

CP-663

April 2007\*

**LEVEL 8.0**  
**A Computer Program**  
**for Solving the Radial Schrödinger Equation**  
**for Bound and Quasibound Levels**

Robert J. Le Roy

*Guelph-Waterloo Centre for Graduate Work in Chemistry*  
*University of Waterloo, Waterloo, Ontario N2L 3G1, Canada*  
*Electronic mail: leroy@UWaterloo.ca*

\* Including corrections as of 11 December, 2012.

University of Waterloo

---

Chemical Physics Research Report

**LEVEL 8.0**  
**A Computer Program**  
**for Solving the Radial Schrödinger Equation**  
**for Bound and Quasibound Levels**

Robert J. Le Roy

*Guelph-Waterloo Centre for Graduate Work in Chemistry*  
*University of Waterloo, Waterloo, Ontario N2L 3G1, Canada*  
*Electronic mail: leroy@UWaterloo.ca*

This manual describes program LEVEL, which can solve the radial or one-dimensional Schrödinger equation for bound and/or quasibound levels of any smooth single or double-minimum potential, and calculate inertial rotation and centrifugal distortion constants, expectation values, and/or Franck-Condon factors and other off-diagonal matrix elements, either for levels of a single potential or between levels of two different potentials.

©Robert J. Le Roy, 2007

# 1 Introduction

Determining the number, energies and properties of vibration-rotational levels of a given one-dimensional or effective radial potential, and calculating matrix elements and transition intensities coupling levels of a single potential or levels of two separate potentials, are ubiquitous problems in chemical physics. The present report describes a robust and flexible computer program for performing such calculations. The original version of this program was based of the famous Franck-Condon intensity program of R.N. Zare [1, 2, 3], but the present version is considerably modified, and incorporates several unique features. In particular:

- (i) it will automatically locate and calculate the widths of quasibound (or orbiting resonance, or tunneling predissociation) levels;
- (ii) it can calculate diatomic molecule inertial rotation and centrifugal distortion constants for levels of a given potential;
- (iii) it can readily locate levels with dominant wave function amplitude over either well of an asymmetric double minimum potential;
- (iv) it can automatically locate and calculate expectation values for all vibration-rotation levels of any well-behaved single-minimum, “shelf state” or double minimum potential;
- (v) as an extension of (iv), it can automatically generate (for example) Franck-Condon factors and the radiative lifetimes for all possible discrete transitions allowed by specified (in the input data file) rotational selection rules between the levels of two different potentials, or among the levels of a single potential. While the present version generates the specified matrix elements and calculates Einstein  $A$  coefficients using the Hönl-London factors for the case of *singlet-singlet* electronic transitions, it may be generalized to treat other cases.

In the following, Section 2 presents the basic equation being solved, describes how the program functions, and outlines some of its options. Section 3 then states the input/output conventions, indicates the units assumed for the physical parameters of interest, and presents a shell to facilitate running the program on a UNIX or Linux system. The program’s operation is controlled by the contents of a data file which is read (on channel-5) during execution. The structure of this data file and the significance of the various read-in parameters are described in Section 4. Section 5 then describes the most significant differences between the current (8.0) and earlier [4] versions of this program. Finally, the Appendices outlines the structure of the program and the roles of its various subroutines, and present listings of illustrative sample data input files and of the resulting output.

The current version of the (extensively commented) source code for LEVEL and a ‘pdf’ file of this manual may be obtained from the www site <http://leroy.uwaterloo.ca/programs/>. While, there are no charges associated with distribution or use of this program, its use should be acknowledged in publications through reference to this report [5]. Users are also requested *not* to distribute the program themselves, but to refer other prospective users to the above web site or to the author. The version described herein includes corrections and enhancements incorporated up to 7 April 2007. Individuals currently utilizing older versions of this code [4] will likely find it desirable to obtain the current version since it has some corrections and additional functionality. I would also appreciate having users inform me of any apparent errors or instabilities in the code, or of additional features which might appear desirable for future versions.

## 2 Outline of Program Operation and Options

### 2.1 Solving the Radial Schrödinger Equation

The core of the program is concerned with determination of the discrete eigenvalues and eigenfunctions of the radial or (effective) one-dimensional Schrödinger equation

$$-\frac{\hbar^2}{2\mu} \frac{d^2 \Psi_{v,J}(r)}{dr^2} + V_J(r) \Psi_{v,J}(r) = E_{v,J} \Psi_{v,J}(r) \quad (1)$$

in which  $\mu$  is the effective or reduced mass of the system,  $J$  the rotational quantum number, and the effective one-dimensional potential  $V_J(r)$  is a sum of the rotationless (electronic) potential  $V(r)$  plus a centrifugal term. For the normal problem of a diatomic molecule rotating in three dimensions, this centrifugal potential has the form  $[J(J+1) - \Omega^2] \hbar^2 / 2\mu r^2$ , where  $\Omega = \text{OMEGA}$  is the projection of the electronic angular momentum onto the internuclear axis. However, for the special case of a diatom rotating in two dimensions, a case invoked by setting the read-in parameter  $\text{OMEGA} > 99$  (see the discussion of data input statement #5 in §4), this term becomes  $[J^2 - 1/4] \hbar^2 / 2\mu r^2$ . The program also defines the reduced mass appearing in Eq. (1) as Watson's "charge-modified reduced mass" [6],  $\mu = \mu_W = (M_A M_B) / (M_A + M_B - m_e Q)$ , where  $M_A$  and  $M_B$  are the atomic masses of the two atoms,  $m_e$  is the electron mass, and  $Q = \text{CHARGE}$  (see input READ statement #1) is the  $\pm$  integer net charge on the molecule (ion).

The core of the calculation is the solution of Eq. (1) to determine the eigenvalues  $E_{v,J}$  and eigenfunctions  $\Psi_{v,J}(r)$  of the potential  $V_J(r)$ . This is done in subroutine `SCHRQ`, which is based on the famous Cooley-Cashion-Zare routines `SCHR` [1, 2, 3, 7, 8], but incorporates special features, such as the ability to automatically locate and calculate the widths of "quasibound" or tunneling-predissociation levels [9, 10, 11, 12]. These are metastable levels which lie above the dissociation limit, but whose dissociation is inhibited by a potential energy barrier.

