GAM*R2)*R2
0020                    RETURN
0021               100  GSS=0.000
0022                    RETURN
0023                    END
```

```
0001                  REAL FUNCTION GS6*8(R)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  COMMON/INT/GAM,DK,RAB,DKQ,ITYPE
0004                  COMMON/POT/POTCK(15), IPCT
0005                  DATA PEACH,DALGAR/ 5HPEACH,8HDALGARNO/
0006                  IF(R.LT.1.D-10) GO TO 100
0007                  R2=R*R
0008                  R3=R2*R
                C*** TAKE OUT GO TO 98 FOR PEACH'S CUTOFF
                C     GO TO 98
                C*** PEACH'S CUTOFF FUNCTION  ***
0009                97  CUT=DKQ*R
0010                    CUT=1.DO-DEXP(-CUT)*(1.D0+CUT*(1.D0+CUT*(.5D0+CUT/6.D0)))
0011                    CUT=CUT*CUT
0012                    POTCK(15)=PEACH
0013                    GO TO 99
0014                98  CUT=DK**8*R2**4
0015                    POTCK(15)=DALGAR
0016                    IF(CUT.GT.1.D-10) CUT=1.DO-DEXP(-CUT)
0017                99  IF(RAB.LT.1.D-10) GO TO 110
0018                    GS6=CUT*(DEXP(-GAM*(R-RAB)**2)
                       U -DEXP(-GAM*(R+RAB)**2))/(R3*2.)
0019                    RETURN
0020               110  GS6=CUT*DEXP(-GAM*R2)
0021                    RETURN
0022               100  GS6=0.0D0
0023                    RETURN
0024                    END
```

APPENDIX D

```
0001                  IMPLICIT REAL*8(A-H,O-Z)
0002                  LOGICAL RESTRT ,LAA
0003                  DIMENSION F(10),X(10),W(470)
0004                  COMMON/FCN/INOUT,K,TIMLIM,DELM(10),LAA(6)
0005                  COMMON/FIT/IS,IC,DD,FMIN,TINC,MAXFUN,IPRINT,DSTEP,
                     .DMAX,ACC,W,RESTRT,MAXW
0006                  EXTERNAL EPFIT
0007                  READ(5,600)K,TIMLIM
0008                  CALL STIME
0009            600   FORMAT(I5,E10.5)
0010                  READ(5,601)NVEXP,NVAR,MAXFUN,IPRINT,DSTEP,DMAX,ACC
0011            601   FORMAT(4I5,3E15.6)
0012                  INOUT=1
0013                  IF(K.LT.0) WRITE(6,620)
0014                  IF(K.GE.0) WRITE(6,621)
0015            620   FORMAT(1H1,'FITTING PROGRAM  -- INITIAL RUN'/)
0016            621   FORMAT(1H1,'FITTING PROGRAM -- RESTART RUN')
0017                  WRITE(6,622) TIMLIM,NVEXP,NVAR,MAXFUN,IPRINT,DSTEP,DMAX,ACC
0018            622   FORMAT(1H ,'TIMLIM=',F10.2/1H ,'NVEXP=',I5,5X,'NVAR=',I5/
                     . 1H ,'MAXFUN=',I5,5X,'IPRINT=',I5/
                     . 1H ,'DSTEP=',E15.6,5X,'DMAX=',E15.6,5X,'ACC=',E15.6)
0019                  RESTRT=K.GE.0
0020                  IF(.NOT.RESTRT) GO TO 200
0021                  READ(5,610) IS,IC,MAXW,DD,FMIN,TINC,(F(I),I=1,NVEXP)
0022            610   FORMAT(3I5,3E15.6/(4E20.14))
0023                  READ(5,611)(W(I),I=1,MAXW)
0024            611   FORMAT(4E20.14)
0025            200   CALL VA05AD(EPFIT,NVEXP,NVAR,F,X,DSTEP,DMAX,ACC,MAXFUN,
                     . IPRINT,W)
0026                  INOUT=-1
0027                  CALL EPFIT(NVEXP,NVAR,F,X)
0028                  STOP
0029                  END
```

```
0001                  BLOCK DATA
0002                  IMPLICIT REAL*8(A-H,C-Z)
0003                  REAL*4 TIMEL
0004                  COMMON/TEST/EEA,EEO,TIMEL(8)
0005                  DATA EEA,EEO/0.0D0,0.0D0/
0006                  END
```

(20c)

```
0001                  SUBROUTINE EFFIT(NVEXP,NVAR,DIFF,PAR)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  REAL*4 TIME1,TIME2,TIMEL(8),TIMER
0004                  LOGICAL RESTRT,RECALC(10),FCNB,LA(6),CLOSED
0005                  LOGICAL NRC,LAR,LAA
0006                  INTEGER*2 ISNX
0007                  DIMENSION VEFF(10),DIFF(10),PAR(1),W(470)
0008                  COMMON/QMAT/Q(1275,8),H(1275),HO(1275)
0009                  COMMON/POT/POTCK(15),IPOT
0010                  COMMON/FCN/INCUT,K,TINLIM,DELM(10),LAA(6)
0011                  COMMON/FIT/IS,IC,DD,FMIN,TINC,MAXFUN,IPRINT,DSTEP,DMAX,ACC,
                     . W,RESTRT,MAXW
0012                  COMMON/SCLR/EXI,EXJ,ITYPE,JTYPE,ICNTR,JCNTR,R,CUT,CUT2,CUT3,
                     .ALPHAC,ALPHOB,AP(4),LA,RINTG(6),LAR
0013                  COMMON/INCUT/VNHF(10),VABN(10),VND(10),VNC(10),VNQD(10),VNQQ(10),
                     . VNM1(10),VNM2(10),VNM3(10),VNM4(10),VEQQ(10),VECC(10),VEM(6,10),
                     . NFIRST(50),NLAST(50),NCNTR(50),NTYPE(50),EX(100),CO(100),
                     . PARMC(10),PARM(10),PARMO(10),ZA,ZB,NVRST,NVXP,IVAR(10)
0014                  COMMON/IN/ITITLE(18),NBF,NORBA,NORBB,NSYM,DA(1275),DBA(1275),
                     .DBAO(1275),
                     . ALPHAO,BETA1,FCNB(50),RVEXP(10), VEXP(10),ISNX(50),CLOSED,IPERT
0015                  COMMON/TEST/EBA,EEO,TIMEL
0016                  DATA IOUT,IIN/8,9/
0017                  DATA PI2/1.57079632679489/
0018                  XI2(X)=1.D0-DEXP(-X)*(1.D0+X*(1.D0+.5D0*X))
0019                  XI3(X)=1.D0-DEXP(-X)*(1.D0+X*(1.D0+X*(.5D0+X/6.D0)))
                C     THIS PROGRAM CALCULATES A SUM OF DIFFERENCES BETWEEN A SET OF
                C     "EXPERIMENTAL" POTENTIAL POINTS AND THE VALUE CALCULATED USING
                C     FIRST ORDER PERTURBATION THEORY FOR THE EFFECTIVE POTENTIAL.
                C     IT IS DESIGNED TO BE CALLED BY TRUDGE AND TO BE USED TO FIT
                C     THE POTENTIAL PARAMETERS
                C
                C     INOUT .LT. 0 OUTPUT FOR RESTART
                C           .EQ.0 ASSUMES ALL NEEDED INPUT PREVIOUSLY READ IN
                C           .GT.0 READ INPUT AND CALCULATE FIRST VALUE
                C     RESTART .TRUE. RESTARTS CALCULATION USING PARAMETER VALUES
                C                    AND INTEGRALS FROM PREVIOUS RUN
                C     IOUT, IIN  ARE FILES FOR RESTART OUTPUT FROM EFFPOT ONLY AND
                C                  INPUT, CURRENTLY 8 AND 9
                C    FILES REQUIRED
                C     8 AND 9 (SEE ABOVE)
                C     1 AND 2
                C      SUBROUTINE POTHF (EXECUTED IF NOT RESTART) REQUIRES PLYATOM IJLK FILES FO
                C      FILES FOR AB AND FOR AE WITH A HAVING ZERO NUCLEAR CHARGE FOR
                C      ALL R VALUES TO BE FIT
                C      FILES FOR AE MUST BE CONCATTENATED ON TAPE1 AND THOSE FOR
                C      AB A(0) ON TAPE2
                C     10 AND 11
                C      WHEN CALCULATING Q INTEGRALS, INTERMEDIATE RESULTS ARE
                C      WRITTEN ON FILES 10 AND 11
0020                  IPOT=1
0021                  NMAX=10
0022                  NVXP=NVEXP
0023                  IF(INCUT.GE.0) GO TO 100
                C   OUTPUT VALUES FOR RESTART
0024                99 CALL OUTR(IOUT,NVAR,NMAX,DIFF,PAR)
0025                   STOP
0026               100 NVRST=1
0027                   IF(INCUT.EQ.0) GO TO 200
```

```
C    READ INPUT AND SET UP C#, CE
0028          CALL SETUP(NVAR,NMAX,PAR)
0029          IF(.NOT.RESTRT) GO TO 170
C    READ IN RESTART PARAM
0030          CALL INR(IIN,IOUT,NVAR,NMAX,PAR)
C    CALCULATE NON-PARAM. DEPENDENT TERMS
0031          IF(NVRST.GT.NVEXP) GO TO 171
0032     170  CALL FOTHF(IOUT)
0033     171  IF(.NOT.RESTRT) GO TO 204
C RESET PARAMETERS AND CHECK TO SEE WHICH TERMS HAVE TO BE RECALCULATED
0034     200  DO 190 I=1,NMAX
0035          IV=IVAR(I)
0036          IF(IV.EQ.0) GO TO 190
0037          PARM(I)=PARM0(I)+DATAN(PAR(IV))*DELM(I)
0038     190  CONTINUE
0039          ALPHAC=PARM(10)
0040          ALPHAC=PARM(9)
0041          BETA1=PARM(8)
0042          DO 201 I=1,4
0043     201  AP(I)=PARM(8-I)
0044          CUT2=PARM(3)
0045          CUT3=PARM(2)
0046          CUT=PARM(1)
0047     204  ALPHAD=ALPHAQ-6.00*BETA1
0048          DO 202 I=1,NMAX
0049          RECALC(I)=.FALSE.
0050     202  IF(DABS(PARM0(I)-PARM(I)).GT.1.D-10) RECALC(I)=.TRUE.
0051          WRITE(6,661)ALPHAD,ALPHAQ,BETA1,CUT,CUT2,CUT3,AP
0052     661  FORMAT(1H0,'EFFPOT EVALUATED WITH'/
              . 1H ,'ALPHAD=',E15.6,5X,'ALPHAC=',E15.6,5X,'BETA1=',E15.6/
              . 1H ,'CUT=',E15.6,5X,'CUT2=',E15.6,5X,'CUT3=',E15.6/
              . 1H ,'AP:',4E15.6)
0053          CALL TTIME(TIME1)
0054          TIMER=TIMLIM-TIME1
0055          IF(TIMER.GE.TIMEL(1)) GO TO 320
0056          IF(RECALC(1)) GO TO 301
0057          IF(RECALC(2)) GO TO 304
0058          IF(RECALC(3)) GO TO 306
0059          IF(TIMER.GE.TIMEL(8)) GO TO 320
0060          GO TO 319
0061     301  IF(RECALC(2)) GO TO 302
0062          IF(RECALC(3)) GO TO 303
0063          IF(TIMER.GE.TIMEL(2).AND.TIMEL(2).GT.0.0) GO TO 320
0064          GO TO 319
0065     302  IF(RECALC(3)) GO TO 319
0066          IF(TIMER.GE.TIMEL(6).AND.TIMEL(6).GT.0.0) GO TO 320
0067          GO TO 319
0068     303  IF(TIMER.GE.TIMEL(5).AND.TIMEL(5).GT.0.0) GO TO 320
0069          GO TO 319
0070     304  IF(RECALC(3)) GO TO 305
0071          IF(TIMER.GE.TIMEL(4).AND.TIMEL(4).GT.0.0) GO TO 320
0072          GO TO 319
0073     305  IF(TIMER.GE.TIMEL(7).AND.TIMEL(7).GT.0.0) GO TO 320
0074          GO TO 319
0075     306  IF(TIMER.GE.TIMEL(3).AND.TIMEL(3).GT.0.0) GO TO 320
0076     319  WRITE(6,670) TIMER
0077     670  FORMAT(1H0,'*****  TIME LIMIT: SET UP RESTART *********'/
              . 1H ,F10.2)
0078          K=0
```

```
0079                GO TO 99
                C RESET PARMO
0080           320  DO 203 I=1,10
0081           203  PARMO(I)=PARM(I)
                C    CALCULATE NUCLEAR TERMS
0082                DO 220 N=1,NVEXP
0083                NRC=N.LT.NVRST .OR.INCUT.EQ.0
0084                R=RVEXP(N)
0085                IF(.NOT.RECALC(3).AND.NRC) GO TO 205
0086                VNQD(N)=XI2(CUT2*R)/R**2
0087                VND(N)=VNQD(N)*VNQD(N)/2.D0
0089           205  IF(.NOT.RECALC(2).AND.NRC) GO TO 206
0099                VNQQ(N)=XI3(CUT3*R)/R**3
0090                VNQ(N)=VNQQ(N)*VNQQ(N)/2.D0
0091           206  IF(.NOT.RECALC(1).AND.NRC) GOTO 220
0092                EXPR=DEXP(-CUT*R)
0093                VNM1(N)=EXPR
0094                VNM2(N)=R*VNM1(N)
0095                VNM3(N)=R*VNM2(N)
0096                VNM4(N)=EXPR/R
0097           220  CONTINUE
                C *** DEBUG                                                    **********
0098                WRITE(6,660)(RVEXP(I),VND(I),VNQD(I),VNQ(I),VNQQ(I),VNM1(I),
                   . VNM2(I),VNM3(I),VNM4(I),I=1,NVEXP)
0099           660  FORMAT(1H0,2X,1HR,7X,3HVND,12X,4HVNQD,11X,3HVNQ,12X,4HVNQQ,
                   . 11X,4HVNM1,11X,4HVNM2,11X,4HVNM3,11X,4HVNM4/(1H ,F8.2,8E15.6))
                C *** DEBUG                                                    **********
                C    CALCULATE Q TERMS
0100                CALL FCTQ(RECALC)
                C    SET UP RINT CALCULATION
0101                LAA(1)=.NOT.RECALC(3)
0102                LAA(2)=.NOT.RECALC(2)
0103                LAA(3)=.NOT.RECALC(1)
0104                DO 230 I=4,6
0105           230  LAA(I)=LAA(3)
0106                LAR=.TRUE.
0107                DO 235 I=1,6
0108                IF(.NOT.LAA(I)) GO TO 240
0109           235  CONTINUE
0110                LAR=.FALSE.
0111                IF(NVRST.LE.NVEXP) GO TO 240
0112                GO TO 245
                C    CALC R INTEGRALS
0113           240  CALL FOTR
                C    CALC EFFECTIVE POTENTIALS AND THE SUM OF DIFFERENCES
0114           245  VPOT=0.000
0115                INOUT=0
0116                ZB2=ZB*ZB
0117                DO 300 N=1,NVEXP
0118                VNDA=-ZB2*ALPHAD*VND(N)
0119                VNQA=-ZB2*ALPHAQ*VNQ(N)
0120                VNMA1=ZB2*AP(1)*VNM1(N)
0121                VNMA2=ZB2*AP(2)*VNM1(N)
0122                VNMA3=ZB2*AP(3)*VNM3(N)
0123                VNMA4=ZB2*AP(4)*VNM4(N)
0124                VNQDA=ZB*ALPHAD*VNQD(N)
0125                VNQQA=ZB*ALPHAQ*VNQQ(N)
0126                VEQDA=-ALPHAD*VEQD(N)
0127                VEMA1=-ALPHAD*VEM(1,N)/2.D0
```

```
0128                VEMA2=-ALPHA3*VEM(2,N)/2.00
0129                VEQQA=-ALPHAC*VEQC(N)
0130                VEMA3=AP(1)*VEM(3,N)
0131                VEMA4=AP(2)*VEM(4,N)
0132                VEMA5=AP(3)*VEM(5,N)
0133                VEMA6=AP(4)*VEM(6,N)
0134                VEFF(N)=VNDA+VNQA+VNMA1+VNMA2+VNMA3+VNMA4+VNQDA+VNQQA+
                   . VEQDA+VEQQA+VNHF(N)+VEMA1+VEMA2+VEMA3+VEMA4+VEMA5+VEMA6
0135                VEFF(N)=EBA+VEFF(N)-EPO
0136                IF(IPERT.EQ.0) VEFF(N)=VEFF(N)+VABN(N)
0137                DIFFN=(VEFF(N)-VEXP(N))/VEXP(N)
0138                VPOT=VPOT+DIFFN*DIFFN
0139                WRITE(6,665)N ,VNDA,VNQA,VNMA1,VNMA2,VNMA3,VNMA4,VNQDA,VNQQA,
                   . VEQDA,VEQQA,VEMA1,VEMA2,VEMA3,VEMA4,VEMA5,VEMA6
0140          665   FORMAT(1H ,'N=',I5 ,5X ,'VNDA=',E15.6,5X,'VNQA=',E15.6/
                   . 1H ,'VNMA1=',E15.6,5X,'VNMA2=',E15.6,5X,'VNMA3=',E15.6,5X,
                   . 'VNMA4=',E15.6/
                   . 1H ,'VNQDA=' ,E15.6,5X,'VNQQA=',E15.6,5X,'VEQDA=',E15.6,5X,
                   . 'VEQQA=',E15.6/
                   . 1H ,'VEMA1=',E15.6,5X,'VEMA2=' ,E15.6/
                   . 1H ,'VEMA3=',E15.6,5X,'VEMA4=',E15.6,5X,'VEMA5=',E15.6,5X,
                   . 'VEMA6=',E15.6/)
0141          300   DIFF(N)=DIFFN
0142                WRITE(6 ,600)  VPOT,(RVEXP(I) ,VEXP(I),VEFF(I),DIFF(I),I=1,NVEXP)
0143          600   FORMAT(1H0,'TOTAL DEVIATION (EFFPOT):',D15.6/
                   . 1H0,4X,1HR,10X,15HEXACT POTENTIAL, 2X,14HCALC POTENTIAL,
                   . 5X,10HDIFFERENCE/(1H ,4E15.6))
0144                CALL TTIME(TIME2)
0145                TIMER=TIME2-TIME1
0146                TIMEL(1)=AMAX1(TIMEL(1),TIMER)
0147                IF(RECALC(1)) GO TO 321
0148                IF(RECALC(2)) GO TO 325
0149                IF(RECALC(3)) GO TO 327
0150                TIMEL(8)=AMAX1(TIMEL(8),TIMER)
0151                GO TO 329
0152          321   IF(RECALC(2)) GO TO 322
0153                IF(RECALC(3)) GO TO 324
0154                TIMEL(2)=AMAX1(TIMEL(2),TIMER)
0155                GO TO 329
0156          322   IF(RECALC(3)) GO TO 323
0157                TIMEL(6)=AMAX1(TIMEL(6),TIMER)
0158                GO TO 329
0159          323   TIM2=TIMER/2.
0160                IF(TIMEL(2).EQ.0.0) TIMEL(2)=TIM2
0161                IF(TIMEL(3).EQ.0.0) TIMEL(3)=TIM2
0162                IF(TIMEL(4).EQ.0.0) TIMEL(4)=TIM2
0163                TIM2=TIMER*.75
0164                IF(TIMEL(5).EQ.0.0) TIMEL(5)=TIM2
0165                IF(TIMEL(6).EQ.0.0) TIMEL(6)=TIM2
0166                IF(TIMEL(7).EQ.0.0) TIMEL(7)=TIM2
0167                GO TO 329
0168          324   TIMEL(5)=AMAX1(TIMEL(5),TIMER)
0169                GO TO 329
0170          325   IF(RECALC(3)) GO TO 326
0171                TIMEL(4)=AMAX1(TIMEL(4),TIMER)
0172                GO TO 329
0173          326   TIMEL(7)=AMAX1(TIMEL(7),TIMER)
0174                GO TO 329
0175          327   TIMEL(3)=AMAX1(TIMEL(3),TIMER)
```

```
0176          329   IF(IPERT.EQ.0) RETURN
0177                EBAT=VEFF(1)+EB0
0178                WRITE(6,601) EBAT
0179          601   FORMAT(1H ,'TOTAL ENERGY FOR B=',E20.12)
0180                RETURN
0181                END
```

```
0001                  SUBROUTINE OUTR(ICUT,NVAR,NMAX,F,PAR)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  LOGICAL RESTRT,FCNB,CLCSED
0004                  INTEGER*2 ISNX
0005                  DIMENSION F(1),PAR(1)
0006                  COMMON/INCUT/VNHF(10),VABN(10),VND(10),VNQ(10),VNQD(10),VNQQ(10),
                     . VNM1(10),VNM2(10),VNM3(10),VNM4(10),VEQD(10),VEQQ(10),VEM(6,10),
                     . NFIRST(50),NLAST(50),NCNTR(50),NTYPE(50),EX(100),CO(100),
                     . PARMO(10),PARM(10),PARMO(10),ZA,ZB,NVRST,NVEXP,IVAR(10)
0007                  COMMON/FCN/ INOUT,K,TIMLIM,DELM(10)
0008                  COMMON/FIT/IS,IC,CO,FMIN,TINC,MAXFUN,IPRINT,CSTEP,DMAX,ACC,
                     . W(470),RESTRT,MAXW
0009                  COMMON/IN/ITITLE(18),NBF,NORBA,NORBB,NSYM,DA(1275),DBA(1275),
                     . DBAO(1275),
                     . ALPHAO,BETA1,FCNB(50),RVEXP(10),VEXP(10),ISNX(50),CLCSED,IPERT
0010                  COMMON/TEST/EBA,EBO
0011                  WRITE(7,699)K,TIMLIM
0012            699   FORMAT(I5,F10.5)
0013                  WRITE(7,673)NVEXP,NVAR,MAXFUN,IPRINT,CSTEP,DMAX,ACC
0014            673   FORMAT(4I5,3E15.6)
0015                  WRITE(7,670)IS,IC,MAXW,CO,FMIN,TINC,(F(I),I=1,NVEXP)
0016            670   FORMAT(3I5,3E15.6/(4E20.14))
0017                  WRITE(7,671)(W(I),I=1,MAXW)
0018            671   FORMAT(4E20.14)
0019                  WRITE(IOUT,600)IVAR,(PARM(I),PARMO(I),I=1,NMAX),
                     .  (PAR(I),I=1,NVAR)
0020            600   FORMAT(1015/(2E20.12))
0021                  WRITE(6,610)NVEXP,NVAR,IVAR,(PARM(I),PARMO(I),I=1,NMAX)
0022            610   FORMAT(1H0,'***** OUTPUT VALUES FOR RESTART *****'/
                     . 1H ,6HNVEXP=,I5,5X,5HNVAR=,I5/
                     . 1H ,5HIVAR:,10I5/
                     . 1H ,4X,4HPARM,16X,5HPARMO/(1H ,2E20.12))
0023                  WRITE(6,609)(PAR(I),I=1,NVAR)
0024            609   FORMAT(1H ,4X,3HPAR/(1H ,E20.12))
0025                  WRITE(IOUT,601) EBA,EBO
0026                  WRITE(IOUT,600) NVEXP
0027                  WRITE(IOUT,601)(RVEXP(I),I=1,NVEXP)
0028                  WRITE(IOUT,601)(VNHF(I),I=1,NVEXP)
0029                  WRITE(IOUT,601)(VABN(I),I=1,NVEXP)
0030                  WRITE(IOUT,601)(VND (I),I=1,NVEXP)
0031                  WRITE(IOUT,601)( VNQ(I),I=1,NVEXP)
0032                  WRITE(IOUT,601)(VNQD(I),I=1,NVEXP)
0033                  WRITE(IOUT,601)(VNQQ(I),I=1,NVEXP)
0034                  WRITE(IOUT,601)(VNM1(I),I=1,NVEXP)
0035                  WRITE(IOUT,601)(VNM2(I),I=1,NVEXP)
0036                  WRITE(IOUT,601)(VNM3(I),I=1,NVEXP)
0037                  WRITE(IOUT,601)(VNM4(I),I=1,NVEXP)
0038                  WRITE(IOUT,601)(VEQD(I),I=1,NVEXP)
0039                  WRITE(IOUT,601)(VEQQ(I),I=1,NVEXP)
0040                  WRITE(IOUT,601)((VEM(I,J),J=1,NVEXP),I=1,6)
0041            601   FORMAT(4E20.12)
0042                  WRITE(6,611)
0043            611   FORMAT(1H0,'***** RESTART OUTPUT COMPLETED ***** ')
0044                  RETURN
0045                  END
```