The accuracy of the eigenvalues and eigenfunctions obtained is largely determined by the size of the (fixed) radial mesh `RH` (READ #4 of the input data file) used in the numerical integration of Eq. (1). For potentials that are not too steep or too sharply curved, adequate accuracy is usually obtained using an `RH` value which yields a minimum of 15 to 30 mesh points between adjacent wavefunction nodes in the classically allowed region. An appropriate mesh size may be estimated using the particle-in-a-box expression

$$\text{RH} = \pi / \left( \text{NPN} \times [\mu \times \max\{E - V(r)\} / 16.857\ 629\ 20]^{1/2} \right) \quad (2)$$

in which `NPN` is the selected minimum number of mesh points per wavefunction node (say 25),  $\max\{E - V(r)\}$  is the maximum of the local kinetic energy (in  $\text{cm}^{-1}$ ) for the levels under consideration (in general it is  $\lesssim$  the potential well depth), and the numerical factor is identified below in Section 3. A value of `NPN` which is too small yields results which are unreliable, while too large a value may require excessive computational effort or cause array dimensions to be exceeded. Thus, while Eq. (2) is a useful guide, a careful user *should always* try different `RH` values in order to ensure that the results calculated achieve the accuracy desired for their particular application.

The numerical integration of Eq. (1) is performed on the range from `RMIN` to `RMAX` (see READ statement #4) using the Numerov algorithm [7, 13]. To initiate this integration, it is necessary to specify initial values of the wave function at two adjacent mesh points at each end of the range. For truly bound states, the wave function at the outer end of the range `RMAX` is initialized at an arbitrary value (say, unity), while its value at the adjacent inner mesh point is determined using the first-order semiclassical or WKB wavefunction [14]

$$\Psi_{v,J}(r) \propto [V_J(r) - E_{v,J}]^{-1/4} \exp\left(-\sqrt{2\mu/\hbar^2} \int^r [V_J(r') - E_{v,J}]^{1/2} dr'\right) \quad (3)$$

At short range, most realistic intermolecular potentials grow very steeply, causing the wavefunctions to die off extremely rapidly with decreasing  $r$ . As a result, the wave function at the inner end of the range of

integration is normally initialized by placing a node at the lower bound of this range, the read-in distance `RMIN`. This is effected by setting  $\Psi_{v,J}(r = \text{RMIN}) = 0$  and giving  $\Psi_{v,J}(r = \text{RMIN} + \text{RH})$  an arbitrary (non-zero) value. This is the normal case for a diatomic molecule problem. Note that one should normally set  $\text{RMIN} > 0$ , as the centrifugal contribution to the potential becomes singular at  $r = 0$ .

A special treatment of the inner boundary condition may be implemented if one is searching for eigenfunctions of a symmetric potential whose midpoint is located at `RMIN`. For asymmetric levels which would have a node at `RMIN`, the normal treatment described above will suffice. However, another approach must be implemented for symmetric levels of such a potential whose wave functions would have zero slope at `RMIN`. This option is built into subroutine `SCHRQ`, and is invoked by setting the control parameter `INNOD1`  $\leq 0$ . However, since this is an unusual case, varying this parameter is not one of the normal options of the current version of the main program, and a user who wishes to deal with this case may choose to either add parameter `INNOD1` to `READ #17` (line #367 of the `MAIN` program), or to recompile the code with the value of `INNOD1` and `INNOD2` defined on lines #365 & 366 of the code reset to 0.

If desired, a hard wall outer boundary condition may be imposed by setting the input integer `IV(i)` (see `READ #18`), which would otherwise represent the vibrational quantum number, equal to a large negative number which specifies the radial mesh point where this wall would be placed. In particular, setting `IV(i) < -10` causes a hard wall (wave function node) to be placed at mesh point number  $|\text{IV}(i)|$  for level  $-i$ .

In the Cooley procedure for finding the eigenvalues of Eq. (1) [7, 8], for any given trial energy the numerical integration proceeds inward from `RMAX` and outward from `RMIN` until the two solution segments meet at a chosen matching point  $r_x$ . The discontinuity in their slopes at  $r_x$  is then used to estimate the energy correction required to converge on the eigenvalue closest to the given trial energy [15], and this process is repeated until the energy improvement is smaller than the chosen convergence criterion (parameter `EPS` of `READ #4`). This procedure usually converges very rapidly, and for a single-minimum potential it is insensitive to the choice of the matching point  $r_x$ , as long as it lies in the classically-allowed region where the wavefunction amplitude is relatively large. However, to ensure high accuracy of calculated expectation values or matrix elements, `EPS` should usually be set 2 orders of magnitude smaller than the the actual eigenvalue precision required.

For an asymmetric double-well potential, wavefunctions usually have amplitudes of very different magnitude over the two wells, and the eigenvalue correction algorithm [15] used by `SCHRQ` tends to become unstable if the matching distance  $r_x$  lies in the well where the wavefunction has very small amplitude. As a result, it is usually necessary to require  $r_x$  to lie in the well where its amplitude is the largest. In the current version of the program, this choice is set by the internal control parameter `INNER` (= `SINNER`), which in turn is set to the appropriate value by the automatic vibrational level-finder subroutine `ALF`. As a result, calculations involving vibrational levels of a double well or “shelf-state” potential are (usually) performed just as routinely (for the user) as those for a normal single-well potential.

In general, the outward and inward numerical integration must start at distances `RMIN` and `RMAX` (input via `READ #4`), respectively, which lie sufficiently far into the classically-forbidden regions (where  $V_J(r) > E_{v,J}$ ) that the wavefunction has decayed by several orders of magnitude relative to its amplitude in the classically-allowed region. The present version of the code prints warning messages if this decay is not by a factor of at least  $10^{-9}$ ; if such warnings are printed, a smaller `RMIN` or larger `RMAX` value will be needed to ensure the desired accuracy for such cases. On the other hand, if `RMIN` or `RMAX` lie sufficiently far into the classically-forbidden regions that  $[V_J(r) - E]$  becomes extremely large, the integration algorithm can become numerically unstable for the given mesh size. For realistic diatomic molecule potential curves, this situation is only likely to occur near `RMIN`. If it does, a warning message is printed and the beginning of the integration range is automatically shifted outward until the problem disappears. However, use of a slightly larger value of `RMIN` will cause these warning messages to disappear and (marginally) reduce

the computational effort. For most diatomic molecules, a reasonable value of `RMIN` is ca. 0.6 – 0.8 times the smallest inner turning point encountered in the calculation, but for hydrides or other species of low reduced mass, even smaller values may be necessary.

The program internally defines the upper bound on the range of numerical integration `RMAX` as the smaller of the value read in (`READ #4`) and the largest distance consistent with the specified mesh and the internally-defined (see §3) potential and distance array dimension `NDIMR`. As with `RMIN`, the choice of `RMAX` is not critical, as long as (for truly bound states) the wave function has decayed to an amplitude much smaller than that in the classically allowed region, and the same amplitude decay test of  $10^{-9}$  is used for it. However, due to the anharmonicity of typical molecular potential curves, the requisite values of `RMAX` are much larger for highly excited vibrational levels than for those lying near the potential minimum. In order to reduce computational effort, an integration range upper bound  $r_{\text{end}}(v, J)$  is therefore determined for each level using the semiclassical result of Eq. (3), which shows that the wavefunction dies off exponentially in the classically forbidden region with an exponent of

$$-\sqrt{2\mu/\hbar^2} \int_{r_2(v, J)}^{r_{\text{end}}(v, J)} [V_J(r) - E_{v, J}]^{1/2} dr \quad (4)$$

where the turning point  $r_2(v, J)$  marks the outer end of the classically accessible region at this energy  $E_{v, J}$ . For each level it considers, `SCHRQ` first locates  $r_2(v, J)$ , and then determines a value of  $r_{\text{end}}(v, J)$  which is sufficiently large to ensure that this starting amplitude is smaller than that in the classically-allowed region by a factor of at least  $10^{-9}$ . In calculations for levels spanning a wide range of energies, the program’s use of this procedure can reduce the overall computation time by a factor of two or more.

## 2.2 Locating Quasibound Levels and Determining Their Widths

Quasibound or orbiting resonance levels are metastable eigenstates of Eq. (1) which lie at energies above the potential asymptote  $V_J(r = \infty)$ , but below a maximum in the outer part of the potential. As discussed in Refs. [9] & [10], the most efficient way of treating such levels is to determine their energies by applying an Airy function boundary condition at the third (outermost) turning point, and to calculate their widths using a uniform semiclassical method. The former appears to be the most accurate and efficient bound-state type method of locating quasibound or tunneling predissociation levels proposed to date [9, 10, 11, 16, 17]. It is virtually exact for narrow (long-lived) states, while for the very broadest levels lying marginally below barrier maxima, its differences with any competitive methods is at most a small fraction ( $\lesssim 0.2$ ) of the level width (FWHM). More accurate predictions for such short-lived states would require a detailed simulation of the actual process by which they are observed, since different methods of observing a given quasibound level may yield apparent energies differing by a small fraction of the level width. For example, if such levels are being observed spectroscopically, the peaks in the actual bound  $\rightarrow$  continuum spectrum would be calculated using a photodissociation simulation code [18, 19].

In the present program, the width calculation is based on Eq. (4.5) of Connor and Smith [11]; a more transparent description of this procedure may be found in §II.B of Ref. [12]. This is a uniform semiclassical procedure in which the predissociation rate may be thought of as being the product of the probability of tunneling past the barrier at the specified energy, times the vibrational frequency (inverse of the vibrational period) for the system trapped in the well behind the barrier. The actual calculation requires the evaluation of an integral of the type seen in Eq. (4) across the interval between the two outermost classical turning points (i.e., with the upper bound  $r_{\text{end}}(v, J)$  replaced by the outermost turning point  $r_3(v, J)$ ), and of an analogous integral across the classically allowed interval between the two inner turning points:

$$\sqrt{2\mu/\hbar^2} \int_{r_1(v, J)}^{r_2(v, J)} [E_{v, J} - V_J(r)]^{-1/2} dr \quad (5)$$

together with (the energy derivative of) a phase correction factor which takes account of the proximity to the barrier maximum [10, 11]. This procedure yields widths which are expected to be very reliable, particularly for narrow (long-lived) levels, but may have uncertainties of up to *ca.* 10% or more for the very broadest levels [10, 11]. To obtain more accurate results for such levels would again require one to perform a direct simulations of the process by which they are observed.

On a practical note, if the end of the numerical integration range **RMAX** is smaller than the outermost turning point  $r_3(v, J)$  of the metastable level of interest, the program attempts to generate a reasonable estimate of the width by completing the quadrature analytically while approximating the potential on the remainder of the interval by a centrifugal-type term  $C_2/r^2$  attached to the potential function at **RMAX**. If this approximation is invoked, warning messages are written to the main channel-6 output file (e.g., see the output for Case 3 in Appendix G).

### 2.3 Calculating Diatomic Molecule Centrifugal Distortion Constants

The rotational sublevels of a given vibrational level of a molecule are conventionally represented by the power series

$$E_{v,J} = G(v) + B_v[J(J+1)] - D_v[J(J+1)]^2 + H_v[J(J+1)]^3 + \dots \quad (6)$$

If desired, the program will calculate values of the inertial rotation constant  $B_v = (\hbar^2/2\mu)\langle v, J|1/r^2|v, J\rangle$  and of the first six centrifugal distortion constants associated with this expansion ( $D_v, H_v, L_v, M_v, N_v$  and  $O_v$ ). These constants have their normal significance for rotationless ( $J = 0$ ) vibrational levels [20], and are simply related to derivatives of the energy with respect to  $[J(J+1)]$  when calculated for vibration-rotation levels with  $J > 0$ . Calculation of these constants is invoked by setting input parameter **LCDC**  $> 0$  (see **READ #17**), and are performed using a subroutine based on Tellinghuisen's reformulation [21] of the exact quantum mechanical method of Hutson [22], which has been extended to higher order to allow the calculation of  $N_v$  and  $O_v$ . To ensure stable, fully converged calculations, it is often necessary to make the eigenvalue convergence parameter **EPS** of **READ #4** quite small (e.g.,  $\lesssim 10^{-6} \text{ cm}^{-1}$ ).