```
0001                  SUBROUTINE SETUP(NVAR,NMAX,PAR)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  REAL*4 TIMEL
0004                  INTEGER*2 ISNX
0005                  DIMENSION PM(10),PAR(1)
0006                  LOGICAL FCNB,LA,CLOSED
0007                  COMMON/FCN/INOUT,K,TIMLIM,DELM(10)
0008                  COMMON/CMAT/DE(1275),DBC(1275),F(50)
0009                  COMMON/INOUT/VNHF(10),VABN(10),VND(10),VNC(10),VNCD(10),VNQQ(10),
                     . VNM1(10),VNM2(10),VNM3(10),VNM4(10),VEQD(10),VEQQ(10),VEM(6,10),
                     . NFIRST(50),NLAST(50),NCNTR(50),NTYPE(50),EX(100),CO(100),
                     . PARMO(10),PARM(10),PARMC(10),ZA,ZB,NVRST,NVEXP,IVAR(10)
0010                  COMMON/IN/ITITLE(18),NBF,NORBA,NORBB,NSYM,DA(1275),DBA(1275),
                     .DBAO(1275),
                     . ALPHAO,BETA1,FCNB(50),RVEXP(10), VEXP(10),ISNX(50),CLOSED,IPERT
0011                  COMMON/SCLR/EXI,EXJ,ITYPE ,JTYPE,ICNTR,JCNTR,R ,CUT ,CUT2,CUT3 ,
                     . ALPHAD,ALPHOB,AP(4) ,LA(6) ,RINTG(6)
0012                  COMMON/TEST/EEA,EEO,TIMEL(8)
0013                  DATA P4/3HCUT,4HCUT3,4HCUT2,5HAP(4),5HAP(3),5HAP(2),5HAP(1),
                     . 5HBETA1,6HALPHAO,6HALPHAD/
0014                  DATA PI2/1.57079632679489/
0015                  READ(5,600) ITITLE
0016            600   FORMAT(18A4)
0017                  READ(5,601) NBF,NORBA ,NORBB ,IPERT
0018            601   FORMAT(16I5)
0019                  WRITE(6,620)ITITLE,NBF ,NORBA ,NORBB,NVEXP
0020            620   FORMAT(1H1//
                     . 5X,'PROGRAM TO OPTIMIZE EFFECTIVE POTENTIAL PARAMETERS'//
                     . 5X,18A4//
                     . 5X,'NO. OF BASIS FUNCTIONS.....' ,I5/
                     . 5X,'NO. OF ATOM A ORBITALS.....',I5/
                     . 5X,'NO. OF ATOM B ORBITALS.....',I5/
                     . 5X ,'NO. OF POTENTIAL POINTS'/
                     . 5X,'    TO BE FIT.............',I5/)
0021                  IF(IPERT.GT.1.OR.IPERT.LT.0) IPERT=0
0022                  IF(IPERT.EQ.0) WRITE(6,660)
0023                  IF(IPERT.EQ.1) WRITE(6,661)
0024            660   FORMAT(1HO,'POTENTIAL FIT USING FIRST ORDER PERTURBATION THEORY')
0025            661   FORMAT(1HO,'POTENTIAL FIT USING "SCF" ENERGY EXPRESSION'/)
                C  FORM ATOM A AND B DENSITY MATRICES FROM INPUT ORBITAL VECTORS
0026                  DO 104 I=1,8
0027            104   TIMEL(I)=0.0
0028                  NSYM=NBF*(NBF+1)/2
0029                  DO 105 I=1,NSYM
0030                  DA(I)=0.0D0
0031                  DE(I)=0.0D0
0032                  DBA(I)=0.0D0
0033                  DBAO(I)=0.0D0
0034            105   DE(I)=0.0D0
0035                  READ(5,600) ITITLE
0036                  WRITE(6,621) ITITLE
0037            621   FORMAT(1H ,'ATOM A:'/1H  ,18A4)
0038                  DO 110 K=1,NORBA
0039                  READ(5 ,602) KX,SYM,NX,E,F,(H(I),I=1,NBF)
0040            602   FORMAT(4X,I3,5X,A8,7X,I2,6X,D15.8,5X,D15.8/(5D15.8))
0041                  WRITE(6,622) KX,SYM ,NX ,E,F  ,(H(I),I=1  ,NBF)
0042            622   FORMAT(/5X,I3,1X,A8,2X,'SHELL ',I2,5X ,'DE =  ',D15.8 ,5X ,
                     . 'F  =  ' ,E15.8/(5X,5D15.8))
0043                  IJ=0
```

```
0044                  DO 106 I=1,NBF
0045                  VAL=H(I)*F
0046                  DO 107 J=1,I
0047                  IJ=IJ+1
0048            107   DA(IJ)=DA(IJ)+VAL*H(J)
0049            106   CONTINUE
0050            110   CONTINUE
0051                  READ(5 ,600)ITITLE
0052                  WRITE(6 ,623) ITITLE
0053            623   FORMAT(1H ,'UNDISTORTED ATOM B:'/1H ,18A4)
0054                  CLOSED=.TRUE.
0055                  DO 120 K=1,NORBB
0056                  READ(5,602) KX,SYM,NX,E,F,(H(I),I=1,NBF)
0057                  WRITE(6,622) KX,SYM,NX,E,F,(H(I),I=1 ,NBF)
0058                  IJ=0
0059                  IF(F.LT.1.D0) CLOSED=.FALSE.
0060                  DO 116 I=1 ,NBF
0061                  VAL=H(I)*F
0062                  DO 117 J=1,I
0063                  IJ=IJ+1
0064                  IF(.NOT.CLOSED) DBO(IJ)=DBO(IJ)+VAL*H(J)
0065            117   DB(IJ)=DB(IJ)+VAL*H(J)
0066            116   CONTINUE
0067            120   CONTINUE
0068                  IF(IPERT.EQ.1) GO TO 122
0069                  DO 121 I=1,NSYM
0070                  DBA(I)=DB(I)
0071            121   DBAO(I)=DBO(I)
0072                  WRITE(6,662)
0073            662   FORMAT(1H0,'FUNCTION FOR DISTORTED ATOM B = UNDISTORTED FUNCTION')
0074                  GO TO 123
0075            122   READ(5,600) ITITLE
0076                  WRITE(6,670)ITITLE
0077            670   FORMAT(1H ,'FUNCTION FOR DISTORTED ATOM B:',18A4)
0078                  DO 300 K=1,NORBB
0079                  READ(5,602) KX,SYM,NX,E,F,(H(I),I=1,NBF)
0080                  WRITE(6,622) KX,SYM,NX,E,F,(H(I),I=1,NBF)
0081                  IJ=0
0082                  DO 306 I=1,NBF
0083                  VAL=H(I)*F
0084                  DO 307 J=1,I
0085                  IJ=IJ+1
0086                  IF(F.LT.1.D0) DBAO(IJ)=DBAO(IJ)+VAL*H(J)
0087            307   DBA(IJ)=DBA(IJ)+VAL*H(J)
0088            306   CONTINUE
0089            300   CONTINUE
                C READ POTENTIAL PARAMETERS
0090            123   READ(5,603) ALPHAD,ALPHAQ,BETA1
0091                  READ(5,603) CUT,CUT2 ,CUT3
0092                  READ(5,603) AP
0093            603   FORMAT(4D15.8)
0094                  WRITE(6 ,624) ALPHAD,ALPHAQ,BETA1,CUT,CUT2,CUT3,AP
0095            624   FORMAT(1H0/5X,'***** EACH''S CUTOFF FUNCTIONS *****'/
                     . 5X,'DIPOLE PARAMETER.............' ,D15.8/
                     . 5X,'QUADRUPOLE PARAMETER.........',D15.8/
                     . 5X,'BETA1........................',D15.8/
                     . 5X,'CUTOFF PARAMETERS............',3D15.8/
                     . 5X,'MONOPOLE PARAMETERS..........' ,4D15.8/)
0096                  READ(5,631)DELM
```

```
0097          631  FORMAT(5E15.6)
0098               WRITE(6,632) DELM
0099          632  FORMAT(1H0,'LIMITS ON VARIATION OF PARM(I)'/
                  . (1H ,5E15.6))
0100               READ(5,626) IVAR
0101          626  FORMAT(10I5)
0102               WRITE(6,627)
0103          627  FORMAT(1H ,'PARAMETERS VARIED IN THIS RUN'/
                  . 1H ,'  ORDER',5X,'PARAMETER')
0104               PARM(10)=ALPHAD
0105               PARM(9)=ALPHAQ
0106               PARM(8)=BETA1
0107               DO 119 I=1,4
0108          119  PARM(8-I)=AP(I)
0109               PARM(3)=CUT2
0110               PARM(2)=CUT3
0111               PARM(1)=CUT
0112               NV=0
0113               DO 130 I=1,10
0114               IV=IVAR(I)
0115               IF(IV.EQ.0) GO TO 130
0116               IF(DABS(DELM(I)).LT.1.D-10) GO TO 131
0117               NV=NV+1
0118               IF(I.EQ.10) GO TO 129
0119               II=I+1
0120               DO 125 J=II,10
0121               IF(IV.EQ.IVAR(J)) GO TO 131
0122          125  CONTINUE
0123          129  WRITE(6,628) IV,PM(I)
0124          628  FORMAT(1H ,5X,I5,5X,A8)
0125          130  CONTINUE
0126               GO TO 132
0127          131  WRITE(6,630) IV,IVAR,DELM
0128          630  FORMAT(1H0,'***** ERROR IN PARAMETER VARIATION ORDER *****'/
                  . 1H ,'IV=',I5/1H ,'IVAR:',10I5/
                  . 1H ,'DELM:'/(1H ,5E15.6))
0129               STOP
0130          132  DO 118 I=1,10
0131               PAR(I)=0.0D0
0132               PARM0(I)=PARM(I)
0133               DELM(I)=DELM(I)/P12
0134          118  PARM0(I)=0.0D0
0135               IF(NV.EQ.NVAR) GO TO 140
0136               WRITE(6,640) NV,NVAR
0137          640  FORMAT(1H0,'**********  ERROR  **********'/
                  . 1H ,'INCORRECT NUMBER OF PARAMETERS VARIED'/
                  . 1H ,' NV=',I5,5X,' SHOULD = NVAR=',I5/)
0138               STOP
              C  READ POTENTIAL POINTS TO BE FIT
0139          140  READ(5,604)(RVEXP(I),VEXP(I),I=1,NVEXP)
0140          604  FORMAT(2D15.8)
0141               WRITE(6,625)(RVEXP(I),VEXP(I),I=1,NVEXP)
0142          625  FORMAT(1H0,'POTENTIAL POINTS TO BE FIT'/
                  . 15X,1HR ,16X,4HV(R)/(1X,D15.8,2X,D15.8))
              C  DETERMINE WHICH BASIS FCNS USED FOR ATOM B
0143               II=0
0144               DO 210 I=1,NBF
0145               II=II+1
0146               FCNB(I)=.TRUE.
```

```
0147          210   IF(DBA(II).EQ.0.0D0) FCNP(I)=.FALSE.
0148                RETURN
0149                END
```

```
0001                SUBROUTINE POTHF(IOUT)
0002                IMPLICIT REAL*8(A-H,O-Z)
0003                LOGICAL FCNB,CLOSED
0004                INTEGER*2 IA,IB,IC,ID,IMU,IF,ISNX
0005                COMMON/INTS/NINTS,LAST,IA(500),IB(500),IC(500),ID(500),IMU(500),
                   . IF(500),V(500),VQ(500)
0006                COMMON/INOUT/VNHF(10),VAFN(10),VND(10),VNQ(10),VNQD(10),VNQQ(10),
                   . VNM1(10),VNM2(10),VNM3(10),VNM4(10),VEQC(10),VEQQ(10),VEM(6,10),
                   . NFIRST(50),NLAST(50),NCNTR(50),NTYPE(50),EX(100),CO(100),
                   . PARMC(10),PARM(10),PARMO(10),ZA,ZB,NVRST,NVEXP,IVAR(10)
0007                COMMON/IN/ITITLE(18),NBF,NORBA,NORBB,NSYM,DA(1275),DBA(1275),
                   .DBAO(1275),
                   . ALPHAQ,BETA1,ECNE(50),EVEXE(10),VEXP(10),ISNX(50),CLOSED,IPERT
0008                COMMON/QMAT/DB(1275),DBO(1275),DA1(1275),DA2(1275),H(1275),
                   . HC(1275),DKO(1275),DKAC(1275)
0009                COMMON/TEST/EBA,EBO
0010                DATA IJLK/4HIJLK/
                C   CALCULATE HARTREE-FOCK TERMS
                C   DIVIDE DIAG DA,DB,ELEMENTS BY 2, CALC ISNX ARRAY, AND FORM
                C   DENSITY MATRICES FOR ATOM A 2J-K POTENTIAL
0011                II=0
0012                DO 129 I=1,NSYM
0013                IF(.NOT.CLOSED) DKO(I)=-DEC(I)
0014                DA1(I)=DA(I)*2.D0
0015             129  DA2(I)=-DA(I)
0016                II=0
0017                DO 128 I=1,NBF
0018                ISNX(I)=II
0019                II=II+I
0020                DA(II)=DA(II)*0.5D0
0021                IF(.NOT.CLOSED) DBO(II)=DEC(II)*0.5D0
0022                DBA(II)=DBA(II)*0.5D0
0023                IF(.NOT.CLOSED)DBAO(II)=DEAO(II)*0.5D0
0024             128  DB(II)=DB(II)*0.5D0
                C   CALCULATE ELECTRONIC HF TERMS AND < B | ZA/RA + ZB/RB | B >
0025                DO 140 N=NVRST,NVEXP
0026                VABN(N)=0.0D0
0027                VAB=0.0D0
0028                READ(1) ITITLE
0029                WRITE(6,626) ITITLE
0030             626  FORMAT(/5X,'FILE #1 TITLE = ',18A4)
0031                IF(ITITLE(18).NE.IJLK) GO TO 121
0032                IF(N.NE.1) GO TO 125
                C   READ BASIS SET INFORMATION AND OUTPUT FOR RESTART
0033                READ(1)II,(NFIRST(I),NLAST(I),NCNTR(I),NTYPE(I),I=1,II)
0034                IF(II.EQ.NBF) GO TO 122
0035             121  WRITE(6,627)
0036             627  FORMAT(//5X,'***** INPUT FILE ERROR *****')
0037                STOP
0038             122  WRITE(IOUT,600)II,(NFIRST(I),NLAST(I),NCNTR(I),NTYPE(I),I=1,II)
0039             600  FORMAT(I5/(4I5))
0040                READ(1) II,(VAL,VAL,VAL,EX(I),CC(I),I=1,II),(V(I),I=1,II)
0041                WRITE(IOUT,601)II,(EX(I),CO(I),I=1,II),(V(I),I=1,II)
0042             601  FORMAT(I5/(4E20.12))
0043                READ(1)II,X1,Y1,Z1,ZA,X2,Y2,Z2,ZB
0044                IF(II.NE.2) GO TO 121
0045                WRITE(IOUT,602) ZA,ZB
0046             602  FORMAT(2E20.12)
0047                WRITE(6,630)X1,Y1,Z1,ZA,X2,Y2,Z2,ZB
```

```
0048          630  FORMAT(1H0,10X,1HX,9X,1HY,9X,1HZ,9X,10HNUC CHARGE/
                  . 1H ,7HATOM A:,4F10.3/1H ,7HATOM B:,4F10.3//)
0049               IF(Z2.NE.RVEXP(N)) GO TO 121
0050               GO TO 130
0051          125  READ(1)II
0052               IF(II.NE.NBF) GO TO 121
0053          126  READ(1)
0054               READ(1)II,X1,Y1,Z1,ZZA,X2,Y2,Z2,ZZB
0055               IF(II.NE.2.OR.ZZA.NE.ZA.OR.ZZB.NE.ZB.OR.Z2.NE.RVEXP(N))
                  . GO TO 121
0056          130  READ(1) II
0057               READ(1) II
0058               DO 124 I=1,NSYM
0059          124  H(I)=0.0D0
0060               III=2
0061               IF(IPERT.EQ.1) III=1
0062               DO 132 I=1,III
0063          131  READ(1) NINTS,LAST
0064               IF(LAST.EQ.0) GO TO 131
0065          132  READ(1) II
0066               IF(IPERT.EQ.0) GO TO 133
0067               TB=0.0D0
0068               TBA=0.0D0
0069          333  READ(1)NINTS,LAST,IA,IB,IC,V
0070               IF(NINTS.EQ.0) GO TO 335
0071               DO 334 I=1,NINTS
0072               TB=TB+V(I)*DB(ISNX(IA(I))+IB(I))
0073          334  TBA=TBA+V(I)*DBA(ISNX(IA(I))+IB(I))
0074          335  IF(LAST.EQ.0) GO TO 333
0075               WRITE(6,620) TB,TBA
0076          620  FORMAT(1H ,'KINETIC ENERGY TERMS:'/
                  . 1H ,'TB=',E20.12/1H ,'TBA=',E20.12)
0077               READ(1)II
0078          133  READ(1) NINTS,LAST,IA,IB,IC,V
0079               IF(NINTS.EQ.0) GO TO 135
0080               DO 134 I=1,NINTS
0081          134  H(ISNX(IA(I))+IB(I))=V(I)
0082          135  IF(LAST.EQ.0) GO TO 133
              C  CALCULATE ATOM A 2J-K POTENTIAL FROM UNIT 1 2E INTS AND ADD TO H
0083               READ(1) II
0084               WRITE(6,680) II
0085          680  FORMAT(1H ,A8)
0086               CALL JKPOT(H,DA1,DA2,ISNX,1,HC,CKC,.TRUE.)
0087               IF(IPERT.EQ.0) READ(1,END=136)
0088          136  IJ=0
0089               DO 137 I=1,NBF
0090               DO 137 J=1,I
0091               IJ=IJ+1
0092          137  VAB=VAB+H(IJ)*DBA(IJ)
0093               VABN(N)=4.D0*VAB
0094          140  CONTINUE
              C *** DEBUG                                              **********
0095               WRITE(6,660)(VABN(I),I=1,NVEXP)
0096          660  FORMAT(1H ,2X,4HVABN/(1H ,E20.12))
              C *** DEBUG                                              **********
0097               IF(IPERT.EQ.0) GO TO 360
0098               REWIND 1
0099               DO 331 I=1,6
0100          331  READ(1)
```

```
0101              DO 332 I=1,3
0102          336 READ(1) NINTS,LAST
0103              IF(LAST.EQ.0) GO TO 336
0104          332 READ(1) II
0105              II=1
0106              DO 340 I=1,NSYM
0107              H(I)=0.0D0
0108              HC(I)=0.0D0
0109              DA2(I)=-DB(I)
0110              IF(I.NE.ISNX(II+1)) GO TO 340
0111              II=II+1
0112              DA2(I)=2.*DA2(I)
0113          340 DA1(I)=-2.D0*DA2(I)
0114              CALL JKPOT(H,DA1,DA2,ISNX,1,HC,DKC,CLOSED)
0115              VJKB=0.0D0
0116              IJ=0
0117              DO 337 I=1,NBF
0118              DO 337 J=1,I
0119              IJ=IJ+1
0120          337 VJKB=VJKB+H(IJ)*DB(IJ)+HO(IJ)*DBO(IJ)
0121              REWIND 1
0122              DO 341 I=1,6
0123          341 READ(1)
0124              DO 342 I=1,3
0125          343 READ(1) NINTS,LAST
0126              IF(LAST.EQ.0) GO TO 343
0127          342 READ(1) II
0128              II=1
0129              DO 350 I=1,NSYM
0130              H(I)=0.0D0
0131              HO(I)=0.0D0
0132              DA2(I)=-DBA(I)
0133              IF(.NOT.CLOSED) DKAO(I)=-DBAC(I)
0134              IF(I.NE.ISNX(II+1)) GO TO 350
0135              II=II+1
0136              IF(.NOT.CLOSED) DKAO(I)=2.*DKAO(I)
0137              DA2(I)=2.D0*DA2(I)
0138          350 DA1(I)=-2.D0*DA2(I)
0139              CALL JKPOT(H,DA1,DA2,ISNX,1,HO,DKAO,CLOSED)
0140              VJKBA=0.0D0
0141              IJ=0
0142              DO 347 I=1,NBF
0143              DO 347 J=1,I
0144              IJ=IJ+1
0145          347 VJKBA=VJKBA+H(IJ)*DBA(IJ)+HO(IJ)*DBAC(IJ)
0146              WRITE(6,621) VJKB,VJKBA
0147          621 FORMAT(1H ,'COULOMB TERMS:'/
                 . 1H ,'VJKB=',E20.12/1H ,'VJKBA=',E20.12)
0148              EBA=4.D0*TBA+2.D0*VJKBA+VABN(1)
                C  CALCULATE NUCLEAR TERMS AND < BO | ZB/RE | BO >
0149          360 DO 160 N=NVRST,NVEXP
0150              VNHF(N)=0.0D0
0151              VHF=0.0D0
0152              VAB=0.0D0
0153              READ(2) ITITLE
0154              WRITE(6,631) ITITLE
0155          631 FORMAT(/5X,'FILE #2 TITLE = ',18A4)
0156              IF(ITITLE(18).NE.IJLK) GO TO 121
0157              READ(2) II
```

```
0158                  IF(II.NE.NBF) GO TO 121
0159            141   READ(2)
0160                  READ(2) II,X1,Y1,Z1,ZZA,X2,Y2,Z2,ZZB
0161                  WRITE(6,671) II,X1,Y1,Z1,ZZA,X2,Y2,Z2,ZZB                    DEBUG***
0162            671   FORMAT(1H ,'II=',I5/(1H ,4E15.6))                            DEBUG***
0163                  IF(II.NE.2.OR.Z2.NE.RVEXP(N)) GO TO 121
0164                  READ(2) II
0165                  READ(2) II
0166                  DO 143 I=1,2
0167            142   READ(2) NINTS,LAST
0168                  IF(LAST.EQ.0) GO TO 142
0169            143   READ(2) II
0170            144   READ(2) NINTS,LAST,IA,IE,IC,V
0171                  IF(NINTS.EQ.0) GO TO 146
0172                  DO 145 I=1,NINTS
0173                  VAB=VAB+V(I)*DB(ISNX(IA(I))+IE(I))
0174            145   VHF=VHF+V(I)*DA(ISNX(IA(I))+IB(I))
0175            146   IF(LAST.EQ.0) GO TO 144
0176                  WRITE(6,672) VAB                                             DEBUG***
0177            672   FORMAT(1H ,'VAB=',E15.6)                                     DEBUG***
0178                  VNHF(N)=VHF*4.D0+ZA*ZE/RVEXP(N)
0179                  IF(IPERT.EQ.0) VABN(N)=-VAB*4.D0+VABN(N)
0180                  READ(2,END=160)
0181            147   READ(2,END=160) NINTS,LAST
0182                  IF(LAST.EQ.0) GO TO 147
0183                  READ(2,END=160)
0184            160   CONTINUE
0185                  IF(IPERT.EQ.1) EB0=4.D0*TB+2.D0*VJKB+4.D0*VAB
0186                  WRITE(6,661)(VNHF(I),VABN(I),I=1,NVEXP)
0187            661   FORMAT(1H ,'HARTREE-FOCK TERMS'/
                     .  1H ,4X,4HVNHF,16X,4HVABN/(1H ,2E20.12))
0188                  RETURN
0189                  END
```