### 2.4 Calculating Expectation Values, Matrix Elements, and Einstein $\mathcal{A}$ Coefficients

If desired, the program will calculate expectation values or matrix elements of a function  $M(r)$  which may be defined either by interpolating over an array of input values, by a user-defined analytic function, or as a power series in a specified radial variable **RFN**( $r$ ):

$$M(r) = \sum_{i=0}^{\text{MORDR}} \text{DM}(i) \times \text{RFN}(r)^i \quad (7)$$

where parameters **MORDR**, **IRFN** and **RREF** defining the extent of the power series and the nature of the radial variable **RFN**( $r$ ) are input via **READ #19**, and the power series coefficients **DM**( $i$ ) are input via **READ #20**. In this last option, the radial variable **RFN**( $r$ ) is defined by the choice of input variable **IRFN** in the range  $-4 \leq \text{IRFN} \leq 9$  (see comments for **READ #19** in § 4), while setting **IRFN**  $\geq 10$  causes  $M(r)$  to be defined by interpolating over and extrapolating beyond a set of read-in numerical values, and setting **IRFN**  $\leq -10$  causes  $M(r)$  to be a user-defined analytic radial function. For this last case, code for calculating the desired function should be inserted in the program in the manner illustrates by the example in lines #498–519 of the main program. If  $M(r)$  is to be defined by interpolating over an array of read-in points, the necessary information is input via **READS #21 – 24**.

The conventional Franck-Condon factor  $\text{FCF} = |\langle \Psi_{v',J'} | \Psi_{v'',J''} \rangle|^2$  is the square of the matrix element of the zero<sup>th</sup> power of **RFN**( $r$ ), and will be generated whenever any off-diagonal matrix elements are calculated (i.e., whenever input parameter **LXPCT**  $\geq 3$ ). In this case, the program also assumes that  $M(r)$  is the transition dipole function (in debye), and uses its matrix element to calculate the Einstein  $\mathcal{A}$  coefficient

coupling the two levels in question. For cases in which a pointwise ( $\text{IRFN} \geq 10$ ) or analytic user-defined ( $\text{IRFN} \leq -10$ ) matrix element argument function  $M(r)$  is chosen, `MORDR` and `RREF` are dummy variables, and no `DM(i)` coefficients are read in. Note: to calculate *only* Franck-Condon factors, one should set `MORDR`  $< 0$ , in which case `IRFN` and `RREF` are dummy variables and no `DM(i)` values are read in.

The Einstein  $\mathcal{A}$  coefficient for the rate of spontaneous emission from initial-state level  $(v', J')$  into final-state level  $(v'', J'')$  is defined by the expression [20, 23]

$$\mathcal{A} = 3.1361891 \times 10^{-7} \frac{S(J', J'')}{2J' + 1} \nu^3 |\langle \Psi_{v', J'} | M(r) | \Psi_{v'', J''} \rangle|^2 \quad (8)$$

Here:  $\mathcal{A}$  has units  $\text{s}^{-1}$ ,  $M(r)$  is the dipole moment (or transition dipole) function in units debye,  $\nu$  the emission frequency in  $\text{cm}^{-1}$ ,  $S(J', J'')$  the Hönl-London rotational intensity factor and  $\Psi_{v', J'}$  and  $\Psi_{v'', J''}$  are the unit normalized initial and final state radial wave functions. The present version of the code incorporates  $S(J', J'')$  expressions for *singlet*  $\leftrightarrow$  *singlet* transitions obeying the parity selection rule, with  $\Delta\Lambda = 0$  or  $\pm 1$ . However, while versions of this code prior to 7.7 used the  $S(J', J'')$  expressions of Herzberg [20], subsequent versions use the revised Hönl-London factors recommended by Hansson and Watson [24], which for  $\Pi - \Sigma$  or  $\Sigma - \Pi$  transitions are a factor of four larger than those reported by Herzberg [20] (see also Bernath [23]). To generalize these selection rules for other cases, a user will need to modify lines #60 – 106 of subroutine `MATXEL`.

## 2.5 Defining the Rotationless Potential $V(r)$

The potential function package which reads required input and returns the potential array and associated parameters is controlled by subroutine `PREPOT`. It uses subroutine package `GENINT` for interpolation/extrapolation over a set of read-in turning points, and subroutine `POTGEN` for generating analytic potential functions. Values of the necessary input parameters enter via `READ` statements #5–16; for the 2–state case invoked by inputting `NUMPOT` = 2, this block of input statements is read twice.

One may choose to define a potential either by a set of `NTP` turning points  $\{\text{XI}(i), \text{YI}(i)\}$  input via `READ` #8, or (if `NTP`  $\leq 0$ ) by an analytic function. In the former case, interpolation over the read-in turning points to produce the array with mesh size `RH` required for the numerical integration of Eq. (1) is performed in a manner specified by the input parameter `NUSE`. For `NUSE`  $> 0$  this involves the use of piecewise `NUSE`–point polynomials (typically `NUSE` = 8 or 10), while for `NUSE`  $\leq 0$  the interpolation uses a cubic spline function. If the range of numerical integration [`RMAX`, `RMIN`] extends beyond that of the input turning points, appropriate extrapolation procedures are invoked. In particular, at distances smaller than the second of the read-in turning points `XI(2)`, the potential is extrapolated inward with an exponential function fitted to the first three turning points. Similarly, if `RMAX`  $>$  `XI(NTP–1)` the potential for  $r >$  `XI(NTP–1)` is extrapolated outward either as an exponential-type function or as a (sum of) inverse-power terms, as specified by parameters `ILR`, `NCN` and `CNN` of `READ` #6 (see § 4).