```
0001                  SUBROUTINE INR(IIN,ICUT,NVAR,NMAX,PAR)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  INTEGER*2 ISNX
0004                  LOGICAL FCNB,RESTRT,CLOSED
0005                  DIMENSION RREST(10),PAR(1),IVRR(10)
0006                  COMMON/FIT/IS,IC,CD,FMIN,TINC,MAXFUN, IPRINT,DSTEP,DMAX,ACC,
                     . W(470),RESTRT,MAXW
0007                  COMMON/INTS/V(100)
0008                  COMMON/INCUT/VNHF(10),VABN(10),VND(10),VNQ(10),VNQD(10),VNQQ(10),
                     . VNM1(10),VNM2(10),VNM3(10),VNM4(10),VEQD(10),VEQC(10),VEM(6,10),
                     . NFIRST(50),NLAST(50),NCNTR(50),NTYPE(50),EX(100),CO(100),
                     . PARMC(10),PARM(10),PARMO(10),ZA,ZB,NVRST,NVEXP, IVAR(10)
0009                  COMMON/IN/ITITLE(18),NBF,NORBA,NORBB,NSYM,DA(1275),DBA(1275),
                     .DBAO(1275),
                     . ALPHAO,BETA1,FCNB(50),RVEXP(10), VEXP(10),ISNX(50),CLOSED
0010                  COMMON/TEST/EBA,EBO
0011                  WRITE(6,660)
0012            660   FORMAT(1H0,'***** RESTART PARAMETERS READ IN *****'/)
0013                  READ(IIN,600) II,(NFIRST(I),NLAST(I),NCNTR(I),NTYPE(I),I=1,II)
0014                  WRITE(IOUT,600)II,(NFIRST(I),NLAST(I),NCNTR(I),NTYPE(I),I=1,II)
0015            600   FORMAT(I5/(4I5))
0016                  READ(IIN,601)II,(EX(I),CC(I),I=1,II),(V(I),I=1,II)
0017                  WRITE(IOUT,601) II,(EX(I),CO(I) ,I=1,II) ,(V(I),I=1,II)
0018            601   FORMAT(I5/(4E20.12))
0019                  READ(IIN,602) ZA,ZB
0020                  WRITE(IOUT,602) ZA ,ZB
0021            602   FORMAT(2E20.12)
0022                  READ(IIN,610) IVRR,(PARM(I),PARMO(I),I=1,NMAX),
                     . (PAR(I),I=1,NVAR)
0023            610   FORMAT(10I5/(2E20.12))
0024                  READ(IIN,611) EBA,EBO
0025                  READ(IIN,610)NVRST
0026                  WRITE(6,605)ZA,ZB,NVRST
0027            605   FORMAT(1H ,'ZA=',E15.6,5X,'ZB=',E15.6/
                     . 1H ,'NVRST=',I5)
0028                  READ(IIN,611)(RREST(I),I=1,NVRST)
0029                  READ(IIN,611)(VNHF(I),I=1,NVRST)
0030                  READ(IIN,611)(VABN(I),I=1,NVRST)
0031                  READ(IIN,611)( VND(I),I=1,NVRST)
0032                  READ(IIN,611)( VNQ(I),I=1,NVRST)
0033                  READ(IIN,611)(VNQD(I),I=1,NVRST)
0034                  READ(IIN,611)(VNQC(I),I=1,NVRST)
0035                  WRITE(6,606) VNQQ
0036            606   FORMAT(1H ,5E15.6)
0037                  READ(IIN,611)(VNM1(I),I=1,NVRST)
0038                  READ(IIN,611)(VNM2(I),I=1,NVRST)
0039                  READ(IIN,611)(VNM3(I),I=1,NVRST)
0040                  READ(IIN,611)(VNM4(I),I=1,NVRST)
0041                  READ(IIN,611)(VEQD(I),I=1,NVRST)
0042                  READ(IIN,611)(VEQC(I),I=1,NVRST)
0043                  WRITE(6,606)VEQQ
0044                  READ(IIN,611)((VEM(I,J),J=1,NVRST),I=1,6)
0045            611   FORMAT(4E20.12)
0046                  IF(NVRST.LT.NVEXP) GO TO 152
0047                  II=0
0048                  DO 100 I=1,NBF
0049                  ISNX(I)=II
0050                  II=II+I
0051                  DA(II)=DA(II)*0.5D0
```

```
0052                  IF(.NOT.CLOSED) DBAO(II)=DBAC(II)*0.5D0
0053            100   DBA(II)=DBA(II)*0.5D0
0054                  DO 150 I=1,NVRST
0055                  IF(DABS(RVEXP(I)-RREST(I)).GT.1.D-10) GO TO 151
0056            150   CONTINUE
0057            152   NVRST=NVRST+1
0058                  DO 160 I=1,10
0059                  IF(IVRR(I).NE.IVAR(I)) GO TO 170
0060            160   CONTINUE
0061                  RETURN
0062            170   FMIN=-1.D0
0063                  DC=0.0D0
0064                  IS=4
0065                  IC=0
0066                  TINC=1.0D0
0067                  DO 171 I=1,MAXW
0068            171   W(I)=0.0D0
0069                  RETURN
0070            151   WRITE(6,661)(RREST(I),I=1,NVRST)
0071            661   FORMAT(1H0,'**********  ERROR  **********'/
                     . 1H ,'ORDER OF RESTART DATA DOES NOT AGREE'/
                     . 1H ,5X,'RREST'/(1H ,E20.12))
0072                  STOP
0073                  END
```

```
0001                  SUBROUTINE PCTR
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  LOGICAL FCNB,LA,LAR,LAA
0004                  INTEGER*2 ISNX
0005                  DIMENSION VALR(6)
0006                  COMMON/CMAT/RI(6,1275)
0007                  COMMON/SCLR/EXI,EXJ,ITYPE,JTYPE,ICNTR,JCNTR,R,CUT,CUT2,CUT3,
                     . ALPHAD,ALPHCB,AP(4),LA(6),RINTG(6),LAR
0008                  COMMON/INOUT/VNHF(10),VAEN(10),VND(10),VNQ(10),VNQQ(10),VNQQ(10),
                     . VNM1(10),VNM2(10),VNM3(10),VNM4(10),VEQQ(10),VEQQ(10),VEM(6,10),
                     . NFIRST(50),NLAST(50),NCNTR(50),NTYPE(50),EX(100),CO(100),
                     . PARMQ(10),PARM(10),PARMC(10),ZA,ZB,NVRST,NVEXP,IVAR(10)
0009                  COMMON/IN/ITITLE(18),NBF,NORBA,NORBB,NSYM,DA(1275),DBA(1275),
                     .DBAQ(1275),
                     . ALPHAQ,BETA1,FCNB(50),RVEXP(10),VEXP(10),ISNX(50)
0010                  COMMON/FCN/INOUT,K,TIMLIM,DELM(10),LAA(6)
                C  INITIALIZE R
0011                  DO 100 J=1,NSYM
0012                  DO 100 I=1,6
0013                100 RI(I,J)=0.0D0
                C  CALCULATE NEEDED NEW R INTEGRALS
0014                  DO 400 N=1,NVEXP
0015                  IF(N.GE.NVRST) GO TO 102
0016                101 IF(LAR) GO TO 105
0017                  GO TO 400
0018                102 IF(INOUT.EQ.0) GO TO 101
0019                  DO 103 I=1,6
0020                103 LA(I)=.FALSE.
0021                  GO TO 110
0022                105 DO 106 I=1,6
0023                106 LA(I)=LAA(I)
0024                110 R=RVEXP(N)
0025                  IJ=0
0026                  DO 350 I=1,NBF
0027                  IIST=NFIRST(I)
0028                  ILST=NLAST(I)
0029                  ITYPE=NTYPE(I)
0030                  ICNTR=NCNTR(I)
0031                  DO 349 J=1,I
0032                  IJ=IJ+1
0033                  IF(.NOT.FCNB(I).OR..NOT.FCNB(J)) GO TO 349
0034                  JIST=NFIRST(J)
0035                  JLST=NLAST(J)
0036                  JTYPE=NTYPE(J)
0037                  JCNTR=NCNTR(J)
0038                  DO 310 IR=1,6
0039                310 VALR(IR)=0.0D0
0040                  DO 346 II=IIST,ILST
0041                  EXI=EX(II)
0042                  TEMP=CO(II)
0043                  DO 346 JJ=JIST,JLST
0044                  EXJ=EX(JJ)
0045                  TCO=TEMP*CO(JJ)
0046                  VALR(1)=VALR(1)+TCO*RINT(ICUM)
0047                  DO 345 IR=2,6
0048                345 VALR(IR)=VALR(IR)+TCO*RINTG(IR)
0049                346 CONTINUE
0050                  DO 348 IR=1,6
0051                  IF(LA(IR)) GO TO 348
```

```
0052                  RI(IP,IJ)=VALR(IR)
0053            348   CONTINUE
0054            349   CONTINUE
0055            350   CONTINUE
0056                  DO 355 IR=1,6
0057            355   VALR(IP)=0.000
0058                  IJ=0
0059                  DO 360 I=1,NBF
0060                  DO 360 J=1,I
0061                  IJ=IJ+1
0062                  DO 358 IR=1,6
0063            358   VALR(IR)=VALR(IR)+RI(IR,IJ)*CBA(IJ)
0064            360   CONTINUE
0065                  DO 361 IR=1,6
0066                  IF(LA(IR)) GO TO 361
0067                  VEM(IR,N)=VALR(IR)*4.00
0068            361   CONTINUE
0069            400   CONTINUE
0070                  WRITE(6,601)((VEM(I,J),I=1,6),J=1,NVEXP)
0071            601   FORMAT(1H ,3HVEM/(1H ,6E15.6))
0072                  RETURN
0073                  END
```

```
0001                  SUBROUTINE PUTQ(RECALC)
0002                  IMPLICIT REAL*8(A-H,O-Z)
0003                  LOGICAL RECALC(10),FCNB,LA,CLOSED
0004                  INTEGER*2 IA,IB,IC,ID,IMU,IF,ISNX,JV(4),ISET(14),IVAL
0005                  DIMENSION DB2(1275),HC(1275),HQQ(1275),DBC2(1275)
0006                  COMMON/QMAT/Q(1275,8),H(1275),HO(1275)
0007                  COMMON/INTS/NINTS,LAST,IA(500),IB(500),IC(500),ID(500),IMU(500),
                     . IF(500),V(500),VQ(500)
0008                  COMMON/SCLR/EXI,EXJ,ITYPE,JTYPE,ICNTR,JCNTR,R,CUT,CUT2,CUT3,
                     . ALPHAO,ALPHQB,AP(4),LA(6),RINTG(6)
0009                  COMMON/INOUT/VNHF(10),VAEN(10),VND(10),VNQ(10),VNQD(10),VNQQ(10),
                     . VNM1(10),VNM2(10),VNM3(10),VNM4(10),VEQC(10),VEQQ(10),VEM(6,10),
                     . NFIRST(50),NLAST(50),NCNTR(50),NTYPE(50),EX(100),CO(100),
                     . PARMC(10),PARM(10),PARMO(10),ZA,ZB,NVRST,NVEXP,IVAR(10)
0010                  COMMON/IN/ITITLE(18),NBF,NORBA,NORBB,NSYM,DA(1275),DBA(1275),
                     .DEAO(1275),
                     . ALPHAQ,BETAI,FCNB(50),EVEXP(10), VEXP(10),ISNX(50),CLOSED
0011                  COMMON/FCN/INOUT
0012                  EQUIVALENCE (VAL,JV),(IVAL,JV(4))
0013                  DATA ISET/9,6,4,1,12,7,5,3,14,8,11,2,13,10/
                     . ,ZERO/1.D-14/
                  C  INITIALIZE Q AND SET UP C CALC
0014                  NI=1
0015                  IF(RECALC(3)) GO TO 212
0016                  IF(RECALC(2)) GO TO 214
0017                  GO TO 301
0018           212    KI=1
0019                  KF=3
0020                  IF(RECALC(2)) GO TO 215
0021                  GO TO 216
0022           214    KI=4
0023           215    KF=8
0024           216    DO 217 K=1,9
0025                  DO 217 I=1,NSYM
0026           217    Q(I,K)=0.0D0
                  C   SET UP DB2
0027                  DO 204 I=1,NSYM
0028                  IF(.NOT.CLOSED) DEO2(I)=-DEAO(I)
0029           204    DB2(I)=-DBA(I)
0030                  II=0
0031                  DO 205 I=1,NBF
0032                  II=II+I
0033                  IF(.NOT.CLOSED) DEC2(II)=2.D0*DB02(II)
0034           205    DB2(II)=DB2(II)+DB2(II)
0035                  DO 300 N=NI,NVEX
0036                  IF(N.LT.NVRST) GO TO 206
0037                  KI=1
0038                  KF=8
0039           206    R=RVEXP(N)
0040                  IJ=0
                  C  CALCULATE NEEDED NEW Q INTS
0041                  DO 230 I=1,NBF
0042                  IIST=NFIRST(I)
0043                  ILST=NLAST(I)
0044                  ITYPE=NTYPE(I)
0045                  ICNTR=NCNTR(I)
0046                  DO 225 J=1,I
0047                  IJ=IJ+1
0048                  IF(.NOT.FCNB(I).OR..NOT.FCNB(J)) GO TO 225
```

```
0049                JIST=NFIRST(J)
0050                JLST=NLAST(J)
0051                JTYPE=NTYPE(J)
0052                JCNTR=NCNTR(J)
0053                DO 220 K=KI,KF
0054                VAL=0.0D0
0055                DO 219 II=IIST,ILST
0056                TEMP=CO(II)
0057                EXI=EX(II)
0058                DO 218 JJ=JIST,JLST
0059                EXJ=EX(JJ)
0060           218  VAL=VAL+TEMP*CO(JJ)*OINT(K)
0061           219  CONTINUE
0062           220  G(IJ,K)=VAL
0063           225  CONTINUE
0064           230  CONTINUE
               C********     DEBUG                                  **************
0065                WRITE(6,605)((Q(I,J),J=1,8),I=1,NSYM)
0066           605  FORMAT(1H ,'Q INTS'/(1H ,8E15.6))
               C********     DEBUG                                  **************
0067                IJ=0
0068                VAB=0.0D0
0069                VAB4=0.0D0
0070                DO 235 I=1,NBF
0071                DO 235 J=1,I
0072                IJ=IJ+1
0073                VAB=VAB+Q(IJ,1)*DBA(IJ)
0074           235  VAB4=VAB4+Q(IJ,4)*DBA(IJ)
0075                IF(KI.EQ.1) VNQD(N)=VNQD(N)*VAB*4.D0
0076                IF(KF.EQ.8) VNQQ(N)=VNQQ(N)*VAB4*4.D0
               C**********    DEBUG                                  ***************************
0077                WRITE(6,610)VAB,VAB4
0078           610  FORMAT(1H ,'Q INT: VAB=' , E15.6,5X,'VAB4=',E15.6)
               C**********    DEBUG                                  ***************************
0079                NINTS=0
0080                LAST=0
0081                IJ=0
               C  FORM 2 EL Q INTS
0082                DO 290 I=1,NBF
0083                IEQJ=0
0084                LSTOP=I-1
0085                DO 275 J=1,I
0086                IJ=IJ+1
0087                IF(I.NE.J) GO TO 236
0088                LSTOP=I
0089                IEQJ=2
0090           236  KSTOP=I
0091                DO 237 K=1,IJ
0092                HQ(K)=0.0D0
0093           237  H(K)=0.0D0
0094                IF(KI.NE.1) GO TO 242
0095                DO 240 K=1,3
0096                DO 239 KL=1,IJ
0097           239  H(KL)=H(KL)+G(KL,K)*Q(IJ,K)
0098           240  CONTINUE
0099           242  IF(KF.NE.8) GO TO 243
0100                DO 241 K=4,8
0101                DO 241 KL=1,IJ
0102           241  HQ(KL)=HQ(KL)+G(KL,K)*Q(IJ,K)
```

```
                  244   LLL=0
                        DO 278 L=1,LSTOP
                        LLL=LLL+L
                        KEQL=1
                        JML=J-L
                        IF(JML.LT.0) KSTOP=LSTOP
                        KL=LLL
                        DO 277 K=1,KSTOP
                        IF(DABS(H(KL)).LT.ZERO.AND.DABS(HQ(KL)).LT.ZERO) GO TO 276
0112                    NINTS=NINTS+1
0113                    VAL=H(KL)
0114                    IVAL=KL
0115                    V(NINTS)=VAL
0116                    VAL=HQ(KL)
0117                    IVAL=KL
0118                    VQ(NINTS)=VAL
0119                    IA(NINTS)=I
0120                    IB(NINTS)=J
0121                    IC(NINTS)=K
0122                    ID(NINTS)=L
0123                    IF(NINTS)=IJ
0124                    M=5
0125                    IF(J-K) 260,265,264
0126                265   M=1
0127                264   M=M+IEQJ
0128                261   M=M+KEQL
0129                      GO TO 266
0130                260   M=9
0131                      IF(JML) 261,262,263
0132                263   M=13
0133                      GO TO 267
0134                262   M=11
0135                267   IF(K.EQ.I) M=M+1
0136                266   IMU(NINTS)=ISET(M)
0137                      IF(NINTS.NE.500) GO TO 276
0138                      WRITE(10)NINTS,LAST,IA,IB,IC,ID,IMU,IF,V
0139                      WRITE(11)NINTS,LAST,IA,IB,IC,ID,IMU,IF,VQ
                  C********     DEBUG                                  **************
0140                      WRITE(6,606)NINTS,LAST,(IA(NN),IB(NN),IC(NN),ID(NN),
                         . IMU(NN),IF(NN),VQ(NN),NN=1,NINTS)
0141                606   FORMAT(1H ,'NINTS=',I5,5X,'LAST=',I5/
                         . 1H ,3X,'IA',3X,'IB',3X,'IC',3X,'ID',2X,'IMU',3X,'IF',4X,'VQ'/
                         . (1H ,6I5,E15.6))
                  C********     DEBUG                                  **************
0142                      NINTS=0
0143                276   KL=KL+K
0144                277   KEQL=0
0145                278   CONTINUE
0146                279   CONTINUE
0147                280   CONTINUE
0148                      LAST=1
0149                      WRITE(10)NINTS,LAST,IA,IB,IC,ID,IMU,IF,V
0150                      WRITE(11)NINTS,LAST,IA,IB,IC,ID,IMU,IF,VQ
                  C**********     DEBUG                               **************
0151                      WRITE(6,606)NINTS,LAST,(IA(NN),IB(NN),IC(NN),ID(NN),
                         . IMU(NN),IF(NN),VQ(NN),NN=1,NINTS)
                  C**********     DEBUG                               ******************
                  C PUT < B | 2JQ-KQ| B > INTO H AND HQ
                  C CHANGE DR FOR JKPCT
```

```
                      DO 281 I=1,NSYM
                      H(I)=0.000
                      HQ(I)=0.000
                      HC(I)=0.000
                      HQQ(I)=0.000
  281                 CBA(I)=-2.00*DB2(I)
                      REWIND 10
                      REWIND 11
                      IF(KI.EQ.1) CALL JKPCT(H,CBA,DB2,ISNX,10,HQ,DBQ2,
                     . CLOSED)
                      IF(KF.EQ.8) CALL JKPCT(HQ,DBA,DB2,ISNX,11,HQQ,DBQ2
                     . ,CLOSED)
                C********     DEBUG                                    **************
                      WRITE(6,607)(HQ(I),I=1,NSYM)
  607                 FORMAT(1H ,'HQ'/(1H ,6E15.6))
                C********     DEBUG                                    **************
                      REWIND 10
                      REWIND 11
                C CHANGE DB BACK
                      DO 282 I=1,NSYM
  282                 DBA(I)=DBA(I)/2.DO
                      II=0
                      DO 283 I=1,NBF
                      II=II+I
  283                 DBA(II)=DBA(II)/2.DO
                C CALC < B B | 2JQ-KQ | B B >
                      VAB=0.000
                      VAB4=0.000
                      IJ=0
                      DO 285 I=1,NBF
                      DO 285 J=1,I
                      IJ=IJ+1
                      VAB=VAB+H(IJ)*DBA(IJ)+HC(IJ)*CBAC(IJ)
  285                 VAB4=VAB4+HQ(IJ)*DBA(IJ)+HQQ(IJ)*DBAQ(IJ)
                      IF(KI.EQ.1) VEQQ(N)=VAB*2.DO
                      IF(KF.EQ.8) VEQQ(N)=VAB4*2.DO
                C**********   DEBUG                                    ****************************
                      WRITE(6,610)VAB,VAB4
                C**********   DEBUG                                    ****************************
  300                 CONTINUE
                      WRITE(6,601)(VNQQ(I),VNQQ(I),VEQQ(I),VEQQ(I),I=1,NVEXP)
  601                 FORMAT(1H ,'Q INTEGRAL TERMS'/
                     . 1H ,4X,4HVNQQ,16X,4HVNCC,16X,4HVEQQ,16X,4HVEQQ/(1H ,4E20.12))
                      RETURN
  301                 IF(NVRST.GT.NVEXP.OR.INCUT.EQ.0) RETURN
                      NI=NVRST
                      GO TO 210
                      END
```

```
0001                  SUBROUTINE MBIIAD(M,N,A,IA,W)                                  1
                    C###### 18/05/70 LAST LIBRARY UPDATE                            2
0002                  IMPLICITREAL*8(A-H,O-Z)                                        3
0003                  DIMENSION A(IA,1),W(1)                                         4
                    C     PARTITION THE WORKING SPACE ARRAY W                      5
                    C     THE FIRST PARTITION HOLDS THE FIRST COMPONENTS OF THE VECTORS OF  6
                    C     THE ELEMENTARY TRANSFORMATIONS                           7
0004                  NRW=M                                                          8
                    C     THE SECOND PARTITION RECORDS ROW INTERCHANGES             9
0005                  NCW=M+M                                                       10
                    C     THE THIRD PARTITION RECORDS COLUMN INTERCHANGES           11
                    C     SET THE INITIAL RECORDS OF ROW AND COLUMN INTERCHANGES    12
0006                  DO 1 I=1,M                                                    13
0007                  W(NRW+I)=0.5D0+DFLOAT(I)                                      14
0008                1 CONTINUE                                                      15
0009                  DO 2 I=1,N                                                    16
0010                  W(NCW+I)=0.5D0+DFLOAT(I)                                      17
0011                2 CONTINUE                                                      18
                    C     'KK' COUNTS THE SEPARATE ELEMENTARY TRANSFORMATIONS       19
0012                  KK=1                                                          20
                    C     FIND LARGEST ROW AND MAKE ROW INTERCHANGES                21
0013                3 RMAX=0.0D0                                                    22
0014                  DO 4 I=KK,M                                                   23
0015                  SUM=0.0D0                                                     24
0016                  DO 5 J=KK,N                                                   25
0017                  SUM=SUM+A(I,J)**2                                             26
0018                5 CONTINUE                                                      27
0019                  IF (RMAX-SUM) 6,4,4                                           28
0020                6 RMAX=SUM                                                      29
0021                  IR=I                                                          30
0022                4 CONTINUE                                                      31
0023                  IF (IR-KK) 7,7,8                                              32
0024                8 SUM=W(NRW+KK)                                                 33
0025                  W(NRW+KK)=W(NRW+IR)                                           34
0026                  W(NRW+IR)=SUM                                                 35
0027                  DO 9 J=1,N                                                    36
0028                  SUM=A(KK,J)                                                   37
0029                  A(KK,J)=A(IR,J)                                               38
0030                  A(IR,J)=SUM                                                   39
0031                9 CONTINUE                                                      40
                    C     FIND LARGEST ELEMENT OF PIVOTAL ROW, AND MAKE COLUMN INTERCHANGES  41
0032                7 RMAX=0.0D0                                                    42
0033                  SUM=0.0D0                                                     43
0034                  DO 10 J=KK,N                                                  44
0035                  SUM=SUM+A(KK,J)**2                                            45
0036                  IF(RMAX-DABS(A(KK,J))) 11,10,10                               46
0037               11 RMAX=DABS(A(KK,J))                                            47
0038                  IR=J                                                          48
0039               10 CONTINUE                                                      49
0040                  IF (IR-KK) 12,12,13                                           50
0041               13 RMAX=W(NCW+KK)                                                51
0042                  W(NCW+KK)=W(NCW+IR)                                           52
0043                  W(NCW+IR)=RMAX                                                53
0044                  DO 14 I=1,M                                                   54
0045                  RMAX=A(I,KK)                                                  55
0046                  A(I,KK)=A(I,IR)                                               56
0047                  A(I,IR)=RMAX                                                  57
0048               14 CONTINUE                                                      58
                    C     REPLACE THE PIVOTAL ROW BY THE VECTOR OF THE TRANSFORMATION  59
```

```
0049            12 SIGMA=DSQRT(SUM)                                                  60
0050               BSQ=DSQRT(SUM+SIGMA*DABS(A(KK,KK)))                               61
0051               W(KK)=DSIGN(SIGMA+DABS(A(KK,KK)),A(KK,KK))/BSQ                    62
0052               A(KK,KK)=-DSIGN(SIGMA,A(KK,KK))                                   63
0053               KP=KK+1                                                           64
0054               IF (KP-N) 15,15,16                                                65
0055            15 DO 17 J=KP,N                                                      66
0056               A(KK,J)=A(KK,J)/BSQ                                               67
0057            17 CONTINUE                                                          68
                C     APPLY THE TRANSFORMATION TO THE REMAINING ROWS OF A            69
0058               IF (KP-M) 18,18,16                                                70
0059            18 DO 19 I=KP,M                                                      71
0060               SUM=W(KK)*A(I,KK)                                                 72
0061               DO 20 J=KP,N                                                      73
0062               SUM=SUM+A(KK,J)*A(I,J)                                            74
0063            20 CONTINUE                                                          75
0064               A(I,KK)=A(I,KK)-SUM*W(KK)                                         76
0065               DO 21 J=KP,N                                                      77
0066               A(I,J)=A(I,J)-SUM*A(KK,J)                                         78
0067            21 CONTINUE                                                          79
0068            19 CONTINUE                                                          80
0069               KK=KP                                                             81
0070               GO TO 3                                                           82
                C     AT THIS STAGE THE REDUCTION OF A IS COMPLETE                   83
                C     NOW WE BUILD UP THE GENERALIZED INVERSE                        84
                C     APPLY THE FIRST ELEMENTARY TRANSFORMATION                      85
0071            16 KK=M                                                              86
0072               KP=M+1                                                            87
0073               SUM=W(M)/A(M,M)                                                   88
0074               IF (N-M) 33,33,34                                                 89
0075            34 DO 35 J=KP,N                                                      90
0076               A(M,J)=-SUM*A(M,J)                                                91
0077            35 CONTINUE                                                          92
0078            33 A(M,M)=1.000/A(M,M)-SUM*W(M)                                      93
                C     NOW APPLY THE OTHER (M-1) TRANSFORMATIONS                      94
0079            36 KP=KK                                                             95
0080               KK=KP-1                                                           96
0081               IF (KK) 37,37,38                                                  97
                C     FIRST TRANSFORM THE LAST (M-KK) ROWS                           98
0082            38 DO 39 I=KP,M                                                      99
0083               SUM=0.000                                                        100
0084               DO 40 J=KP,N                                                     101
0085               SUM=SUM+A(KK,J)*A(I,J)                                           102
0086            40 CONTINUE                                                         103
0087               DO 41 J=KP,N                                                     104
0088               A(I,J)=A(I,J)-SUM*A(KK,J)                                        105
0089            41 CONTINUE                                                         106
0090               W(I)=-SUM*W(KK)                                                  107
0091            39 CONTINUE                                                         108
                C     THEN CALCULATE THE NEW ROW IN POSITION KK                     109
0092               DO 42 J=KP,N                                                     110
0093               SUM=-W(KK)*A(KK,J)                                               111
0094               DO 43 I=KP,M                                                     112
0095               SUM=SUM-A(I,KK)*A(I,J)                                           113
0096            43 CONTINUE                                                         114
0097               A(KK,J)=SUM/A(KK,KK)                                             115
0098            42 CONTINUE                                                         116
                C     AND REVISE THE COLUMN IN POSITION KK                          117
0099               SUM=1.000-W(KK)**2                                               118
```

```
0100                  DO 44 I=KP,M                                          119
0101                  SUM=SUM-A(I,KK)*W(I)                                  120
0102                  A(I,KK)=W(I)                                          121
0103               44 CONTINUE                                              122
0104                  A(KK,KK)=SUM/A(KK,KK)                                 123
0105                  GO TO 36                                              124
                   C  RESTORE THE ROW INTERCHANGES                          125
0106               37 DO 45 I=1,M                                           126
0107               46 IR=IDINT(W(NRW+I))                                    127
0108                  IF (I-IR) 47,45,45                                    128
0109               47 SUM=W(NRW+I)                                          129
0110                  W(NRW+I)=W(NRW+IR)                                    130
0111                  W(NRW+IR)=SUM                                         131
0112                  DO 48 J=1,N                                           132
0113                  SUM=A(I,J)                                            133
0114                  A(I,J)=A(IR,J)                                        134
0115                  A(IR,J)=SUM                                           135
0116               48 CONTINUE                                              136
0117                  GO TO 46                                              137
0118               45 CONTINUE                                              138
                   C  RESTORE THE COLUMN INTERCHANGES                       139
0119                  DO 49 J=1,N                                           140
0120               50 IR=IDINT(W(NCW+J))                                    141
0121                  IF (J-IR) 51,49,49                                    142
0122               51 SUM=W(NCW+J)                                          143
0123                  W(NCW+J)=W(NCW+IR)                                    144
0124                  W(NCW+IR)=SUM                                         145
0125                  DO 52 I=1,M                                           146
0126                  SUM=A(I,J)                                            147
0127                  A(I,J)=A(I,IR)                                        148
0128                  A(I,IR)=SUM                                           149
0129               52 CONTINUE                                              150
0130                  GO TO 50                                              151
0131               49 CONTINUE                                              152
0132                  RETURN                                                153
0133                  END                                                   154
```

```
0001                  SUBROUTINE VA05AC(CALFUN,M,N,F,X,DSTEP,DMAX,ACC,MAXFUN,IPRINT,W)
                C****** 10/09/74 LAST LIBRARY UPDATE                                    2
0002                  IMPLICIT REAL*8(A-H,O-Z)                                          3
0003                  LOGICAL RESTRT
                C     NOTE  THAT THE INSTRUCTION CALLING SUBROUTINE 'MBIIA',            4
                C      ON LINE NUMBER '138' , IS NOT STANDARD FORTRAN                   5
0004                  DIMENSION F(1),X(1),W(1)                                          6
0005                  COMMON/FIT/IS,IC,DD,FMIN,TINC,EXX(474),RESTRT,MAXW
                C     SET VARIOUS PARAMETERS                                            7
0006                  MAXC=0                                                            8
                C     'MAXC' COUNTS THE NUMBER OF CALLS OF CALFUN                       9
0007                  MPN=M+N                                                          10
0008                  NT=N+2                                                           11
0009                  NTEST=0                                                          12
                C     'NT' AND 'NTEST' CAUSE AN ERROR RETURN IF F(X) DOES NOT DECREASE  13
0010                  DTEST=FLOAT(N+N)-0.5                                             14
                C     'DTEST' IS USED IN A TEST TO MAINTAIN LINEAR INDEPENDENCE         15
                C     PARTITION THE WORKING SPACE ARRAY W                               16
                C     THE FIRST PARTITION HOLDS THE JACOBIAN APPROXIMATION              17
0011                  NWI=M*N                                                          18
                C     THE NEXT PARTITION HOLDS THE GENERALIZED INVERSE                  19
0012                  NWX=NWI+MPN*N                                                    20
                C     THE NEXT PARTITION HOLDS THE BEST VECTOR X                        21
0013                  NWF=NWX+N                                                        22
                C     THE NEXT PARTITION HOLDS THE BEST VECTOR F                        23
0014                  NWC=NWF+M                                                        24
                C     THE NEXT PARTITION HOLDS THE COUNTS OF THE INDEPENDENT DIRECTIONS 25
0015                  NWD=NWC+N                                                        26
                C     THE NEXT PARTITION HOLDS THE INDEPENDENT DIRECTIONS               27
0016                  NWW=NWD+N*N                                                      28
                C     THE REMAINDER OF W IS USED FOR SCRATCH VECTORS                    29
0017                  NWV=NWW+N                                                        30
0018                  NWT=NWV+M                                                        31
0019                  NWU=NWT+N                                                        32
0020                  MAXW=2*(N*(M+N)+M)+5*N
                C     USUALLY 'FMIN' IS THE LEAST CALCULATED VALUE OF F(X)              34
                C     USUALLY 'DD' IS THE SQUARE OF THE CURRENT STEP LENGTH             36
0021                  DSS=DSTEP*DSTEP                                                  37
0022                  DM=DMAX*DMAX                                                     38
0023                  PARM=DSQRT(ACC)/DMAX                                             39
                C     'PARM' IS THE LEAST VALUE OF THE MARQUARDT PARAMETER              40
0024                  DPAR=10.0D0*DM                                                   41
                C     'DPAR' AND 'NTPAR' ARE USED TO REGULATE THE MARQUARDT PARAMETER   42
                C     'IS' CONTROLS A GO TO STATEMENT FOLLOWING A CALL OF CALFUN        44
                C     'TINC' IS USED IN THE CRITERION TO INCREASE THE STEP LENGTH       47
                C     START A NEW PAGE FOR PRINTING                                     48
0025                  IF(RESTRT) GO TO 200
0026                  FMIN=-1.D0
0027                  DD=0.0D0
0028                  IS=4
0029                  IC=0
0030                  TINC=1.D0
0031            200   IF (IPRINT) 1,3,1                                               49
0032                1 PRINT 2                                                         50
0033                2 FORMAT (1H1,4X,'THE FOLLOWING OUTPUT IS PROVIDED BY SUBROUTINE',51
                   1' VA05AC'//)                                                      52
0034                  IPC=0                                                           53
0035                  GO TO 3                                                         54
                C     TEST WHETHER THERE HAVE BEEN MAXFUN CALLS OF CALFUN             55
```

```
0036            4 IF (MAXFUN-MAXC) 5,5,3                                        56
0037            5 IF (IPRINT) 139,140,139                                       57
0038          140 IPRINT=2                                                      58
0039              GO TO 19                                                      59
0040          139 PRINT 6,MAXC                                                  60
0041            6 FORMAT (///5X,'ERROR RETURN FROM VA05A BECAUSE THERE HAVE BEEN',61
                 1I5,' CALLS OF CALFUN')                                        62
0042              GO TO 7                                                       63
C               CALL THE SUBROUTINE CALFUN                                      64
0043            3 MAXC=MAXC+1                                                   65
0044              CALL CALFUN (M,N,F,X)                                         66
C               CALCULATE THE SUM OF SQUARES                                    67
0045              FSQ=0.0D0                                                     68
0046              DO 8 I=1,M                                                    69
0047              FSQ=FSQ+F(I)*F(I)                                             70
0048            8 CONTINUE                                                      71
C               TEST FOR ACC
0049              IF(FSQ-ACC) 180,180,181
0050          180 WRITE(6,780)
0051          780 FORMAT(/1H ,'FSQ .LT. ACC'/)
0052              GO TO 7
C               TEST FOR ERROR RETURN BECAUSE F(X) DOES NOT DECREASE            72
0053          181 GO TO (9,10,9,10),IS                                          73
0054            9 IF (FSQ-FMIN) 11,12,12                                        74
0055           12 IF (DC-DSS) 13,13,10                                          75
0056           13 NTEST=NTEST-1                                                 76
0057              IF (NTEST) 14,14,10                                           77
0058           14 IF (IPRINT) 15,17,15                                          78
0059           17 IPRINT=1                                                      79
0060              GO TO 19                                                      80
0061           15 PRINT 16                                                      81
0062           16 FORMAT (///5X,'ERROR RETURN FROM VA05A BECAUSE F(X) NO LONGER',82
                 1' DECREASES'//5X,'THIS MAY BE DUE TO THE VALUES OF DSTEP',    83
                 2' AND ACC, OR TO LOSS OF RANK IN THE JACOBIAN MATRIX')        84
C               PROVIDE PRINTING OF FINAL SOLUTION IF REQUESTED                 85
0063            7 IF (IPRINT) 18,19,18                                          86
0064           18 PRINT 20,MAXC                                                 87
0065           20 FORMAT (///5X,'THE FINAL SOLUTION CALCULATED BY VA05A REQUIRED',88
                 1I5,' CALLS OF CALFUN, AND IS')                                89
0066              PRINT 21,(I,W(NWX+I),I=1,N)                                   90
0067           21 FORMAT (//4X,'I',7X,'X(I)',10X,'I',7X,'X(I)',10X,'I',7X,'X(I)',91
                 110X,'I',7X,'X(I)',10X,'I',7X,'X(I)'//5(I5,D17.8))             92
0068              PRINT 22,(I,W(NWF+I),I=1,M)                                   93
0069           22 FORMAT (//4X,'I',7X,'F(I)',10X,'I',7X,'F(I)',10X,'I',7X,'F(I)',94
                 110X,'I',7X,'F(I)',10X,'I',7X,'F(I)'//5(I5,D17.8))             95
0070              PRINT 23,FMIN                                                 96
0071           23 FORMAT (/5X,'THE SUM OF SQUARES IS',D17.8)                    97
C               RESTORE THE BEST VALUES OF X AND F                              98
0072           19 DO 135 I=1,N                                                  99
0073              X(I)=W(NWX+I)                                                100
0074          135 CONTINUE                                                     101
0075              DO 136 I=1,M                                                 102
0076              F(I)=W(NWF+I)                                                103
0077          136 CONTINUE                                                     104
0078              RETURN                                                       105
0079           11 NTEST=NT                                                     106
C               PROVIDE ORDINARY PRINTING IF REQUESTED                         107
0080           10 IF(IABS(IPRINT)-1) 39,38,40                                  108
0081           38 PRINT 41,MAXC                                                109
```

```
0092            41 FORMAT (///5X,'AT THE ',I5,' TH CALL OF CALFUN WE HAVE')                110
0083            42 PRINT 21,(I,X(I),I=1,N)                                                 111
0084               PRINT 23,FSQ                                                            112
0085               IF (IPRINT) 39,39,142                                                   113
0086           142 PRINT 22, (I,F(I),I=1,M)                                                114
0087               GO TO 39                                                                115
0088            40 IPC=IPC-1                                                               116
0089               IF (IPC) 43,43,39                                                       117
0090            43 PRINT 44,MAXC                                                           118
0091            44 FORMAT (///5X,'THE BEST ESTIMATE AFTER',I5,' CALLS OF CALFUN IS')       119
0092               IPC=IABS(IPRINT)                                                        120
0093               IF (FSQ-FMIN) 42,45,45                                                  121
0094            45 IF (FMIN) 42,46,46                                                      122
0095            46 PRINT 21,(I,W(NWX+I),I=1,N)                                             123
0096               PRINT 23,FMIN                                                           124
0097               IF (IPRINT) 39,39,143                                                   125
0098           143 PRINT 22, (I,W(NWF+I),I=1,M)                                            126
0099            39 GO TO (49,47,47,48),IS                                                  127
                C     STORE THE INITIAL VECTORS X AND F                                    128
0100            48 IF (IC) 50,50,51                                                        129
0101            50 DO 52 I=1,N                                                             130
0102               W(NWX+I)=X(I)                                                           131
0103            52 CONTINUE                                                                132
0104               GO TO 54                                                                133
                C     CALCULATE THE INITIAL JACOBIAN APPROXIMATION                         134
0105            51 K=IC                                                                    135
0106               DO 55 I=1,M                                                             136
0107               W(K)=(F(I)-W(NWF+I))/DSTEP                                              137
0108               K=K+N                                                                   138
0109            55 CONTINUE                                                                139
                C     TEST WHETHER THE MOST RECENT X IS BEST                               140
0110               IF (FMIN-FSQ) 56,56,57                                                  141
0111            56 X(IC)=W(NWX+IC)                                                         142
0112               GO TO 58                                                                143
0113            57 W(NWX+IC)=X(IC)                                                         144
0114            54 DO 53 I=1,M                                                             145
0115               W(NWF+I)=F(I)                                                           146
0116            53 CONTINUE                                                                147
0117               FMIN=FSQ                                                                148
                C     SET X FOR THE NEXT CALL OF CALFUN                                    149
0118            58 IC=IC+1                                                                 150
0119               IF (IC-N) 59,59,60                                                      151
0120            59 X(IC)=W(NWX+IC)+DSTEP                                                   152
0121               GO TO 3                                                                 153
                C     SET THE DIRECTION MATRIX                                             154
0122            60 K=NWD                                                                   155
0123               DO 61 I=1,N                                                             156
0124               DO 62 J=1,N                                                             157
0125               K=K+1                                                                   158
0126               W(K)=0.0D0                                                              159
0127            62 CONTINUE                                                                160
0128               W(K+I-N)=1.D0                                                           161
0129               W(NWC+I)=1.0D0+DFLOAT(N-I)                                              162
0130            61 CONTINUE                                                                163
                C     SET THE MARQUARDT PARAMETER TO ITS LEAST VALUE                       164
0131            24 PAR=PARM                                                                165
                C     COPY THE JACOBIAN AND APPEND THE MARQUARDT MATRIX                    166
0132            25 PPAR=PAR*PAR                                                            167
0133               NTPAR=0                                                                 168
```

```
0134            63 KK=0                                                          169
0135               K=NWI+NWI                                                     170
0136               DO 26 I=1,N                                                   171
0137               DO 141 J=1,M                                                  172
0138               KK=KK+1                                                       173
0139               W(KK+NWI)=W(KK)                                               174
0140           141 CONTINUE                                                      175
0141               DO 27 J=1,N                                                   176
0142               K=K+1                                                         177
0143               W(K)=0.0D0                                                    178
0144            27 CONTINUE                                                      179
0145               W(K+I-N)=PAR                                                  180
0146            26 CONTINUE                                                      181
                C     CALCULATE THE GENERALIZED INVERSE OF J                     182
0147                  CALL MRILAD(N,MPN,W(NWI+1),N,W(NWW+1))                     183
                C     NOTE THAT THE THIRD AND FIFTH ENTRIES OF THIS ARGUMENT LIST 184
                C     STAND FOR ONE-DIMENSIONAL ARRAYS.                          185
                C     START THE ITERATION BY TESTING FMIN                        186
                C                                                                187
0148            64 IF (FMIN-ACC) 7,7,65                                          188
                C     NEXT PREDICT THE DESCENT AND MARQUARDT MINIMA              189
0149            65 DS=0.0D0                                                      190
0150               DN=0.0D0                                                      191
0151               SP=0.0D0                                                      192
0152               DO 66 I=1,N                                                   193
0153               X(I)=0.0D0                                                    194
0154               F(I)=0.0D0                                                    195
0155               K=I                                                           196
0156               DO 67 J=1,M                                                   197
0157               X(I)=X(I)-W(K)*W(NWF+J)                                       198
0158               F(I)=F(I)-W(NWI+K)*W(NWF+J)                                   199
0159               K=K+N                                                         200
0160            67 CONTINUE                                                      201
0161               DS=DS+X(I)*X(I)                                               202
0162               DN=DN+F(I)*F(I)                                               203
0163               SP=SP+X(I)*F(I)                                               204
0164            66 CONTINUE                                                      205
                C     PREDICT THE REDUCTION IN F(X) DUE TO THE MARQUARDT STEP    206
                C     AND ALSO PREDICT THE LENGTH OF THE STEEPEST DESCENT STEP   207
0165               PRED=SP*SP                                                    208
0166               DMULT=0.0D0                                                   209
0167               K=0                                                           210
0168               DO 68 I=1,M                                                   211
0169               AP=0.0D0                                                      212
0170               AD=0.0D0                                                      213
0171               DO 69 J=1,N                                                   214
0172               K=K+1                                                         215
0173               AP=AP+W(K)*F(J)                                               216
0174               AD=AD+W(K)*X(J)                                               217
0175            69 CONTINUE                                                      218
0176               PRED=PRED-AP*AP                                               219
0177               DMULT=DMULT+AD*AD                                             220
0178            68 CONTINUE                                                      221
                C     TEST FOR CONVERGENCE                                       222
0179               IF (DN-DM) 28,28,29                                           223
0180            28 AP=DSQRT(DN)
0181               IF(PRED+2.0D0*PPAR*AP*(DMAX-AP)-ACC)770,770,70
0182            29 IF (PRED+PPAR*(DM-DN)-ACC) 770,770,70
0183           770 WRITE(6,771)
0184           771 FORMAT(1H ,'PREDICTED REDUCTION IN F(X) IS LESS THAN ACC')
```