To define the potential by an analytic function, rather than by an array of points, the integer input parameter `NTP` of `READ` #5 should be set  $\leq 0$ . The program then *skips* `READS` #6–8 and proceeds instead to #9–16 (see § 4), where it reads values of the parameters defining the chosen analytic potential. The present version of the code allows for the following seven families of analytic potential energy functions.

- (i) The familiar Lennard-Jones( $m, n$ ) potential:

$$V(r) = \mathfrak{D}_e \left[ \left( \frac{n}{m-n} \right) \left( \frac{r_e}{r} \right)^m - \left( \frac{m}{m-n} \right) \left( \frac{r_e}{r} \right)^n \right] \quad (9)$$

- (ii) Various polynomial potentials based on Seto's modification [25] of Šurkus' GPEF potential form [26], which incorporates the familiar Dunham [27], Simons-Parr-Finlan [28], and Ogilvie–Tipping [29] expansions as special cases invoked by particular definitions of the expansion variable  $z = [(r)^p - (r_e)^p] / [a_S (r)^p + b_S (r_e)^p]$  (see discussion of `READ` #9 in § 4):



$$V(r) = c_0 z^2 \left( 1 + \sum_{i=1} c_i z^i \right) \quad (10)$$

(iii) The ‘‘Extended Morse Oscillator’’ EMO function [25, 30, 31, 32]

$$V(r) = \mathfrak{D}_e \left[ 1 - e^{\phi(r)(r-r_e)} \right]^2 \quad (11)$$

where  $\phi(r)$  is a simple polynomial in the Šurkus variable [26]

$$y_p(r) = [(r)^p - (r_e)^p] / [(r)^p + (r_e)^p] \quad (12)$$

in which  $p$  is a selected small positive integer. The exponent polynomial may in general have different orders for  $r > r_e$  and  $r \leq r_e$ , while truncating it at the constant term yields the familiar simple Morse potential.

Another Morse-type function allowed by POTGEN is Hua Wei’s 4-parameter potential [33]

$$V(r) = \mathfrak{D}_e \left( [1 - e^{-b(r-r_e)}] / [1 - C e^{-b(r-r_e)}] \right)^2 \quad (13)$$

(iv) The ‘‘Morse/Long-Range’’ (MLR) [34] or ‘‘Morse/Lennard-Jones’’ (MLJ) [35, 32] functions, in which a flexible form for the potential well incorporates one (for MLJ) or more (for MLR) of the leading inverse-power terms which define the theoretically predicted long-range interaction energy [35, 32]:

$$V(r) = \mathfrak{D}_e \left[ 1 - \frac{u_{\text{LR}}(r)}{u_{\text{LR}}(r_e)} e^{-\phi(r)y_p(r)} \right]^2 \quad (14)$$

where  $\phi(r)$  is a constrained polynomial in the Šurkus variable [26] which is normally written as

$$\phi(r) = [1 - y_p(r)] \sum_{i=0}^N \phi_i [y_p(r)]^i + y_p(r) \phi_\infty \quad (15)$$

in which  $\phi_\infty = \lim_{r \rightarrow \infty} \phi(r) = \ln \{2\mathfrak{D}_e / u_{\text{LR}}(r_e)\}$ . The general form of  $u_{\text{LR}}(r)$  is [36, 34]

$$u_{\text{LR}}(r) = \frac{C_{m_1}}{r^{m_1}} + \frac{C_{m_2}}{r^{m_2}} + \dots \quad (16)$$

and as  $r \rightarrow \infty$ , the overall potential takes on the form  $V(r) \simeq \mathfrak{D}_e - u_{\text{LR}}(r)$ . The one-term version of Eq. (16) yields is what was called an ‘MLJ’ function, while for two or more terms it is called an ‘MLR’ potential. While there is a single set of  $\phi_i$  expansion coefficients, the polynomial in Eq. (15) may have different orders for  $r > r_e$  and  $r \leq r_e$ . Alternate expressions for the exponent coefficient  $\phi(y_p)$  are also allowed; see comments associated with READ #9 in §4.

(v) The ‘‘Double-Exponential Long-Range’’ or DELR potential of Ref. [12]:

$$V(r) = A e^{-2\phi(r)(r-r_e)} - B e^{-\phi(r)(r-r_e)} + V_{\text{LR}}(r) \quad (17)$$

where  $\phi(r)$  is a simple power series in the Šurkus variable of Eq. (12),  $V_{\text{LR}}(r)$  is a function chosen to represent the long-range region,

$$V_{\text{LR}}(r) = \sum_m D_m(r) C_m / r_m \quad (18)$$

and factors  $A$  and  $B$  are defined in terms of the well depth and the properties of  $V_{\text{LR}}(r)$  at  $r_e$ :

$$A = \mathfrak{D}_e + V_{\text{LR}}(r_e) + V'_{\text{LR}}(r_e) / \phi_0 \quad (19)$$

$$B = 2\mathfrak{D}_e + 2V_{\text{LR}}(r_e) + V'_{\text{LR}}(r_e) / \phi_0 \quad (20)$$

where  $V'_{\text{LR}}(r_e) \equiv dV_{\text{LR}}(r)/dr|_{r=r_e}$ . Two choices for the damping functions  $D_m(r)$  in Eq. (18) are those proposed by Tang and Toennies [37] and by Douketis *et al.* [38]:

$$D_m^{\text{TT}}(r) = 1 - e^{-3.16\rho_d r} \sum_{k=0}^m (3.16\rho_d r)^k / k! \quad (21)$$

$$D_m^{\text{DS}}(r) = [1 - \exp\{-3.97(\rho_d r)/m - 0.39(\rho_d r)^2/\sqrt{m}\}]^m \quad (22)$$

where  $\rho_d = \text{PARM}(\text{NVARB} - 1)$  is a system-dependent scaling parameter.