```
0185                GO TO 7                                                    226
                C   TEST WHETHER TO APPLY THE FULL MARQUARDT CORRECTION      227
0186              70 DMULT=DS/DMULT                                            228
0187                 CS=DS*DMULT*DMULT                                         229
0188              71 IS=2                                                      230
0189                 IF (DN-DD) 72,72,73                                       231
                C   TEST THAT THE MARQUARDT PARAMETER HAS ITS LEAST VALUE     232
0190              72 IF (PAR-PARM) 30,30,24                                    233
0191              30 DD=DMAX1(DN,DSS)                                          234
0192                 DS=0.2500*DN                                              235
0193                 TINC=1.000                                                236
0194                 IF (DN-DSS) 74,132,132                                    237
0195              74 IS=3                                                      238
0196                 GO TO 103                                                 239
                C   TEST WHETHER TO INCREASE THE MARQUARDT PARAMETER           240
0197              73 IF (DN-DPAR) 31,31,32                                     241
0198              31 NTPAR=0                                                   242
0199                 GO TO 33                                                  243
0200              32 IF (NTPAR) 34,34,35                                       244
0201              34 NTPAR=1                                                   245
0202                 PTM=DN                                                    246
0203                 GO TO 33                                                  247
0204              35 NTPAR=NTPAR+1                                             248
0205                 PTM=DMIN1(PTM,DN)                                         249
0206                 IF (NTPAR-NT) 33,36,36                                    250
                C   SET THE LARGER VALUE OF THE MARQUARDT PARAMETER           251
0207              36 PAR=PAR*(PTM/DM)**0.2500                                  252
0208                 IF(6.0D0*DD-DM)137,25,25                                  253
0209             137 AP=DSQRT(PRED/DN)                                         254
0210                 IF (AP-PAR) 25,25,138                                     255
0211             138 PAR=DMIN1(AP,PAR*(DM/(6.0D0*DD))**0.2500)                 256
0212                 GO TO 25                                                  257
                C   TEST WHETHER TO USE THE STEEPEST DESCENT DIRECTION        258
0213              33 IF (DS-DD) 75,76,76                                       259
                C   TEST WHETHER THE INITIAL VALUE OF DD HAS BEEN SET         260
0214              76 IF (DD) 77,77,78                                          261
0215              77 DD=DMIN1(DM,DS)                                           262
0216                 IF (DD-DSS) 79,78,78                                      263
0217              79 DD=DSS                                                    264
0218                 GO TO 71                                                  265
                C   SET THE MULTIPLIER OF THE STEEPEST DESCENT DIRECTION      266
0219              78 ANMULT=0.00                                               267
0220                 DMULT=DMULT*DSQRT(DD/DS)                                  268
0221                 GO TO 80                                                  269
                C   INTERPOLATE BETWEEN THE STEEPEST DESCENT AND MARQUARDT DIRECTIONS 270
0222              75 SP=SP*DMULT                                               271
0223                 ANMULT=(DD-DS)/((SP-DS)+DSQRT((SP-DD)**2+(DN-DD)*(DD-DS))) 272
0224                 DMULT=DMULT*(1.0D0-ANMULT)                                273
                C   CALCULATE THE CORRECTION TO X, AND ITS ANGLE WITH THE FIRST 274
                C   DIRECTION                                                  275
0225              80 DN=0.000                                                  276
0226                 SP=0.000                                                  277
0227                 DO 81 I=1,N                                               278
0228                 F(I)=DMULT*X(I)+ANMULT*F(I)                               279
0229                 DN=DN+F(I)*F(I)                                           280
0230                 SP=SP+F(I)*W(NWD+I)                                       281
0231              81 CONTINUE                                                  282
0232                 DS=0.2500*DN                                              283
                C   TEST WHETHER AN EXTRA STEP IS NEEDED FOR INDEPENDENCE
```

```
                      IF (W(NWC+1)-DTEST) 132,132,82                                    284
0234                82 IF (SP*SP-DS) 83,132,132                                          285
                    C     TAKE THE EXTRA STEP AND UPDATE THE DIRECTION MATRIX            286
0235                83 DO 84 I=1,N                                                       287
0236                   X(I)=W(NWX+I)+DSTEP*W(NWC+I)                                      288
0237                   W(NWC+I)=W(NWC+I+1)+1.DO                                          289
0238                84 CONTINUE                                                          290
0239                   W(NWD)=1.DO                                                       291
0240                   IF(N.LE.1)GO TO 4                                                 292
0241                   DO 85 I=1,N                                                       293
0242                   K=NWD+I                                                           294
0243                   SP=W(K)                                                           295
0244                   DO 86 J=2,N                                                       296
0245                   W(K)=W(K+N)                                                       297
0246                   K=K+N                                                             298
0247                86 CONTINUE                                                          299
0248                   W(K)=SP                                                           300
0249                85 CONTINUE                                                          301
0250                   GO TO 4                                                           302
                    C     EXPRESS THE NEW DIRECTION IN TERMS OF THOSE OF THE DIRECTION   303
                    C     MATRIX, AND UPDATE THE COUNTS IN W(NWC+I) ETC.                304
0251              132 IF(N.GE.2)GO TO 153                                                305
0252                  IS=1                                                             306
0253                  GO TO 152                                                        307
0254              153 SP=0.DO                                                          308
0255                  K=NWD                                                            309
0256                  DO 87 I=1,N                                                      310
0257                  X(I)=DW                                                          311
0258                  DW=0.DO                                                          312
0259                  DO 88 J=1,N                                                      313
0260                  K=K+1                                                            314
0261                  DW=DW+F(J)*W(K)                                                  315
0262               88 CONTINUE                                                         316
0263                  GO TO (89,90),IS                                                 317
0264               90 W(NWC+I)=W(NWC+I)+1.DO                                           318
0265                  SP=SP+DW*DW                                                      319
0266                  IF (SP-DS) 87,87,91                                              320
0267               91 IS=1                                                             321
0268                  KK=I                                                             322
0269                  X(I)=DW                                                          323
0270                  GO TO 92                                                         324
0271               89 X(I)=DW                                                          325
0272               92 W(NWC+I)=W(NWC+I+1)+1.DO                                         326
0273               87 CONTINUE                                                         327
0274                  W(NWD)=1.DO                                                      328
                    C     REORDER THE DIRECTIONS SO THAT KK IS FIRST                   329
0275                  IF (KK-1) 93,93,94                                               330
0276               94 KS=NWC+KK*N                                                      331
0277                  DO 95 I=1,N                                                      332
0278                  K=KS+I                                                           333
0279                  SP=W(K)                                                          334
0280                  DO 96 J=2,KK                                                     335
0281                  W(K)=W(K-N)                                                      336
0282                  K=K-N                                                            337
0283               96 CONTINUE                                                         338
0284                  W(K)=SP                                                          339
0285               95 CONTINUE                                                         340
                    C     GENERATE THE NEW ORTHOGONAL DIRECTION MATRIX                 341
0286               93 DO 97 I=1,N                                                      342
```

```
0297                W(NWW+I)=0.0D0                                          343
0298             97 CONTINUE                                                344
0299                SP=X(1)*X(1)                                            345
0300                K=NWD                                                   346
0301                DO 98 I=2,N                                             347
0302                 DS=DSQRT(SP*(SP+X(I)*X(I)))                            348
0303                DW=SP/DS                                                349
0304                DS=X(I)/DS                                              350
0305                SP=SP+X(I)*X(I)                                         351
0306                DO 99 J=1,N                                             352
0307                K=K+1                                                   353
0308                W(NWW+J)=W(NWW+J)+X(I-1)*W(K)                           354
0309                W(K)=DW*W(K+N)-DS*W(NWW+J)                              355
0310             99 CONTINUE                                                356
0311             98 CONTINUE                                                357
0312                SP=1.0D0/DSQRT(DN)                                      358
0313                DO 100 I=1,N                                            359
0314                K=K+1                                                   360
0315                W(K)=SP*F(I)                                            361
0316            100 CONTINUE                                                362
                C   PREDICT THE NEW RIGHT HAND SIDES                        363
0317            152 FNP=0.0D0                                               364
0318                K=0                                                     365
0319                DO 101 I=1,M                                            366
0320                W(NWW+I)=W(NWF+I)                                       367
0321                DO 102 J=1,N                                            368
0322                K=K+1                                                   369
0323                W(NWW+I)=W(NWW+I)+W(K)*F(J)                             370
0324            102 CONTINUE                                                371
0325                FNP=FNP+W(NWW+I)**2                                     372
0326            101 CONTINUE                                                373
                C   CALCULATE THE NEXT VECTOR X, AND THEN CALL CALFUN      374
0327            103 DO 104 I=1,N                                            375
0328                X(I)=W(NWX+I)+F(I)                                      376
0329            104 CONTINUE                                                377
0330                GO TO 4                                                 378
                C   UPDATE THE STEP SIZE                                   379
0331             49 DMULT=0.9D0*FMIN+0.1D0*FNF-FSQ                          380
0332                IF (DMULT) 105,105,105                                  381
0333            105 DD=DMAX1(DSS,0.25D0*DD)                                 382
0334                 TINC=1.0D0                                             383
0335                IF (FSQ-FMIN) 106,107,107                               384
                C   TRY THE TEST TO DECIDE WHETHER TO INCREASE THE STEP LENGTH  385
0336            108 SP=0.0D0                                                386
0337                SS=0.0D0                                                387
0338                DO 109 I=1,M                                            388
0339                SP=SP+DABS(F(I)*(F(I)-W(NWW+I)))                        389
0340                SS=SS+(F(I)-W(NWW+I))**2                                390
0341            109 CONTINUE                                                391
0342                PJ=1.0D0+DMULT/(SP+DSQRT(SP*SP+DMULT*SS))               392
0343                SP=DMIN1(4.0D0,TINC,PJ)                                 393
0344                TINC=PJ/SP                                              394
0345                DD=DMIN1(DM,SP*DD)                                      395
0346                GO TO 106                                               396
                C   IF F(X) IMPROVES STORE THE NEW VALUE OF X               397
0347             47 IF (FSQ-FMIN) 106,110,110                               398
0348            106 FMIN=FSQ                                                399
0349                DO 111 I=1,N                                            400
0350                SP=X(I)                                                 401
```

```
0341                  X(I)=W(NWX+I)                                          402
0342                  W(NWX+I)=SP                                            403
0343               111 CONTINUE                                              404
0344                  DO 112 I=1,M                                           405
0345                  SP=F(I)                                                406
0346                  F(I)=W(NWF+I)                                          407
0347                  W(NWF+I)=SP                                            408
0348               112 CONTINUE                                              409
0349               110 GO TO (107,107,113),IS                                410
0350               113 IS=2                                                  411
0351                  IF (FMIN-ACC) 7,7,83                                   412
                   C     CALCULATE THE CHANGES IN X AND IN F                413
0352               107 DS=0.000                                              414
0353                  DO 114 I=1,N                                           415
0354                  X(I)=X(I)-W(NWX+I)                                     416
0355                  DS=DS+X(I)*X(I)                                        417
0356               114 CONTINUE                                              418
0357                  DO 115 I=1,M                                           419
0358                  F(I)=F(I)-W(NWF+I)                                     420
0359               115 CONTINUE                                              421
                   C     CALCULATE THE GENERALIZED INVERSE TIMES THE CHANGE IN X 422
0360                  K=NWI                                                  423
0361                  SS=0.000                                               424
0362                  DO 116 I=1,NPN                                         425
0363                  SP=0.000                                               426
0364                  DO 117 J=1,N                                           427
0365                  K=K+1                                                  428
0366                  SP=SP+W(K)*X(J)                                        429
0367               117 CONTINUE                                              430
0368                  W(NWV+I)=SP                                            431
0369                  SS=SS+SP*SP                                            432
0370               116 CONTINUE                                              433
                   C     CALCULATE J TIMES THE CHANGE IN F                  434
                   C     ALSO APPLY PROJECTION TO THE GENERALIZED INVERSE   435
0371                  DO 118 I=1,N                                           436
0372                  ST=0.000                                               437
0373                  K=NWI+I                                                438
0374                  DO 119 J=1,NPN                                         439
0375                  ST=ST+W(K)*W(J+NWV)                                    440
0376                  K=K+N                                                  441
0377               119 CONTINUE                                              442
0378                  ST=ST/SS                                               443
0379                  K=NWI+I                                                444
0380                  DO 120 J=1,NPN                                         445
0381                  W(K)=W(K)-ST*W(J+NWV)                                  446
0382                  K=K+N                                                  447
0383               120 CONTINUE                                              448
0384                  ST=PPAR*X(I)                                           449
0385                  K=I                                                    450
0386                  DO 121 J=1,M                                           451
0387                  ST=ST+W(K)*F(J)                                        452
0388                  K=K+N                                                  453
0389               121 CONTINUE                                              454
0390                  W(NWW+I)=ST                                            455
0391               118 CONTINUE                                              456
                   C     REVISE J AND CALCULATE ROW VECTOR FOR CORRECTION TO INVERSE 457
0392                  IC=0                                                   458
0393                  K=0                                                    459
0394                  KK=NWI                                                 460
```

```
0395                SP=0.000                                                    461
0396                SPP=0.000                                                   462
0397                DO 122 I=1,M                                                463
0398                SS=F(I)                                                     464
0399                ST=F(I)                                                     465
0400                DO 123 J=1,N                                                466
0401                IC=IC+1                                                     467
0402                KK=KK+1                                                     468
0403                SS=SS-W(IC)*X(J)                                            469
0404                ST=ST-W(KK)*W(NWW+J)                                        470
0405           123  CONTINUE                                                    471
0406                SS=SS/D                                                     472
0407                W(NWV+I)=ST                                                 473
0408                SP=SP+F(I)*ST                                               474
0409                SPP=SPP+ST*ST                                               475
0410                DO 124 J=1,N                                                476
0411                K=K+1                                                       477
0412                W(K)=W(K)+SS*X(J)                                           478
0413           124  CONTINUE                                                    479
0414           122  CONTINUE                                                    480
0415                DO 125 I=1,N                                                481
0416                ST=PAR*X(I)                                                 482
0417                DO 126 J=1,N                                                483
0418                KK=KK+1                                                     484
0419                ST=ST-W(KK)*W(NWW+J)                                        485
0420           126  CONTINUE                                                    486
0421                W(NWT+I)=ST                                                 487
0422                SP=SP+PAR*X(I)*ST                                           488
0423                SPP=SPP+ST*ST                                               489
0424           125  CONTINUE                                                    490
C                   TEST THAT THE SCALAR PRODUCT IS SUFFICIENTLY ACCURATE       491
0425                IF(0.0100*SPP-DABS(SP-SPP))63,63,127                        492
C                   CALCULATE THE NEW GENERALIZED INVERSE                       493
0426           127  DO 128 I=1,N                                                494
0427                K=NWI+I                                                     495
0428                ST=X(I)                                                     496
0429                DO 129 J=1,M                                                497
0430                ST=ST-W(K)*F(J)                                             498
0431                K=K+N                                                       499
0432           129  CONTINUE                                                    500
0433                SS=0.000                                                    501
0434                DO 130 J=1,N                                                502
0435                SS=SS+W(K)*X(J)                                             503
0436                K=K+N                                                       504
0437           130  CONTINUE                                                    505
0438                ST=(ST-PAR*SS)/SP                                           506
0439                K=NWI+I                                                     507
0440                DO 131 J=1,MPN                                              508
0441                W(K)=W(K)+ST*W(NWV+J)                                       509
0442                K=K+N                                                       510
0443           131  CONTINUE                                                    511
0444           128  CONTINUE                                                    512
0445                GO TO 64                                                    513
0446                END                                                         514
```

APPENDIX E

ON THE POSSIBILITY OF EXPERIMENTAL SEPARATION OF RESONANCES AND CUSPS FROM BACKGROUND IN ELECTRON SCATTERING

C.Jung * and H.S.Taylor

Department of Chemistry,University of Southern California,
Los Angeles,Calif. 90007 , USA

Subject index: 34.80

* Present address : Fachbereich Physik,Universität,
6750 Kaiserslautern , West Germany

Abstract

It is demonstrated with the help of model calculations that a laser field can be used to supress the background in electron scattering cross sections and to pick out only those parts which are rapidly varying as function of the incoming electron energy.Therefore it becomes possible to measure the pure Breit-Wigner peaks and threshold effects independent of the fact that interference between the resonances and the background in radiationless electron scattering is strong.The results are interpreted within the low frequency limit for free-free transitions.The AC Stark shift of the resonance is also observed.

1. Introduction

Some of the most spectacular physics in electron-atom scattering is in the appearance of resonances and threshold effects.Unfortunately,in many cases these effects are small structures buried in a big background and their shape is usually modified by interference with the background. Sometimes it is difficult to be sure if a measured structure is real or noise and sometimes it is hard to pull the interesting physical parameters out of the measured data. For a review of these effects and their experimental difficulties see ref.1 .If on the other hand,an experimentalist were able,by some method,to supress all the background in the neighborhood of a resonance,then a pure Breit-Wigner peak would be observed and it would be straightforward to determine the position E_R and the width Γ of this resonance. The purpose of this paper is to explain a method of performing an electron scattering experiment in such a way that only those parts of the scattering amplitude lead to a signal,which vary rapidly as function of the incoming electron energy.Thereby we pick resonances and cusps out of the total scattering amplitude and get rid of all background influences.

The basic idea is to perform the electron scattering process within a strong laser field and to collect only those scattered electrons which have emitted or absorbed a certain number of photons.This method has been suggested first in ref.2 in a treatment of free-free transitions in first order perturbation theory and low frequency limit.

It will become clear in this paper that the idea works under much more general circumstances.Processes in which an electron scatters off an atom or molecule and emits or absorbs photons at the same time are known under the name free-free transitions.(For a review of these processes see ref.3).In these processes three systems (electron, target,laser field) interact with each other and this makes a theoretical treatment so complicated that up to now only special cases and approximations have been investigated.If the photon energy is far away from all transition energies of the target states involved in the process then the interaction between the laser and the target can to a high degree of approximation be neglected and the problem reduces to the description of electrons under the simultaneous influence of the target potential and the laser field.To check this the influence of the laser-target interaction on the processes considered here can be estimated by the following considerations:The external field induces an electric dipole moment in the target atom and this dipole field can be felt by the scattered electron and can thereby modify the electron-target interaction potential V.Next let us estimate the order of magnitude of this effect. In most of our calculations a photon energy of 0.005 a.u. is used which is close to the CO_2 laser photon energy of 117meV.Because this photon energy is smaller by a factor 10 to 100 compared to the first excitation energy of the

atomic ground states we use the static polarizability to calculate the induced dipole moment (Thereby we overestimate the effect).For atoms the polarizability is between $2 \cdot 10^{-25}$ cm^3 (in e.s. units) for such a small closed shell atom like Helium and $400 \cdot 10^{-25}$ cm^3 for such a big alkali atom like Cesium.

To see strong signals in free-free transitions we need laser power fluxes in the order of 10^8 Watt/cm^2 and for a CO_2 laser this corresponds to an electric field amplitude of 270000 V/cm or 900 e.s.u..Therefore the induced dipole moment is between $1.8 \cdot 10^{-22}$ e.s.u. and $3.6 \cdot 10^{-20}$ e.s.u. for the various atoms.A scattering electron at a distance of 10^{-8} cm,which is in the order of magnitude of the average distance of an electron in a resonance state from the atomic center,will feel a potential energy E_d of a magnitude between 10^{-3}eV and 10^{-1}eV.In electron scattering dipole effects are usually of importance only at very low energy where the electron-dipole interaction energy E_d is of the same order of magnitude as the kinetic energy E_i of the electron.In the case considered here $E_d/E_i \approx 0.01$ in the worst case.Therefore it seems to be justified to neglect the laser-atom interaction.

This neglect of the laser-target interaction does not of course apply at all to molecules which have vibrational transition energies close to the photon energy.In general, the laser frequency must be far out of resonance with the target system.

In addition the laser field strength considered here is small compared to the internal field in the atoms and therefore it is not necessary to worry about the ionization of the target.

Now even this simplified problem without laser-target interaction has not yet been solved in general.Various treatments have either treated the laser field in perturbation expansion or have used the low frequency approximation.The method used here to supress the background can be interpreted theoretically in the low frequency limit. The relevant ideas of this approximation will be given later in section 3 (For some additional information about more general aspects of the low frequency approximation see refs. 4-7).

To fully appreciate the method it is perhaps better to forget about formal theories and instead to make some simple model calculations in the spirit of a numerical experiment.In section 2 we investigate in this way what may happen in an experiment since up to now the new method has not yet been realized in the laboratory.As such we construct a simple 1-dimensional model,which can be solved in terms of known functions.We do not use the low frequency approximation and we do not make a perturbation expansion of the electron-laser interaction.The approximations we must make are the neglect of the laser-target interaction, mentioned above,the dipole approximation and a cut off of the photon Fock space due to the inability to numerically

handle more than a finite number of photon number states. In practice this is not a problem as all reported results, upon inclusion of further photon states in the model,do not significantly change.

In section 3 we present a theoretical interpretation of our numerical results.Section 4 contains conclusions and final remarks and some suggestions onto how to optimise possible experiments.In the appendix we explain our model in detail and show how we calculate all transition amplitudes in terms of known functions.

We use atomic units for all numbers throughout the whole rest of the paper.

2. Results of the numerical experiments

We take a scattering problem in 1 dimension and represent the target by a 2-state system with an excitation energy $E_{12}=0.6$ .We assume that this target provides a 2-channel square well potential V to the incoming electron (see fig.1). In each figure below we have indicated the special choice of V used.We always start the process with the target in its lower state.If the laser is switched off and the incoming kinetic energy E_i of the electron is below 0.6 then the electron may either be transmitted or reflected without energy change.We denote this probability for transmission by T_{rl} and the probability for reflexion by R_{rl} (rl is an abbreviation for "radiationless").If E_i is above 0.6 the electron may excite the target and leave it in its upper state.We denote the probability for excitation in radiationless transmission by S_{rl} and the probability for excitation in radiationless reflexion by Q_{rl}. S_{rl} and Q_{rl} exist only for $E_i > 0.6$.

If we switch on the laser field,then the electron can, in addition to the above,exchange energy with the field in integer multiples of the photon energy ω.We denote the probabilities for processes in which the electron gains (loses) the energy $N\omega$ from the field by Q_N, R_N, S_N, T_N respectively.$R_N(E_i)$ is the probability that an electron comes in with initial kinetic energy E_i,hits the target,is reflected leaving the target behind in its lower state and flies away with a final kinetic energy $E_f = E_i + N\omega$.

Similarly, $Q_N(E_i)$ is the probability that the electron comes in with energy E_i, hits the target, is reflected leaving the target behind in its upper state and flies away with final energy $E_f = E_i - E_{12} + N\omega$. $T_N(E_i)$ and $S_N(E_i)$ are the probabilities for the corresponding processes in transmission. Of course, $Q_N(E_i)$ and $S_N(E_i)$ only exist for $E_f > 0$ i.e. $E_i + N\omega > E_{12}$. The quantities Q,R,S,T are the 1-dimensional analogs to the differential cross sections in 3 dimensions.

In fig.2 we have chosen a potential V which causes an elastic scattering resonance at an incoming energy of $E_R \approx 0.233$ with a width of $\Gamma \approx 0.0002$. This Γ is in the same order as the natural width of electron-rare gas atom resonances. The energy E_R is far below the threshold at $E_T = 0.6$ and therefore no excitation of the target is possible near this energy. In the top line of fig.2 we show the quantities $T_{r1}(E_i)$ on the left and $R_{r1}(E_i)$ on the right as functions of E_i. Below we show the quantities $T_N(E_i)$ and $R_N(E_i)$ as functions of E_i for several values of N. The photon energy is always $\omega = 0.005$ which is close to the photon energy of a CO_2 laser. The power density of the field is so big that the quantity $\alpha = eA / mc\hbar\omega$ has a magnitude of exactly 1. (e is the electron charge, m is the electron mass, c is the speed of light, A is the amplitude of the vector potential). This is still a moderate power density and therefore only one photon processes give a strong signal. Two photon transitions can just be seen. In fig.3 we show the results for a calculation in which we have increased the laser

power so that $\alpha = 2$ but left all other parameters unchanged.

The important results of these figures are:The elastic resonance causes a whole series of resonances in each R_N and T_N but only a few ones are strong enough so that we can observe them immediately.The distance between two adjacent structures is exactly ω.The strength of the individual structures depends on the laser power.In general we see more resonances if we increase the laser power.The most striking observation is,that in $T_N, N \neq 0$ there is no background at all and the resonances appear as pure Breit-Wigner peaks.In all other cases (i.e. for T_0 and all R_N) the interference of the resonances with the background is either similar to the corresponding radiationless case only with the difference that the relative resonance effect is smaller or the resonance shape is reflected compared to the radiationless case.The best examples for this reflection are in R_{+1} in fig.3.We see that it is easier to determine E_R and Γ out of T_{+1} or T_{-1} than out of T_{r1} or R_{r1}.In fig.2 note that T_N and T_{-N} or R_N and R_{-N} look similar in shape but are shifted by $N\omega$.

In fig.4 we have changed the offdiagonal elements of V so that the width Γ of the elastic resonance is now 0.0012 which is in the order of the resolution of common electron spectrometers.This change in V causes also a small shift in the position E_R of the resonance (see the top line in fig.4).All other parameters are the same as in fig.2.The width Γ is now so big that there are overlaps of the

various resonance structures in R_N and T_N which influence the resonance shapes in general. For $T_N, N \neq 0$ the wings of two adjacent resonance peaks simply add without any visible interference. Again T_N and T_{-N} or R_N and R_{-N} look similar in shape but shifted by $N\omega$. In fig.5 we show the resonance curves for $\alpha = 3$ and otherwise the same parameters as in fig.4. We did not make calculations with another value of ω because the interesting quantity is the ratio between ω and Γ and a increase of ω would cause similar effects as a decrease of Γ.