- (vi) The generalized HFD function defined in terms of the reduced distance variable  $x = r/r_e$  [39, 40]:

$$V(r) = A x^\gamma e^{-\beta_1 x - \beta_2 x^2} - D(x) \sum_m C_m / r^m \quad (23)$$

in which  $D(x) = \exp\{-\alpha_1[\alpha_2/x - 1]^{\alpha_3}\}$  is a damping function which cuts off the inverse-power terms at short distances.

- (vii) A form introduced by Tiemann and co-workers [41] which represents the main part of the potential by a GPEF-like power series:

$$V(r) = a_0 + a_1 \xi + a_2 \xi^2 + a_3 \xi^3 + a_4 \xi^4 + \dots \quad (24)$$

in which  $\xi \equiv \xi(r, b, r_m) = (r - r_m)/(r + b r_m)$ , and the parameter  $b$  is chosen to take account of the different steepness of the potential for  $r$  greater and less than  $r_m$ . The existence of a non-zero linear term ( $a_1 \neq 0$ ) in Eq.(24) means that the parameter  $r_m$  only approximately corresponds to the equilibrium distance. At a specified small distance, this power series is smoothly joined to an exponential function, and beyond a specified large distance, it is replaced by a sum of inverse power terms (see § 4).

Note that except for the simple polynomial potentials of type (ii), all of these analytic potentials are defined relative to the absolute energy at the asymptote, which is specified by input parameter VLIM (see READ #5). For GPEF-type potentials of type (ii), input parameter VLIM specifies the absolute energy at the potential minimum.

A user may readily introduce their own analytic potential function form by simply replacing subroutine POTGEN with their own potential routine. To retain consistency with the rest of the present code, such a user-prepared POTGEN subroutine should have the argument list:

POTGEN(LNPT, N, IAN1, IAN2, IMN1, IMN2, VLIM, R, RM2, VV, NCN, CNN)

The first argument, parameter LNPT, is an integer which in program LEVEL has a value  $\geq 1$ .<sup>\*</sup> The other input quantities are the integer N specifying the size of the array of radial distances R(i) (in Å) at which potential values are to be generated, the squared inverse distance array RM2(i) = 1/R(i)<sup>2</sup>, the absolute energy VLIM (in cm<sup>-1</sup>) at the potential asymptote, and integers giving the atomic numbers (IAN1 & IAN2) and mass numbers (IMN1 & IMN2) of that particular isotopologue (required for calculating BOB terms, see below). The subroutine is expected to return the desired N-point array of potential function values VV(i) (in units cm<sup>-1</sup>), as well as the integer NCN and real positive coefficient CNN. Under the option in which the program automatically searches for many or all vibrational levels of a given potential (when input parameter NLEV1 is large and negative, see READ #17), the limiting long-range potential is assumed to have the form  $V(r) \simeq \mathfrak{D} - \text{CNN}/r^{\text{NCN}}$ , and the parameters NCN and CNN returned from POTGEN are used in a near-dissociation theory [42, 43, 44] algorithm to estimate the number and energies of missing levels. If the user's analytic potential has a barrier maximum or dies off exponentially rather than as an inverse power, NCN should be set at some large integer value (e.g., NCN = 99).

<sup>\*</sup> While not used in LEVEL, LNPT is retained in the calling sequences of subroutines GENINT and POTGEN to facilitate the use of these subroutines in other programs.

## 2.6 Born-Oppenheimer Breakdown Radial Strength functions

In recent years, it has become increasingly common for combined-isotopologue spectroscopic data analyses to require the inclusion of atomic-mass-dependent Born-Oppenheimer breakdown (BOB) corrections to the rotationless and centrifugal potential energy functions. LEVEL will include such terms if the value of input parameter `IBOB`  $> 0$ . These (optional) functions are defined as in Ref. [45]. In particular, for each atom  $a = A$  or  $B$ , the additive correction to the potential energy function is defined as a constrained polynomial in the Šurkus variable of Eq. (12)

$$\left( \frac{M_a - M_a^{\text{ref}}}{M_a} \right) \left( y_m(r) u_\infty^a + [1 - y_m(r)] \sum_{j=0}^{N_{\text{ad}}^a} u_j^a [y_p(r)]^j \right) \quad (25)$$

where  $M_a$  is the mass of the particular isotope of atom- $a$ ,  $M_a^{\text{ref}}$  the mass of the chosen reference isotope of that species [45],  $y_m(r)$  and  $y_p(r)$  have the form of Eq. (12), and the integers  $m$  and  $p$  are chosen as discussed in Ref. [45].

The centrifugal BOB correction is a multiplicative factor  $[1 + g_A(r) + g_B(r)]$  applied to the centrifugal contribution to the overall potential function  $V_J(r)$  of Eq. (1), where the terms associated with the two atoms  $a = A$  or  $B$  have the same type of radial form as the ‘‘adiabatic’’ potential function corrections of Eq. (25):

$$g_a(r) = \frac{M_a^{\text{ref}}}{M_a} \left( y_p(r) t_\infty^a + [1 - y_p(r)] \sum_{j=0}^{N_{\text{na}}^a} t_j^a [y_p(r)]^j \right) \quad (26)$$

## 3 Units, Physical Constants, Array Dimensions, Input/Output Conventions, and Program Execution

Unless otherwise specified, the units of length and energy used throughout this program, and assumed for all input data, are Å and  $\text{cm}^{-1}$ , respectively. The main exception is that the transition dipole function  $M(r)$  of Eqs. (7) used for calculating the Einstein coefficients of Eq. (8), defined in terms of the expansion coefficients  $\text{DM}(i)$  of `READ #20` (see §4), is assumed to be in debye (where  $1 \text{ debye} = 3.335\,640\,952 \times 10^{-30} [\text{C} \cdot \text{m}] = 0.393\,430\,295 [\text{au}]$ ). Note, however, that in the `IRFN  $\geq 10$`  option for generating the radial function defining the matrix element argument (see `READ #19`) by numerical interpolation over a set of read-in points, the channel-6 output describing the read-in transition moment function values being interpolated over may (incorrectly) refer to their units as  $\text{cm}^{-1}$  rather than debye, since the interpolation is done by the same subroutine package set up to deal with an input pointwise potential. Note too that if set of read-in points is used to define the potential or the matrix element argument  $M(r)$ , the values may be in any convenient units, as appropriate conversion factors are always also read in (see `READS #7 & 23`) to convert them to the appropriate units.