The only threshold of our 2-channel model is at $E_T = 0.6$ and it is interesting to compare radiationless scattering and free-free transitions close to this energy. Fig.6 shows radiationless scattering in the top line and free-free transitions below. In the drawings T and R are represented by solid lines and S and Q are represented by dotted lines. First we see that in free-free transitions the threshold for S_N and Q_N lies at $E_i = E_T - N\omega$. This is easy to understand because the energy which the electron gains/losses from the laser field goes into the energy balance for the excitation of the target by the electron impact and the new channel opens as soon as the final kinetic energy of the electron is above zero i.e. as soon as $E_i + N\omega - E_T > 0$ or $E_i > E_T - N\omega$. In T_{-1} we see a plateau between E_T and $E_T + \omega$ and in T_{+1} a plateau between $E_T - \omega$ and E_T with a sharp drop on both sides. In addition there are smaller threshold effects shifted by ω away from the sharp drops. In T_{-2} and T_{+2} we see a sharp peak at $E_T + \omega$ or $E_T - \omega$

respectively and again smaller effects shifted away by ω. In the other cases (T_0 and all R_N) the threshold effects occur at various $E_i=E_T + n\omega$ and their shape is either of the same type as in the corresponding radiationless scattering or it is just turned upside down.In S_N and Q_N there are only very weak structures,too weak to be seen clearly in fig.6.

In fig.7 we see results for $\alpha=2$ and otherwise the same parameters as in fig.6.Now the threshold effects are distributed over more energy values and for example in T_{-1} the main drop on both sides of the plateau is shifted outwards by ω on both sides compared to fig.6.In the picture for Q_0 we see an "accidential" zero near $E_i=0.607$ which will be explained below.In fig.8 we see results for $\omega=0.002$ and otherwise the same parameters as in fig.6. All curves look nearly the same as in fig.6 only the energy scale is changed by a factor 2.5 which is the ratio between the photon energies in the two examples.

The most important results for the thresholds are in summary:A sudden drop (or increase) in the radiationless scattering as function of the incoming energy produces in the free-free probabilities T_{+1} and T_{-1} a plateau near the threshold energy and zero signal otherwise.At moderate laser power the length of the main structure is exactly ω. Thereby the presense or absense of a threshold makes a 100% change in the relative signal strength but the absolute signal strength is always very low.Depending on

the values of ω and α the strong contributions to the threshold effects in T_N appear at different energies.

3.Interpretation of the numerical results

A theoretical explanation of the results of section 2 can be given most easily within the low frequency approximation and therefore let us first explain its basic idea. In order to be more general we give all formulas for the 3-dimensional case and consider the numerical results from section 2 as the special cases of the scattering angles 0 and π .If the laser wavelength is very long compared to atomic distances,then we can separate the free-free transition into three steps.First a free electron moves within the laser field and can virtually emit and absorb photons.Let us assume that the laser field is single mode and in the pure number state $|N\rangle$ in absense of the electron.We denote the state of a free electron moving with momentum $\vec{p}$ by $|\vec{p}\rangle$ .If we neglect photon depletion effects and use the dipole approximation,then the exact state of the electron in the field is given by (see appendix)

$$|\phi_{\vec{p},N}\rangle = \sum_n J_n(\vec{\alpha}\cdot\vec{p}\,)\,|\vec{p}\rangle|N+n\rangle \qquad (1)$$

J_n is the Bessel function of first kind and order n. $\vec{\alpha} = eA\vec{\varepsilon} / mc\hbar\omega$ where $\vec{\varepsilon}$ is the polarization vector of the laser field.This $\vec{\alpha}$ is the 3-dimensional generalization of the α given above in section 2.

The Bessel functions $J_n(\vec{\alpha}\,\vec{p})$ can be viewed as interaction coefficients for virtual n photon absorbtion/emission at absense of a target.

In dipole approximation there is no recoil of the electron during emission or absorbtion of a photon and therefore the electron momentum does not depend on how many photons the electron has absorbed or emitted i.e. $|\vec{p}\rangle$ in (1) is independent of n.The energy of the electron is different in its various states and the electron wave connected with the photon number state $|N+n\rangle$ is at energy $E=p^2/2m - n\omega$.Only the n=0 term is on the energy shell and all other terms are off shell.

Then in the second step of the free-free transition this mixture of electron waves hits the target and is scattered. The main idea of the low frequency approximation is to neglect the laser-electron interaction during this second step and therefore the electron-target scattering is described by a radiationless scattering amplitude.But according to what was said above each term of the sum in (1) is scattered at its particular intermediate energy $E_i - n\omega$ and its scattering is therefore described by a scattering amplitude at this shifted energy.

In the third step the scattered electron waves interact again with the laser and evolve into their final states.

Along these lines the following formula (3) has been derived first in ref.8 and confirmed later in ref.9 by another derivation.This formula gives the scattering

amplitude $f_N(E_i, \vartheta)$ for an electron to come in with kinetic energy E_i, be scattered by an angle ϑ and to have a final kinetic energy E_f of

$$E_f = E_i + N\omega \tag{2}$$

$$f_N(E_i, \vartheta) = \sum_k J_{N-k}(\vec{\alpha}\,\vec{p}_f)\; f_{rl}(E_i + k\omega, \vartheta)\; J_k(-\vec{\alpha}\,\vec{p}_i) \tag{3}$$

f_{rl} is the corresponding amplitude for radiationless electron-target scattering. The three factors correspond to the three physical stages above. An analoguous formula holds for scattering with excitation of the target if we take on the r.h.s. of (3) the corresponding scattering amplitude for radiationless excitation of the target and take the excitation energy into the energy balance (2).

Now let us decompose f_{rl} in (3) into a resonance part and a background part

$$f_{rl}(E_i + k\omega, \vartheta) = f^R_{rl}(E_i + k\omega, \vartheta) + f^{BG}_{rl}(E_i + k\omega, \vartheta) \tag{4}$$

The Bessel functions decrease rapidly for increasing order and fixed argument as soon as the absolute value of the order becomes larger than the absolute value of the argument. Therefore only a few k give a strong contribution to the k sum in (3). Since f^{BG}_{rl} depends only weakly on its energy argument then we can neglect this dependence, take all f^{BG}_{rl} at the energy E_i, pull f^{BG}_{rl} out of the k sum and apply the addition theorem of the Bessel functions and find

$$f_N(E_i, \vartheta) = J_N(\vec{\alpha}(\vec{p}_f - \vec{p}_i))\; f^{BG}_{rl}(E_i, \vartheta) + \sum_k J_{N-k}(\vec{\alpha}\,\vec{p}_f) f^R_{rl}(E_i + k\omega, \vartheta) J_k(-\vec{\alpha}\,\vec{p}_i) \tag{5}$$

Our model calculation is independent of the low frequency

approximation and so a comparison between the exact numerical results of sec.2 and (5) can be viewed as test of (5).

The present state of the experimental verification of (5) is as follows:In ref.10 $|f_N|^2$ has been measured as function of N for several values of $\vartheta, E_i, \vec{\varepsilon}$.All results are in qualitative agreement with (5).It has not been possible to check for quantitative agreement because the exact laser power distribution in space and time has not been known. For quantitative calculations its knowledge would be absolutely necessary since multiphoton transitions are a nonlinear effect and depend essentially on the exact power distribution in space and time and not only on the average power (For these problems see also ref.11).In ref.10 resonances did not play any role and therefore only the first term in (5) has been measured.From such a measurement we cannot,in contrast to the present work,learn anything new about electron-atom scattering because the result is essentially the product of the elastic background cross section and a Bessel function - both well known quantities.

In refs.12,13 $|f_{-1}|^2$ has been measured as function of E_i near the argon resonances at 11eV but unfortunately only in backward direction for the one scattering angle $\vartheta = 160^\circ$.Since a weak laser has been used only the resonance structures at E_R and $E_R + \omega$ could be seen.Their shape and strength is in full agreement with (5).

For scattering without excitation of the target $|p_i| \approx |p_f|$ since $\omega \ll E_i$ and therefore in foreward direction we find

$\vec{\alpha}(\vec{p_f}-\vec{p_i})\approx 0$. The background term in (5) disappears in foreward direction for $N\neq 0$ because $J_N(0)=\delta_{N,0}$. This is the explanation why we did not see a background contribution for $T_N, N\neq 0$ in the model calculations. f^R_{rl} produces a resonance structure when its energy argument is at E_R and therefore we get a resonance structure in the k-sum in (5) every time when $E_i+k\omega=E_R$ for a $k\in\mathbb{Z}$. The magnitude of the resonance effect at $E_i=E_R-k\omega$ depends on the exact value of the two Bessel functions in the corresponding term in (5). If $\Gamma\ll\omega$ we get a series of well separated resonances with an energy spacing ω between adjacent peaks. This is exactly what we saw in figs.2-5.

In backward direction $\vec{p_f}\approx-\vec{p_i}$ and in general $J_N(\vec{\alpha}(\vec{p_f}-\vec{p_i}))$ is not near a zero and the background does not disappear. Depending on the relative signs of the various Bessel functions the relative phase between the resonance term and the background term is either the same as in the radiationless case or it is just shifted by π. Therefore the resonance shape is either the same as in the radiationless case or it is just reflected. The magnitude of the relative resonance effects depends again on the exact values of the corresponding Bessel functions and thereby on the parameters E_i, α. We saw that T_N and T_{-N} or R_N and R_{-N} look nearly the same in shape but are only shifted by $N\omega$. This can be explained by the fact that $J_N=(-1)^N J_{-N}$.

One effect which is not contained in (5) is the AC Stark shift of the resonances. Let us look at fig.9.

In this figure we show resonance structures in T_N and R_N at various initial energies $E_i = E_R + k\omega$ and for various values of ω, α and N. The potential V is the same as in figs. 2 and 3. In each picture we show the low frequency approximation of (3) as broken line and the exact numerical result as solid line. We see a shift of the resonance structures which does not depend on E_i, N and ϑ but depends on ω and α. Besides this shift the low frequency limit gives the correct shape and height for all resonance peaks. The fact that the shift does not depend on the particular peak at which we look i.e. does not depend on E_i, N and ϑ is an indication that the resonance itself is shifted and not only its appearance in the free-free transitions. The Stark shift goes quadratically with ω and quadratically with α. This is in exact agreement with the results of ref. 9. In our examples in fig. 9 the shift $\Delta E_R \approx -0.25\, \omega^2 \alpha^2$. According to ref. 9 the proportionality constant between ΔE_R and $\omega^2 \alpha^2$ is a measure of the laser induced coupling between the resonance wave function φ_2 in channel 2 and the continuum wave function φ_1 in channel 1. Therefore a measurement of $\Delta E_R / \omega^2 \alpha^2$ could give experimental information about the dipole matrix element $\langle \varphi_1 | \vec{\varepsilon}\, \vec{p} | \varphi_2 \rangle$.

We could not see a change of Γ with increasing laser power. For this extra width caused by the Stark decay to become important much higher laser powers would be necessary.

Now let us try to explain the threshold effects in figs.6-8 by making a quite crude model for a threshold effect in radiationless scattering and representing the amplitude for scattering without target excitation by

$$f_{rl}(E_i,\vartheta)=c(\vartheta)+\Theta(E_i-E_T)\ d(\vartheta) \qquad (6)$$

where c and d do not depend on E_i and Θ is the unit step function.If we insert (6) into (5) we get

$$f_N(E_i,\vartheta)=c(\vartheta)\ J_N(\vec{\alpha}(\vec{p_f}-\vec{p_i}))\ +$$
$$d(\vartheta)\ \sum_k J_{N-k}(\vec{\alpha}\vec{p_f})\ \Theta(E_i+k\omega-E_T)\ J_k(-\vec{\alpha}\vec{p_i}) \qquad (7)$$

For $\vec{p_f}\approx\vec{p_i}$ and $N\neq 0$ the first term drops.The second term produces a step at each $E_i=E_T-k\omega, k\in\mathbb{Z}$ .The relative strength of the various steps depends on the exact values of the Bessel functions and thereby on the values of the parameters E_i and α.

If we set $\vartheta=0$ in (7) and assume a moderate laser power for which only first order contributions are important, we obtain

$$f_{-1}(E_i,0)=\frac{1}{2}\alpha|p|\ d(0)\left\{\Theta(E_i-E_T)-\Theta(E_i-\omega-E_T)\right\}$$

This is just a plateau between E_T and $E_T+\omega$ and it is zero otherwise.An analoguous calculation applies to T_{+1} and gives a plateau between $E_T-\omega$ and E_T.T_{rl} in fig.6 does not have a sharp step at E_T but a rounded one and therefore also T_{+1} and T_{-1} have a rounded drop off on both sides of the plateau.

We see additional smaller threshold effects at other E_i which come from higher order contributions.For higher laser powers as e.g. in fig.7 higher order contributions

may become more important than first order contributions and therefore the steps at other E_i become more important.

If we look at T_{-2} and take only second order contributions of (7) we get

$$f_{-2}(E_i,0)=d(0)\,\frac{1}{8}\alpha^2 p^2\left\{\theta(E_i-E_T)-2\,\theta(E_i-E_T-\omega)+\theta(E_i-E_T-2\omega)\right\}$$

This is a positive plateau between E_T and $E_T+\omega$ and a negative plateau of the same absolute value between $E_T+\omega$ and $E_T+2\omega$ and it is 0 otherwise. $T_{-2}=\frac{p_f}{p_i}\left|f_{-2}(E_i,0)\right|^2$ is then just a plateau between E_T and $E_T+2\omega$. The step in T_{rl} in fig.6 is rounded and therefore a wedge remains at $E_T+\omega$ in T_{-2} in fig.6. This wedge is a left over between two rounded cut offs on both sides. For higher power density in fig.7 again additional threshold effects at other E_i become important in T_{-2}. An analoguous reasoning applies to T_{+2}.

For T_0 and R_N both terms in (7) contribute and the relative phase between both terms is either the same as in the radiationless scattering or it is shifted by π. Therefore the shape of the threshold structures is either the same as in the radiationless case or it is just turned upside down as e.g. in R_0 in fig.6 at $E_T\pm\omega$.

For $E_i=0.607$, $\alpha=2$, $N=0$, $\vartheta=\pi$ and excitation of the target the quantity $\vec{\alpha}(\vec{p_f}-\vec{p_i})$ is just at the first zero of the Bessel function $J_0(\vec{\alpha}(\vec{p_f}-\vec{p_i}))$ and this explains the accidental zero of Q_0 in fig.7.

4. Discussions and conclusions

We have seen that the low frequency approximation given in (3) explains all qualitative features of resonances and thresholds in free-free transitions except the AC Stark shift.The quantitative error of the low frequency approximation - corrected for the AC Stark shift - has always been below a few percent (see fig.9).But the dependence of this error on the various parameters will probably depend on the dimension of the space.Therefore it would not be of general interest to add here a detailed investigation of the numerical error of the low frequency approximation in our 1-dimensional model.

What advice can we give to an experimentalist,who wants to utilize free-free transitions in order to look for resonances and cusps in electron-atom scattering?

The most interesting property of free-free transitions shown in this paper is that only rapidly varying parts of the radiationless scattering causes any free-free signal in foreward direction.This fact may be used in an experiment to project out resonances and threshold effects.The price to pay for this removal of the background is a reduction of the number of electrons which contribute to the signal.As shown in figs.2-5 the ratio between the resonance height in T_N,$N\neq 0$ and in T_{rl} is 0.1 in the most

favorable cases.

In an experiment it is not possible to measure at an angle of exactly $\vartheta=0$. By proper adjustment of the laser polarization vector $\vec{\varepsilon}$ it is possible to fulfil the relation

$$\vec{\varepsilon}(\vec{p}_f-\vec{p}_i)=0 \qquad (8)$$

for any choice of $\vec{p}_i$ and $\vec{p}_f$, just turn $\vec{\varepsilon}$ perpendicular to the momentum transfer $\vec{q}=\vec{p}_f-\vec{p}_i$. At the same time we want to see a strong resonance signal and according to (5) $\vec{\varepsilon}\cdot\vec{p}_f$ and $\vec{\varepsilon}\,\vec{p}_i$ should be big enough to give a big value for the Bessel functions. Therefore, $\vec{\varepsilon}$ should be as parallel to $\vec{p}_f$ and $\vec{p}_i$ as possible. All these conditions can be met best if $\vec{p}_f$ and $\vec{p}_i$ are as parallel to each other as the construction of the electron spectrometer allows and if $\vec{\varepsilon}$ is in the direction exactly in between $\vec{p}_f$ and $\vec{p}_i$.

As indicated by the figs. 2-5 resonance signals are biggest and clearest in T_{+2} or T_{-2} at $E_i=E_R-\omega$ or $E_i=E_R+\omega$ respectively if the laser power density is so big that $\vec{\alpha}\vec{p}_f$ and $\vec{\alpha}\vec{p}_i$ are in between 1.5 and 2.2 i.e. if $J_1(\vec{\alpha}\vec{p})$ is near its first maximum. In this case the resonance signal comes from the term $J_{\pm1}(\vec{\alpha}\vec{p}_f)\, f_{r1}(E_i\pm\omega,\vartheta)\, J_{\pm1}(\vec{\alpha}\vec{p}_i)$ in (5). This means: The electron comes in shifted by one photons energy away from the resonance energy. Then it absorbs/emits one photon and can go into the resonance state. The resonance decays and the outgoing electron absorbs/emits another photon on its way out. For this choice of the parameters the resonance signal in $T_{\pm2}$ is about 10% of the resonance signal in the radiationless scattering, which is the best one

can hope for.In addition,for these values of the parameters the other resonance structures in $T_{\pm 2}$ (i.e. the ones at $E_R, E_R \pm 2\omega, E_R \pm 3\omega$, etc.) are quite small and this decreases the sources of confusion if several elastic resonances occur close together.

For the investigation of thresholds choose a moderate power density of the laser so that only first order contributions give a strong signal to the free-free amplitudes and look at T_{+1} or T_{-1} for structures of the length ω.Two sharp drops in T_{+1} at $E_a - \omega$ and E_a and a smooth behavior in between indicate a threshold at E_a.A similar structure in T_{-1} between E_a and $E_a + \omega$ indicates the same threshold at E_a.

As indicated in figs.6-8 the absolute signal in $T_{N,N\neq 0}$ is extremely weak but the relative threshold effect is 1 and can therefore be seen clearly.This is in sharp contrast to elastic scattering where the relative threshold effects are generally small.Usually these effects are observed in electron impact excitation,where their relative effect is bigger than in the elastic channel.Our idea is unusual in so far that it filters out threshold effects in a process which is elastic with respect to the electron-target interaction.The big disadvantage of our method is the extremely small absolute size of the signal and there is the possibility that in a real experiment machine noise will bury the weak threshold signal and make it impossible to utilize our idea in the laboratory.

In any case for resonances or thresholds the photon energy should be chosen larger than the energy interval over which the radiationless scattering varies rapidly in order to avoid confusing overlaps of the structures in free-free transitions.

Can "accidental" zeros like the one in Q_0 shown in fig.7 (i.e. those values of parameters for which $\vec{\alpha}(\vec{p_f}-\vec{p_i})$ is at a zero of some Bessel function) used for anything?We don't think so,because their position depends strongly on the exact laser power and in an experiment the laser power varies in time and space and the electrons,collected in the detector,have experienced quite different laser powers during their scattering process.Therefore,such an accidental zero would be completely smeared out in any real experiment.

There is also the interesting possibility that a free-free experiment might be used to probe the laser field strength.In ref.11 it has been shown how the results of a measurement of $|f_N(E_i,\vartheta)|^2$ as function of N can be used to calculate the average power density of the laser field in which the experiment has been performed.It is probably possible to generalize the results of ref.11 and to work backwards to even more detailed information about the laser field.

Appendix

In this section we explain our model in detail and show how we calculate the amplitudes for all processes.

We work in a 1-dimensional space and choose x as space coordinate.We describe the electron-target interaction by a 2-channel square well potential of the form

$$\mathbb{V}(x) = \begin{pmatrix} V_{11} & V_{12} \\ V_{21} & V_{22} \end{pmatrix} \cdot \Theta(R - |x|) \qquad (A1)$$

$V_{12}=V_{21} \in \mathbb{R}$ so that $\mathbb{V}$ is a self adjoint operator.
Θ is the unit step function.The matrix of excitation energies is

$$\mathbb{B} = \begin{pmatrix} 0 & 0 \\ 0 & E_{12} \end{pmatrix} \qquad (A2)$$

We cut the x-axis into the three intervals
$I_1=(+R,\infty)$, $I_2=(-R,+R)$, $I_3=(-\infty,-R)$
The target is in its ground state initially and the electron comes in with momentum k_{in}.Then the matrix of the electron momentum in intervalls I_1 and I_3 is

$$\mathbb{K} = \begin{pmatrix} k_o & 0 \\ 0 & h_o \end{pmatrix} = \begin{pmatrix} k_{in} & 0 \\ 0 & \sqrt{k_{in}^2 - 2mE_{12}} \end{pmatrix} \qquad (A3)$$

In I_2 we set $\mathbb{M} = \mathbb{B} + \mathbb{V}$ and construct the orthogonal matrix $\mathbb{U}$ which diagonalizes $\mathbb{M}$ according to

$\mathbb{U}^+ \mathbb{M} \mathbb{U} = \Sigma$ with

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{pmatrix} \qquad (A4)$$

$\left[k_{in}^2 \mathbb{1} - 2m \Sigma\right]^{\frac{1}{2}}$ is the diagonalized momentum matrix in I_2.
We assume a single mode laser field and use the dipole approximation.

The total Hamiltonian $\mathbb{H}$ for the motion of the electron under the simultaneous influence of the target and the laser field is

$$\mathbb{H} = \left\{ \frac{p^2}{2m} + \hbar\omega a^+ a - \frac{ep}{mc}\beta(a^+ + a) \right\} \mathbb{1} + \mathbb{B} + \mathbb{V}(x) \qquad (A5)$$

β is an abbreviation for $\beta = (2\pi c^2 / \omega L^3)^{\frac{1}{2}}$ where L^3 is the quantization volume of the electromagnetic field. p is the electron momentum operator, a and a^+ are the annihilation and creation operators for a laser photon. We write $|N\rangle$ for an eigenstate of a^+a with eigenvalue $N \in \mathbb{N}$.

First let us look at eigenfunctions of $\mathbb{H}$ in the intervalls I_1 and I_3. There $\mathbb{H}$ is diagonal with respect to the target states. We neglect photon depletion effects in the sense that we set $\sqrt{N+n} = \sqrt{N}$ in the coupling strength between various photon number states, where N is the initial number of photons in the laser beam and N+n is the number of photons in intermediate or final states. Using the recursion formula of the Bessel functions

$$2n\, J_n(y) = y \left\{ J_{n-1}(y) + J_{n+1}(y) \right\} \qquad (A6)$$

we see that functions of the form

$$\varphi(k_o, x, N) = \exp (ik_o x) \sum_n J_n(\alpha k_o) \, |N+n\rangle \qquad (A7)$$

are eigenfunctions of the operator

$$\frac{p^2}{2m} + \hbar\omega a^+ a - \frac{ep}{mc} \beta (a^+ + a) \qquad (A8)$$

with eigenvalue

$$N\omega + k_o^2 / 2m \qquad (A9)$$

α is an abbreviation for $\alpha = e\sqrt{N}2\beta / mc\hbar\omega$.
The quantity $2\beta\sqrt{N}$ is the amplitude of the corresponding classical vector potential A of the laser field. Note that k_o is independent of n in (A7) because recoil effects of the electron are neglected in dipole approximation. In order to start from the most general eigenfunctions of H we must consider that the functions φ in (A7) are degenerate in two ways:

1.: φ remains an eigenfunction of operator (A8) to the same energy (A9) if we reverse k_o i.e. if we replace k_o by $-k_o$.

2.: We get an eigenfunction to the same energy if we replace N by N+L and simultaneously replace k_o by k_L where $k_L^2 / 2m + L\omega = k_o^2 / 2m$

Therefore the most general eigenfunction of H with eigenvalue E in the intervall I_1 is the 2 component column vector

$$\Phi_1(E,x) = \sum_L \begin{pmatrix} r_{1,L}\,\varphi(k_L,x,L+N) + w_{1,L}\,\varphi(-k_L,x,L+N) \\ r_{2,L}\,\varphi(h_L,x,L+N) + w_{2,L}\,\varphi(-h_L,x,L+N) \end{pmatrix} \qquad (A10)$$

where all k_L and h_L are given by

$$E = k_L^2 / 2m + (N+L)\omega = h_L^2 / 2m + (N+L)\omega + E_{12} \qquad (A11)$$

$r_{i,L}$ and $w_{i,L}$ are arbitrary complex constants to be fixed later by boundary conditions.

In a completely analoguous way we find for the most general eigenfunction of H with eigenvalue E in interval I_3

$$\Phi_3(E,x) = \sum_L \begin{pmatrix} j_{1,L}\ \varphi(k_L,x,L+N) + & t_{1,L}\ \varphi(-k_L,x,L+N) \\ j_{2,L}\ \varphi(h_L,x,L+N) + & t_{2,L}\ \varphi(-h_L,x,L+N) \end{pmatrix} \qquad (A12)$$

$j_{i,L}$ and $t_{i,L}$ are again constants to be fixed later by boundary conditions.

In the interval I_2 H is not diagonal with respect to target states but $U^+HU = \tilde{H}$ is and therefore we first construct eigenfunctions to $\tilde{H}$. In the same way as before we find for the most general eigenfunction of $\tilde{H}$ with eigenvalue E

$$\tilde{\Phi}_2(E,x) = \sum_L \begin{pmatrix} a_{1,L}\ \varphi(\lambda_L,x,L+N) + & b_{1,L}\ \varphi(-\lambda_L,x,L+N) \\ a_{2,L}\ \varphi(\mu_L,x,L+N) + & b_{2,L}\ \varphi(-\mu_L,x,L+N) \end{pmatrix} \qquad (A13)$$

where $$\lambda_L^2 / 2m + (N+L)\omega + \sigma_1 = E = \mu_L^2 / 2m + (N+L)\omega + \sigma_2 \qquad (A14)$$

with σ_1 and σ_2 given in (A4).

$a_{i,L}$ and $b_{i,L}$ are again constants to be fixed later by boundary conditions.

Because of $H U \tilde{\Phi}_2 = U \tilde{H} \tilde{\Phi}_2 = E\ U \tilde{\Phi}_2$

we see that $\Phi_2 = U\tilde{\Phi}_2$ is an eigenfunction of H with eigenvalue E in interval I_2.

At $x=+\infty$ we choose the boundary condition that there is only one incoming wave with the target in state 1 and the laser and electron in state $\varphi(-k_{in},x,N)$ i.e.

$$w_{i,L} = \delta_{i,1} \cdot \delta_{L,0} \qquad (A15)$$

At $x=-\infty$ we choose the boundary condition that there is no incoming wave at all from the left i.e.

$$j_{i,L} = 0 \text{ for all i and all L} \qquad (A16)$$

At $x=+R$ and $x=-R$ we require that the wavefunction Φ is

continuous and has a continuous first derivative.This gives 4 equations for 2-component column functions or 8 equations for linear combinations of photon number states. These equations must be fulfilled for the coefficients of each photon state separately and therefore each photon state gives us 8 equations which connect the constants $a_{i,L}, b_{i,L}, t_{i,L}, r_{i,L}$ (i=1,2).Some of these equations are inhomogeneous.

Al together,we get an infinite inhomogeneous system of linear equations to determine the free constants.In our model calculations we could only handle a finite number of equations with a finite number of unknowns and therefore we had to cut off the system of equations.Out of the coefficient matrix of the system of linear equations we have cut out the $(2l+1)\cdot 8 \times (2l+1)\cdot 8$ matrix centered at the 8×8 block which comes from the L=0 terms in (A10,A12, A13) and the n=0 terms in (A7).Accordingly we have cut out of the inhomogenity vector a (2l+1)8 component piece centered at the block which comes from the n=0 terms in (A7).Then we have solved these (2l+1)8 coupled linear equations.

We have found rapid convergence of the results with increasing l values as soon as $l > |\alpha p|$.For all calculations shown in figs.2-9 it has been sufficient to choose a l value between 5 and 10.The rapid convergence can be understood from the fact that Bessel functions decrease rapidly as soon as the absolute value of the order becomes larger than

the absolute value of the argument.If we take a larger value of l,then we include coefficients which contain Bessel functions of higher orders.

As a last step we calculate the quantities

$$R_L = \frac{k_L}{k_{in}} \left| r_{1,L} \right|^2 \quad , \quad Q_L = \frac{h_L}{k_{in}} \left| r_{2,L} \right|^2$$

$$T_L = \frac{k_L}{k_{in}} \left| t_{1,L} \right|^2 \quad , \quad S_L = \frac{h_L}{k_{in}} \left| t_{2,L} \right|^2$$

These are the quantities plotted in figs.2-9 as function of the incoming electron energy E_i.

All these calculations can also be performed for the case that the target is in state 2 initially.The only change is to replace $\mathbb{K}$ of (A3) by

$$\mathbb{K} = \begin{pmatrix} k_o & 0 \\ 0 & h_o \end{pmatrix} = \begin{pmatrix} \sqrt{k_{in}^2 + 2mE_{12}} & 0 \\ 0 & k_{in} \end{pmatrix}$$

and to replace (A15) by

$$w_{i,L} = \delta_{i,2}\, \delta_{L,0}$$

Acknowledgment

This work has been supported by NSF Grant CHE 76-15656 A02 and ONR Grant N000-14-77-C-0102. We thank Prof.J.Hinze and the Zentrum für interdisziplinäre Forschung of the Universität Bielefeld for their hospitality while this work was being completed.

References

1.G.Schulz, Rev.Mod.Phys 45,378 (1973)

2.C.Jung and H.Krüger, Z.Physik A287,7 (1978)

3.M.Gavrila and M.Van der Wiel, Comments At.Mol.Phys 8,1 (1978)

4.F.Low, Phys.Rev. 110,974 (1958)

5.L.Heller, Phys.Rev. 174,1580 (1968)

6.M.Mittleman, Phys.Rev.A21,79 (1980)

7.L.Rosenberg, Phys.Rev.A21,1939 (1980)

8.H.Krüger and C.Jung, Phys.Rev.A17,1706 (1978)

9.M.Mittleman, Phys.Rev.A20,1965 (1979)

10.A.Weingartshofer,E.Clarke,J.Holmes and C.Jung,
Phys.Rev.A19,2371 (1979)

11.C.Jung, Phys.Rev.A21,408 (1980)

12.D.Andrick and L.Langhans, J.Phys.B 11,2355 (1978)

13.L.Langhans, J.Phys.B 11,2361 (1978)

Figure captions

Fig.1:Plot of the diagonal elements of the model potential for electron-target interaction used in all numerical calculations.The offdiagonal elements $V_{12}=V_{21}$ (not plotted in fig.1) are of the same shape as V_{11} but of different depth.Bound states of the 1-channel potential $V_{22}+E_{12}$ become Feshbach type resonances of the full coupled 2-channel potential.Their width Γ is determined by the magnitude of V_{12}.For small V_{12} we find $\Gamma \propto (V_{12})^2$.See the two resonances in the upper lines of fig.2 and fig.4. A threshold effect occurs at the rim of the upper well i.e. at $E=E_{12}$.

Fig.2:Plots of resonance structures in radiationless scattering and in free-free transitions.For more explanations see main text.

Fig.3:Plots of resonance structures in free-free transitions.For more explanations see main text.

Fig.4:Plots of resonance structures in radiationless scattering and in free-free transitions.For more explanations see main text.

Fig.5:Plots of resonance structures in free-free transitions.For more explanations see main text.

Fig.6:Plots of threshold structures in radiationless scattering and in free-free transitions.For more explanations see main text.

Fig.7:Plots of threshold structures in free-free transitions.For more explanations see main text.

Fig.8:Plots of threshold structures in free-free transitions.For more explanations see main text.

Fig.9:Some examples for the AC Stark shift of resonances in free-free transitions.The solid lines are the results of numerical calculations.The broken lines are the results of eq.(3).For more explanations see main text.

Fig. 1

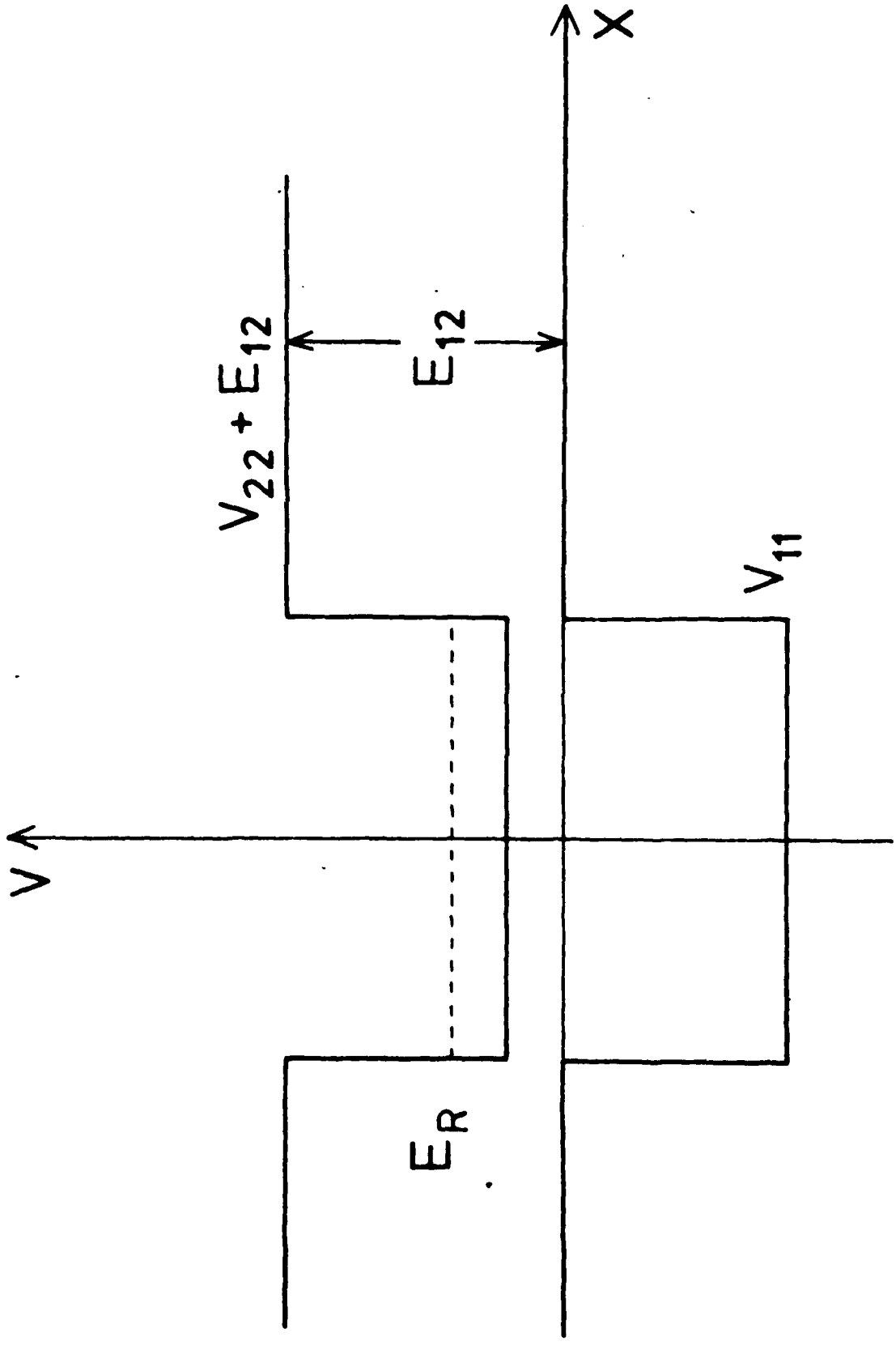


Fig. 2

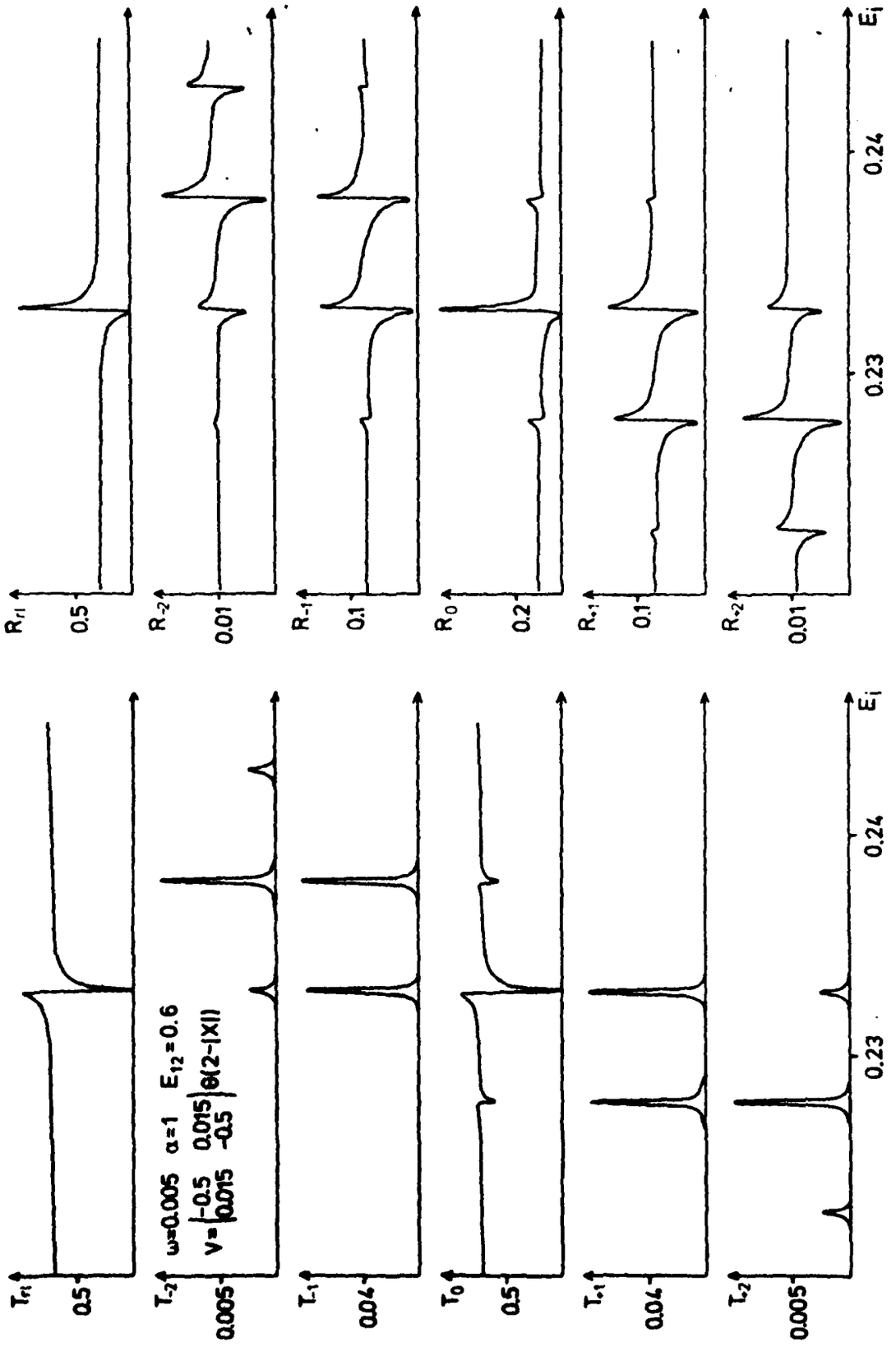

T_{r1}
0.5
T_{-2}
0.005
T_{-1}
0.04
T_0
0.5
T_{+1}
0.04
T_{+2}
0.005
ω=0.005 α=1 E_{12}=0.6
V= |-0.5 0.015 / 0.015 -0.5| θ(2-|X|)
0.23
0.24
E_i
R_{r1}
0.5
R_{-2}
0.01
R_{-1}
0.1
R_0
0.2
R_{+1}
0.1
R_{+2}
0.01
0.23
0.24
E_i

Fig. 3

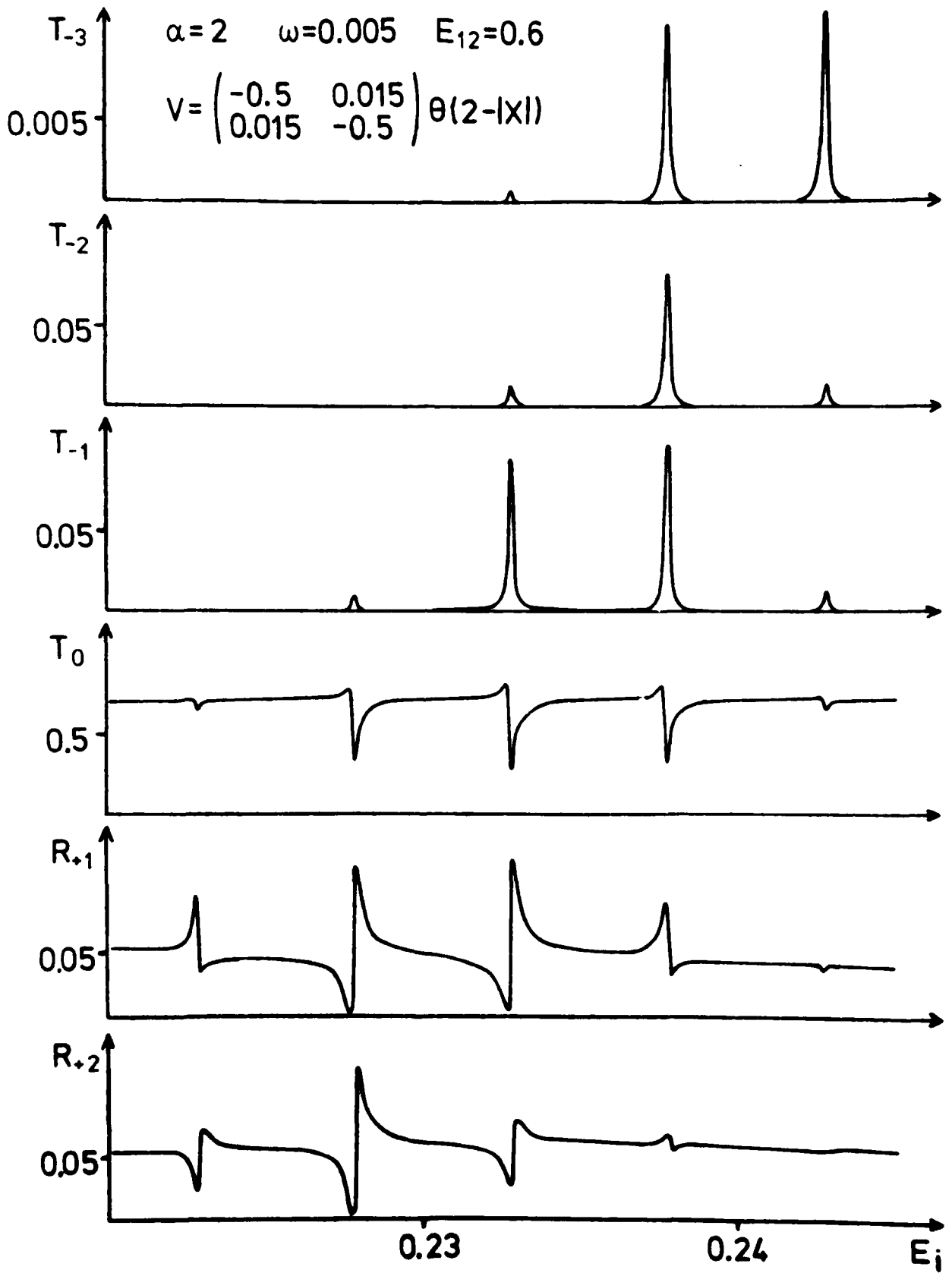

T-3
α = 2
ω=0.005
E12=0.6
V = (-0.5 0.015 / 0.015 -0.5) θ(2-|x|)
0.005
T-2
0.05
T-1
0.05
T0
0.5
R+1
0.05
R+2
0.05
0.23
0.24
Ei

Fig. 4

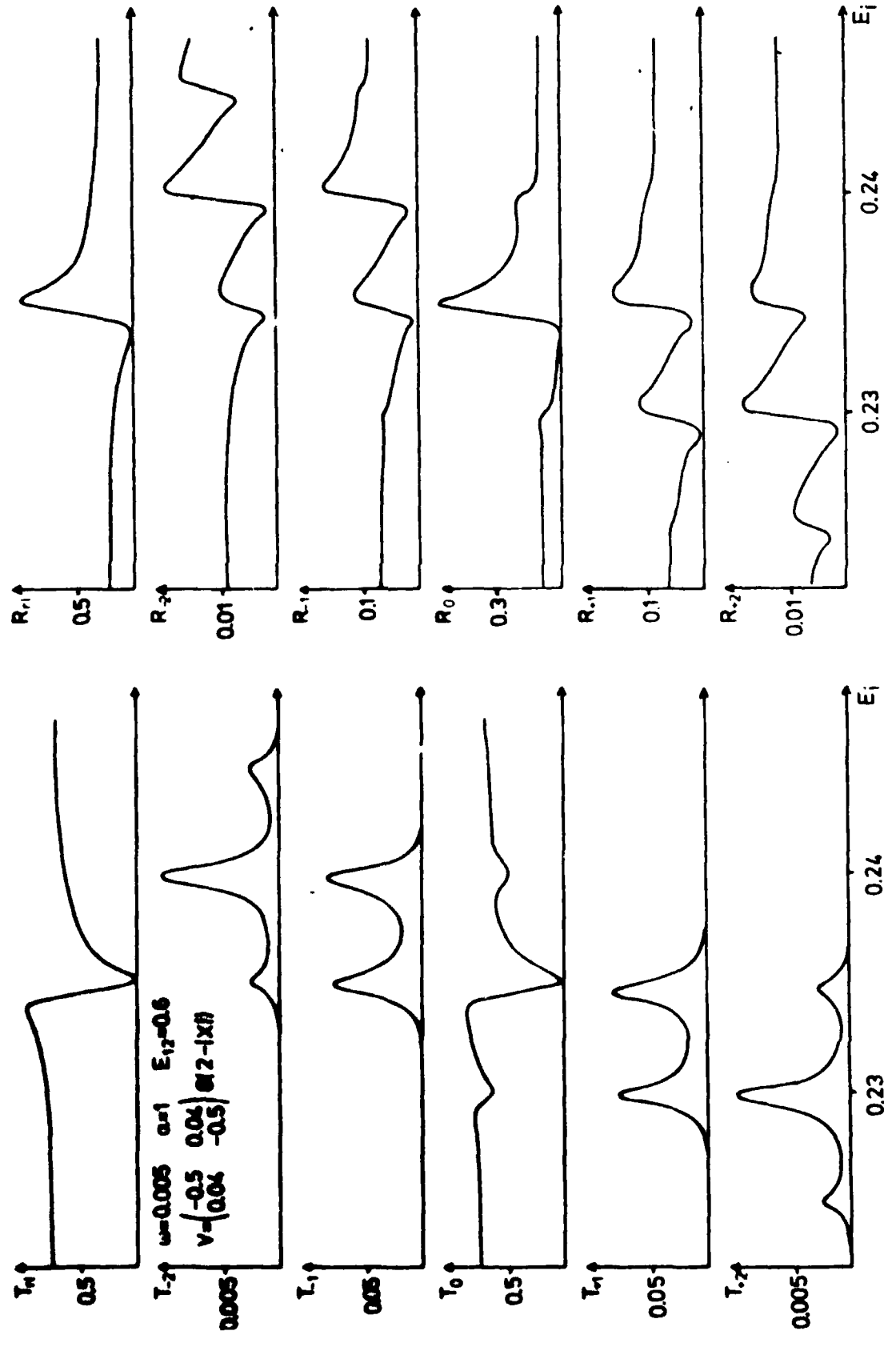

R+1
0.5
R+2
0.01
R-1
0.1
R0
0.3
R+1
0.1
R+2
0.01
Ei
0.23
0.24
T+1
0.5
T+2
0.005
ω=0.005 α=1 E12=0.6
v=(-0.5 0.04 / 0.04 -0.5) (α(2-|x|))
T-1
0.05
T0
0.5
T+1
0.05
T+2
0.005
Ei
0.23
0.24

Fig. 5

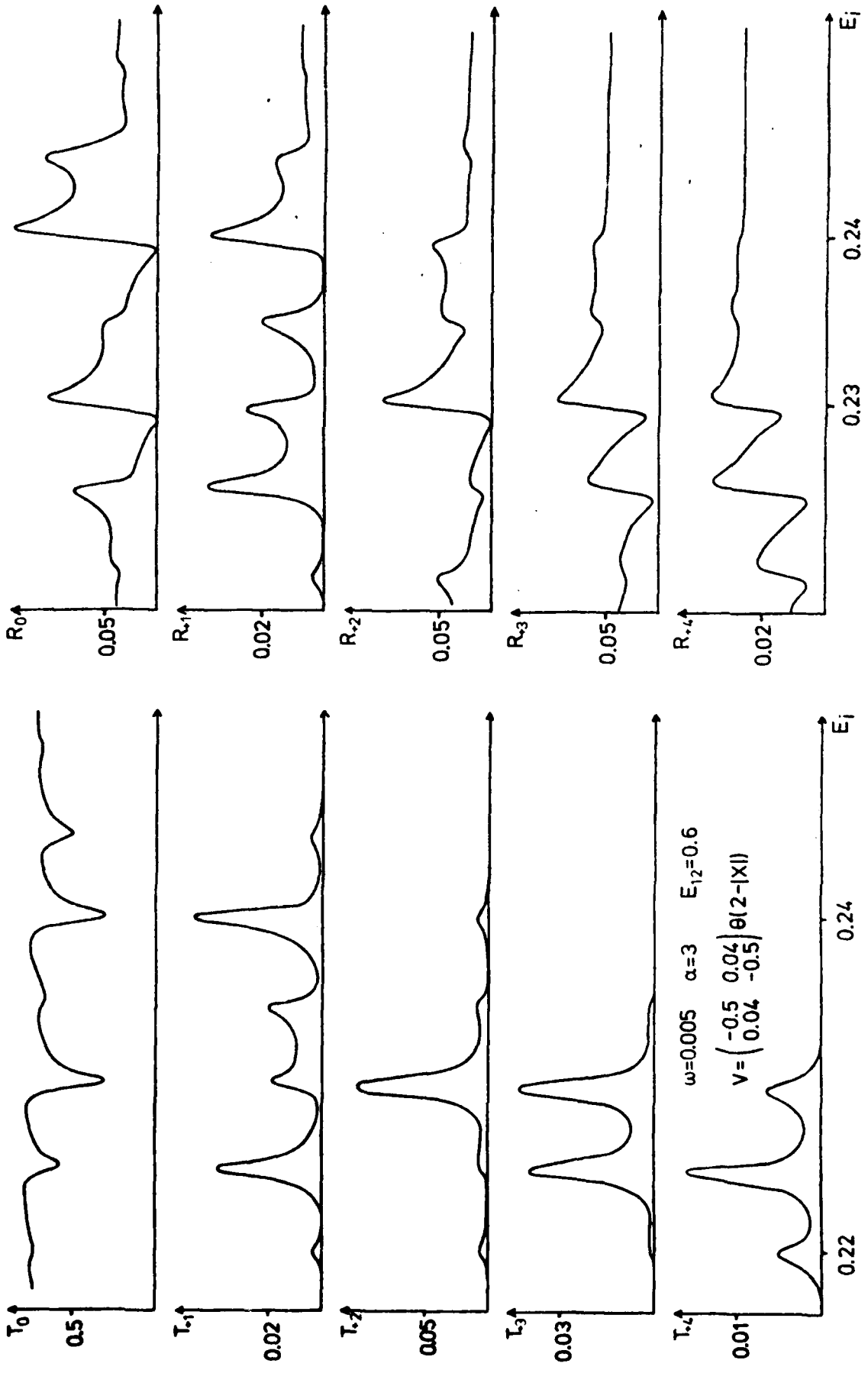

R0
0.05
R-1
0.02
R-2
0.05
R-3
0.05
R-4
0.02
Ei
0.24
0.23
T0
0.5
T-1
0.02
T-2
0.05
T-3
0.03
T-4
0.01
ω=0.005 α=3 E12=0.6
V=(-0.5 0.04 / 0.04 -0.5) θ(2-|X|)
0.22
0.24
Ei

Fig. 6

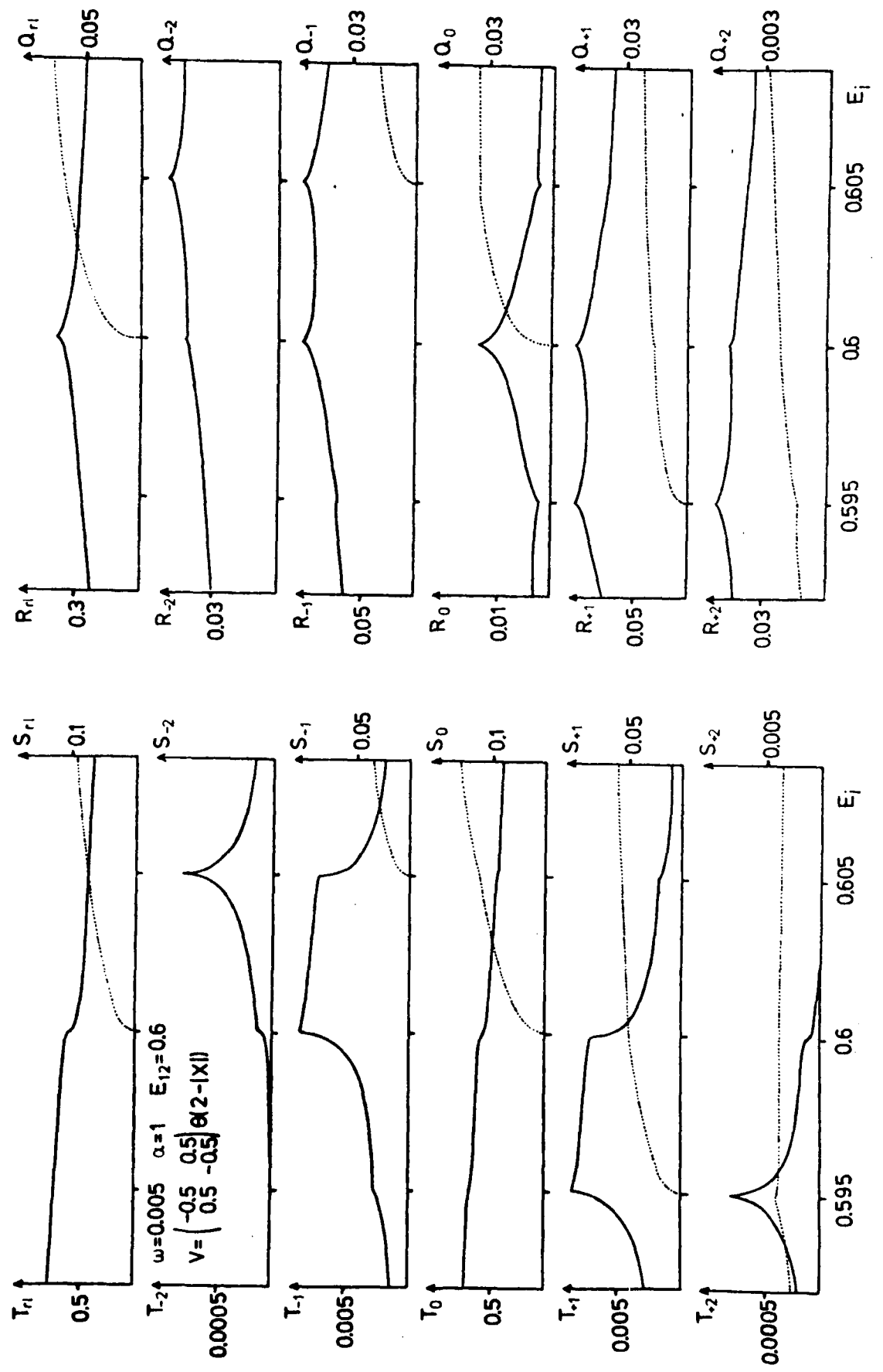

ω=0.005 α=1 E12=0.6
V=(-0.5 0.5) e^(2-|x|)
(0.5 -0.5)
T+1
0.5
T-2
0.0005
S+1
0.1
S-2
T-1
0.005
S-1
0.05
T0
0.5
S0
0.1
T+1
0.005
S+1
0.05
T+2
0.0005
S+2
0.005
0.595
0.6
0.605
Ei
R+1
0.3
R-2
0.03
R-1
0.05
R0
0.01
R+1
0.05
R+2
0.03
Q+1
0.05
Q-2
Q-1
0.03
Q0
0.03
Q+1
0.03
Q+2
0.003
0.595
0.6
0.605
Ei

Fig. 7

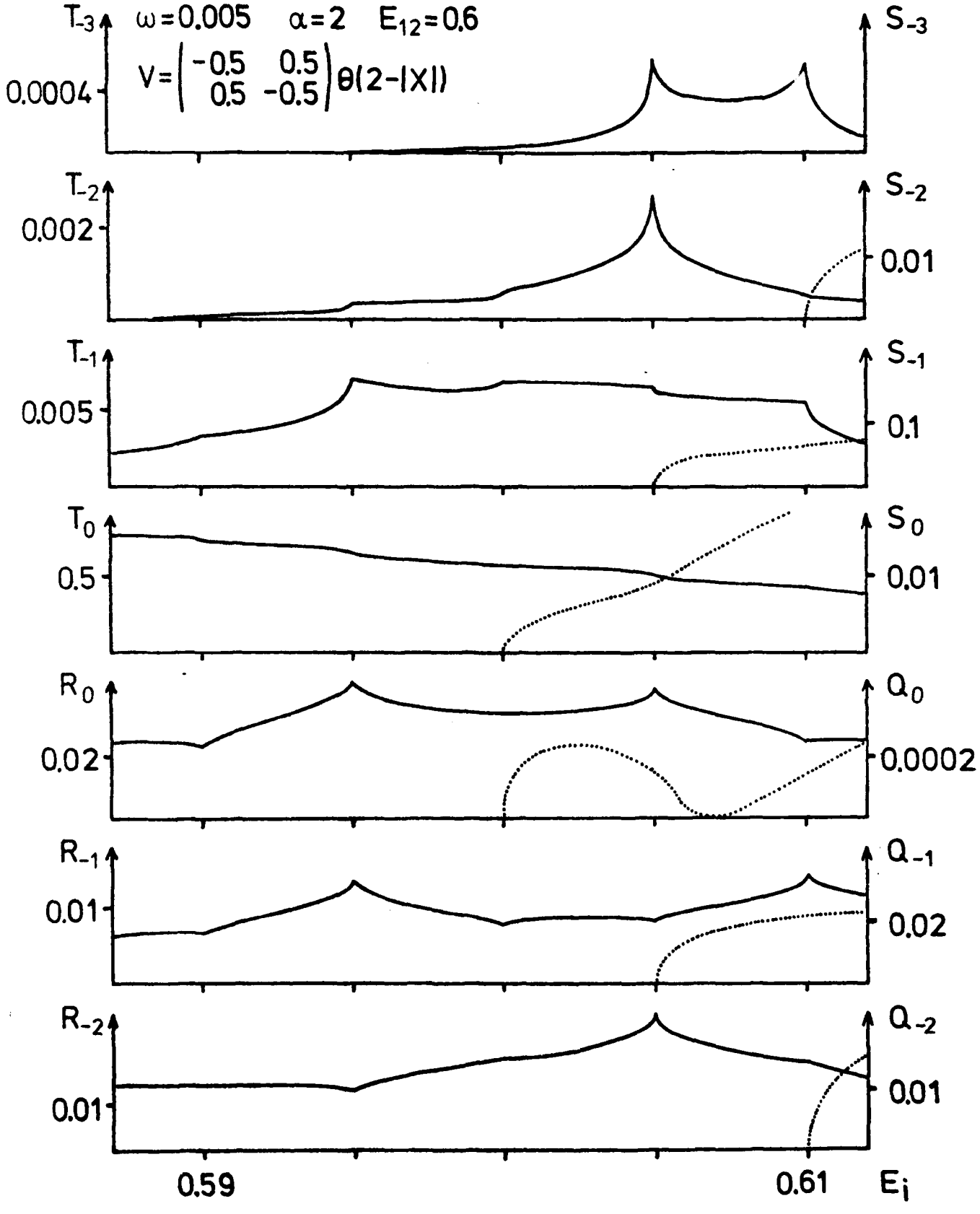
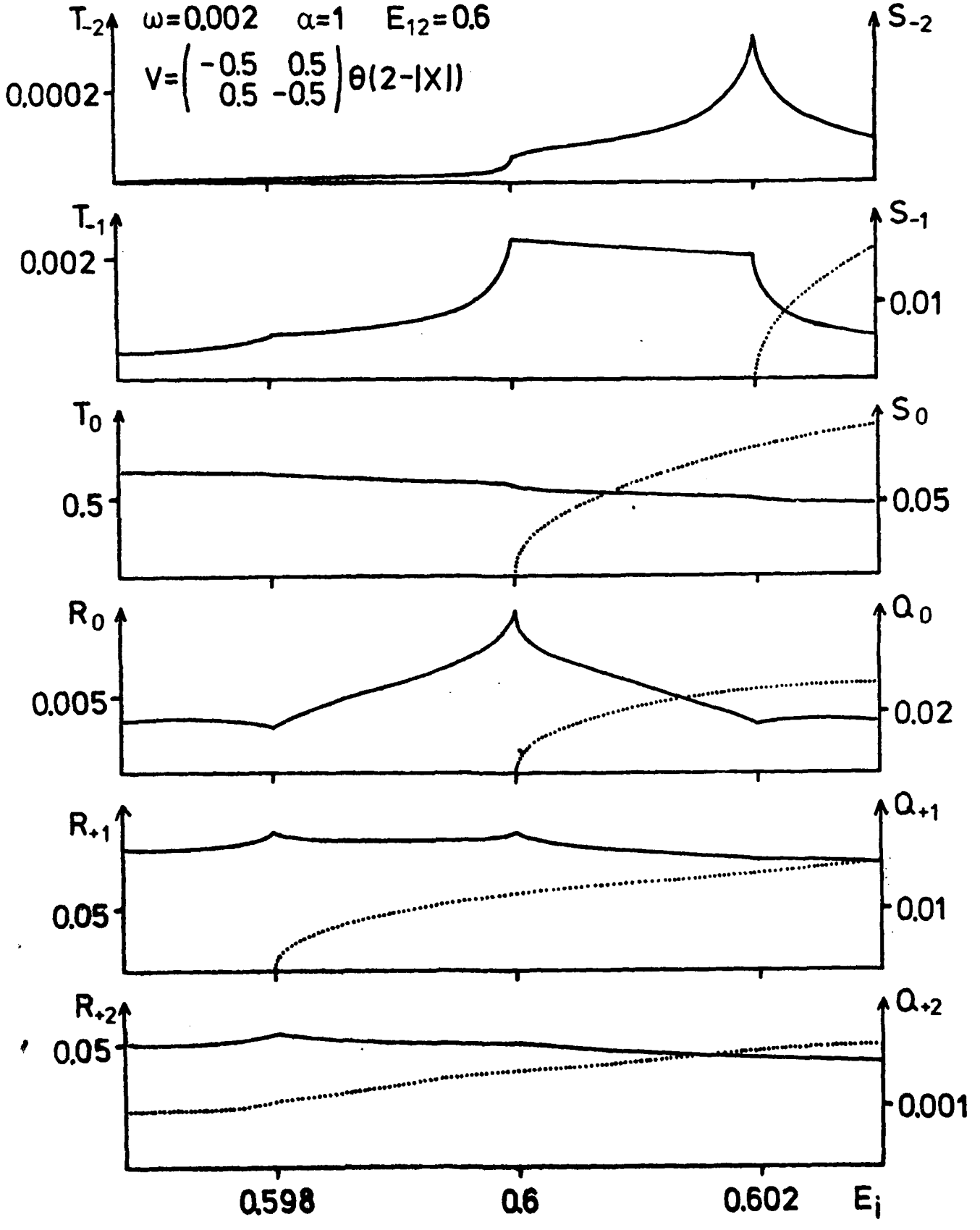

T_{-3}
$\omega = 0.005$ $\alpha = 2$ $E_{12} = 0.6$
$V = \begin{pmatrix} -0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix} \theta(2-|X|)$
0.0004
S_{-3}
T_{-2}
0.002
S_{-2}
0.01
T_{-1}
0.005
S_{-1}
0.1
T_0
0.5
S_0
0.01
R_0
0.02
Q_0
0.0002
R_{-1}
0.01
Q_{-1}
0.02
R_{-2}
0.01
Q_{-2}
0.01
0.59
0.61
E_i

Fig. 8


T_{-2}
$\omega=0.002$ $\alpha=1$ $E_{12}=0.6$
$V=\begin{pmatrix} -0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix}\theta(2-|X|)$
0.0002
S_{-2}
T_{-1}
0.002
S_{-1}
0.01
T_0
0.5
S_0
0.05
R_0
0.005
Q_0
0.02
R_{+1}
0.05
Q_{+1}
0.01
R_{+2}
0.05
Q_{+2}
0.001
0.598
0.6
0.602
E_i

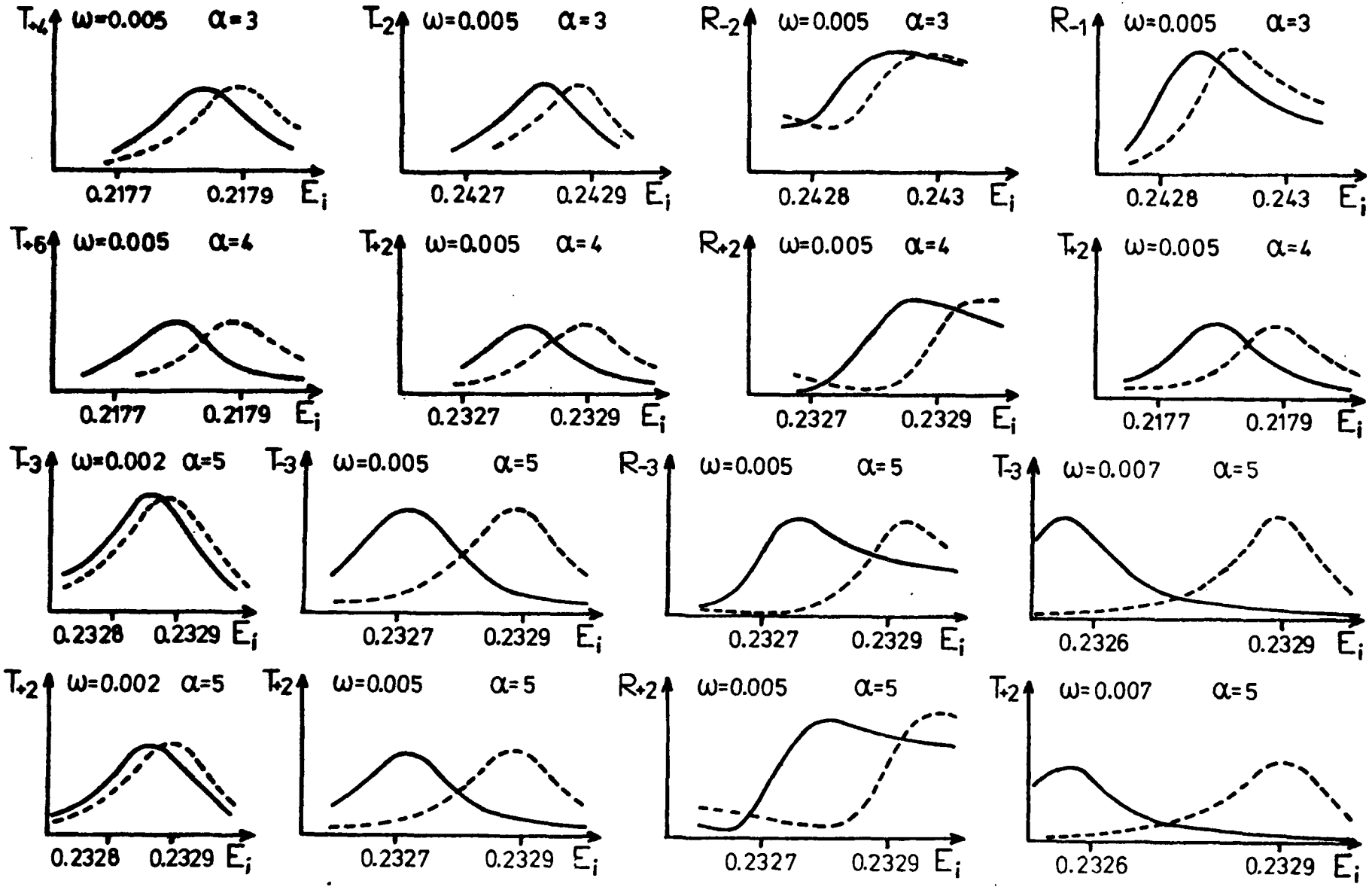

T+4 ω=0.005 α=3
0.2177 0.2179 Ei
T+2 ω=0.005 α=3
0.2427 0.2429 Ei
R-2 ω=0.005 α=3
0.2428 0.243 Ei
R-1 ω=0.005 α=3
0.2428 0.243 Ei
T+6 ω=0.005 α=4
0.2177 0.2179 Ei
T+2 ω=0.005 α=4
0.2327 0.2329 Ei
R+2 ω=0.005 α=4
0.2327 0.2329 Ei
T+2 ω=0.005 α=4
0.2177 0.2179 Ei
T-3 ω=0.002 α=5
0.2328 0.2329 Ei
T-3 ω=0.005 α=5
0.2327 0.2329 Ei
R-3 ω=0.005 α=5
0.2327 0.2329 Ei
T-3 ω=0.007 α=5
0.2326 0.2329 Ei
T+2 ω=0.002 α=5
0.2328 0.2329 Ei
T+2 ω=0.005 α=5
0.2327 0.2329 Ei
R+2 ω=0.005 α=5
0.2327 0.2329 Ei
T+2 ω=0.007 α=5
0.2326 0.2329 Ei

Fig. 9