The values of the physical constants appear in the program in two places. The first is the dimensionless factor  $\hbar^2/(2u_0 E_0 r_0^2) = 16.857\,629\,20$  in the radial Schrödinger equation of Eq. (1), where the choice of reference mass  $u_0 = 1 \text{ amu}$ , distance  $r_0 = 1 \text{ Å}$ , and energy  $E_0 = 1 \text{ cm}^{-1}$  effectively define the units of the input/output variables. The second is in the collections of terms defining the numerical factor in Eq. (8) used in calculating the Einstein coefficient for the rate of spontaneous emission. Our current values of these constants are based on the 2002 CODATA recommended values of Ref. [46], while the atomic isotope masses stored in subroutine `MASSSES` were taken from the compilation of Ref. [47].

The array dimension limits which a user may wish to change are set in `PARAMETER` statements in the main driver routine and in subroutines `GENINT` and `SPLINT`. In the former, `NDIMR` (currently 80001) is the maximum dimension of the radial mesh array on which the potential, wave functions and radial expectation

value/matrix element arguments are defined. For systems of small reduced mass, it could be safely set considerably smaller than this. The second parameter set in the main program is VIBMX (currently 400), which defines the maximum number of vibration/rotation levels for which vibrational eigenvalues may be read and stored, and the upper bound on the number of rotational sublevels which may be stored when applying the  $NJM > 0$  option to automatically generate many  $J$  sublevels for a given  $v$  (see READ #17). The two other array-size parameters set inside the code are NTPMX (currently 1600) which is the maximum number of potential function points (or radial matrix element  $M(r)$  values, for  $IRFN > 10$ ) which may be read in, which is set in GENINT, and MAXSP (set in SPLINT), the number of spline coefficients required when interpolating over the read-in function values, and should be set equal to  $4 \times NTPMX$ .

The program reads input data on channel-5, writes standard output to channel-6, and optionally (controlled by parameters LPPOT, LCDC and LXPCT of READS #5 & 17) writes a condensed output file to one or more of channels 7-10. Those executing the program using a UNIX or Linux operating system environment may wish to create and store in the system or user's 'bin' directory a shell named (say) 'rlev', such as that shown here:

```
# UNIX shell 'rlev' to execute the compiled version of program LEVEL named
#   lev.x, which is stored in the user's directory /upath/ with input data
#   file $1.5, and write output to $1.6, $1.7, etc. all in the current
#   directory.
#
time ~/upath/lev.x < $1.5 > $1.6
mv fort.7 $1.7 >& /dev/null
mv fort.8 $1.8 >& /dev/null
mv fort.9 $1.9 >& /dev/null
mv fort.10 $1.10 >& /dev/null
```

This shell allows the program to be executed with the simple command: `rlev <filename>` where `<filename>.5` is the input data file the user has created (`<filename>` may be any name, usually chosen to identify a particular case). In this case the standard output from channel-6 will be written to file `<filename>.6`, and the channel-7, channel-8, channel-9, and channel-10 output will be written to files `<filename>.7`, ... , `<filename>.10`, respectively.

## 4 Data file Structure and input parameter definitions

All of the READ statements for inputting data to the program, and the associated logical structure, are listed here. The following subsection then provides a detailed description of the nature and/or options associated with each of the input variables.

```
#1      READ(5,*,END=999) IAN1, IMN1, IAN2, IMN2, CHARGE, NUMPOT
#2a     IF(IAN1.LE.0) READ(5,*) NAME1, MASS1
#2b     IF(IAN2.LE.0) READ(5,*) NAME2, MASS2
#3      READ(5,*,END=999) TITL
#4      READ(5,*) RH, RMIN, RMAX, EPS
        DO IPOT= 1, NUMPOT
#5      READ(5,*) NTP, LPPOT, OMEGA, VLIM
        IF(NTP.GT.0) THEN
#6      READ(5,*) NUSE, IR2, ILR, NCN, CNN
#7      READ(5,*) RFACT, EFACT, VSHIFT
#8      READ(5,*) (XI(I), YI(I), I= 1,NTP)
        ELSE
#9      READ(5,*) IPOTL, MPAR, NSR, NCMM, NVARB, IBOB, DSCM, REQ
#10     IF(IPOTL.GE.4) READ(5,*) (MMLR(I), CMM(I), I= 1, NCMM)
#11     IF(NVARB.GT.0) READ(5,*) (PARM(I), I=1,NVARB)
```

```

                IF(IBOB.GT.0) THEN
#12                READ(5,*) MN1R, MN2R, PAD, MAD, NU1, NU2, PNA, NT1, NT2
                    IF(PAD.GT.0) THEN
#13                        IF(NU1.GE.0) READ(5,*) U1INF, (U1(I), I=0,NU1)
#14                        IF(NU2.GE.0) READ(5,*) U2INF, (U2(I), I=0,NU2)
#15                        IF(NT1.GE.0) READ(5,*) T1INF, (T1(I), I=0,NT1)
#16                        IF(NT2.GE.0) READ(5,*) T2INF, (T2(I), I=0,NT2)
                    ELSE
#13a                       IF(NU1.GE.0) READ(5,*) U1INF, (U1(I), I=0,NU1),Aad1,Rad1
#14a                       IF(NU2.GE.0) READ(5,*) U2INF, (U2(I), I=0,NU2),Aad2,Rad2
#15a                       IF(NT1.GE.0) READ(5,*) T1INF, (T1(I), I=0,NT1),Ana1,Rna1
#16a                       IF(NT2.GE.0) READ(5,*) T2INF, (T2(I), I=0,NT2),Ana2,Rna2
                    ENDIF
                ENDIF
            ENDDO
#17            READ(5,*) NLEV1, AUTO1, LCDC, LXPCT, NJM, JDJR, IWR, LPRWF
                NLEV= MAX(1,NLEV1)
#18a           IF(AUTO1.GT.0) READ(5,*) (IV(I), IJ(I), I= 1,NLEV)
#18b           IF(AUTO1.LE.0) READ(5,*) (IV(I), IJ(I), GV(I), I= 1,NLEV)
                IF((LXPCT.NE.0).AND.(LXPCT.NE.-1)) THEN
#19                READ(5,*) MORDR, IRFN, RREF
#20                IF((IABS(IRFN).LE.9).AND.(MORDR.GE.0)) READ(5,*) (DM(J), J= 0,MORDR)
                    IF(IRFN.GE.10) THEN
#21                        READ(5,*) NRFN, RFLIM
#22                        READ(5,*) NUSEF, ILRF, NCNF, CNNF
#23                        READ(5,*) RFACTF, MFACTF
#24                        READ(5,*) (XIF(I), YIF(I), I= 1,IRFN)
                    ENDIF
                ENDIF
            IF(IABS(LXPCT).GE.3) THEN
#25                READ(5,*) NLEV2, AUTO2, J2DL, J2DU, J2DD
#26a           IF(AUTO2.GT.0) READ(5,*) (IV2(I), I= 1,NLEV2)
#26b           IF(AUTO2.GT.0) READ(5,*) (IV2(I), GV(I), I= 1,NLEV2)
            ENDIF

```

## Definitions and Descriptions of Input File Data

Read integers identifying the molecule or system.

#1. READ(5,\*) IAN1, IMN1, IAN2, IMN2, CHARGE, NUMPOT

**IAN1 & IAN2:** integer atomic numbers of the atoms/particles #1 & 2 forming the molecule. If both are positive and  $\leq 109$ , atomic masses from the tabulation in subroutine MASSES will generate the reduced mass of the system. If either is  $\leq 0$  or  $> 109$  the mass of that particle will be input via READ #2.

**IMN1 & IMN2:** integer mass numbers of the atoms/particles #1 & 2 forming the molecule. For a normal stable atomic isotope, its mass is taken from the tabulation in subroutine MASSES; if outside the range for the normal stable isotopes of that atom, the abundance-averaged atomic mass will be used.

**CHARGE:**  $\pm$  integer for the total charge on the molecule. Used to generate Watson's charge-modified reduced mass for molecular ions [6]:  $\mu = M_A M_B / (M_A + M_B - m_e \times \text{CHARGE})$ , where  $m_e$  is the electron mass.

**NUMPOT**: the number of potentials considered: =1 for calculations involving only a single potential function; =2 to input and generate two different potentials and calculate matrix elements coupling their levels.

In the special case when IAN1 and/or IAN2 is either  $\leq 0$  or  $> 109$ , read in a two-character alphanumeric name for that particle and its mass (in amu). This facilitates the treatment of model systems or exotic species such as muonium or positronium “molecules”.

#2.a IF(IAN1.LE.0) READ(5,\*) NAME1, MASS1

#2.b IF(IAN2.LE.0) READ(5,\*) NAME2, MASS2

**NAME1 & NAME2**: a two-character alphanumeric name for the (1 or 2) particle(s) whose mass is being read, enclosed in single quotes, as in 'mu'.

**MASS1 & MASS2**: the masses of particles 1 and 2, in amu.

Read a text title or description for the calculation.

#3. READ(5,510) (TITL(I), I= 1,20)

**TITL**: a title or output header for the calculation, consisting of up to 78 characters on a single line, enclosed between single quotes: e.g., 'title of problem'.

Read real numbers defining the mesh and range of the numerical integration (all in Å), and the eigenvalue convergence criterion to be used (in  $\text{cm}^{-1}$ ).

#4 READ(5,\*) RH, RMIN, RMAX, EPS

**RH**: the numerical integration mesh size; see discussion associated with Eq. (2) in §2.1.

**RMIN & RMAX**: the inner and outer limits, respectively, of the range of numerical integration (see §2). Plausible zero<sup>th</sup> order estimates would be  $\text{RMIN} \approx 0.6 \times (\text{potential inner wall position})$  and  $\text{RMAX}$  fairly large (say 40 Å). Internally  $\text{RMAX}$  is set to the smaller of this read-in value and the largest distance allowed by  $\text{RMIN}$ ,  $\text{RH}$  and the array dimension  $\text{NDIMR}$  (see §3).

**EPS**: the eigenvalue convergence parameter used by SCHRQ (in  $\text{cm}^{-1}$ ). To ensure that appropriately accurate expectation values or matrix elements are generated, it should usually be set ca. 2 orders of magnitude smaller than the eigenvalue precision actually required.

Some combination of the next 12 READ statements defines the potential energy function. A particular case always starts with READ #5, but then uses *either* READS #6–8 for numerical interpolation over a set of input turning points, *or* (some of) READS #9–16 for the case of an analytic potential function.

#5. READ(5,\*) NTP, LPPOT, OMEGA, VLIM

**NTP**: An integer which is set  $\leq 0$  to generate an analytic potential using POTGEN, in which case the program skips READS #6–8 and goes directly to READ #9. If  $\text{NTP} > 0$ , it is the number of turning point pairs to be input via READ #8.

**LPPOT**: controls printing of the potential array (usually set = 0 to have no printing). If  $\text{LPPOT} > 0$  write to standard output (channel-6) the potential and its first 2 derivatives-by-differences at every  $\text{LPPOT}^{\text{th}}$  mesh point; it is sometimes useful to do this when troubleshooting. Setting  $\text{LPPOT} < 0$  writes the resulting potential in condensed format to channel-8 at every  $|\text{LPPOT}|^{\text{th}}$  mesh point; this is useful if one wants to input this potential into a plotting routine.

**OMEGA**: is the (integer) projection of the electronic orbital angular momentum on the molecular axis for this state, so that the reduced centrifugal potential becomes  $[J(J+1) - \text{OMEGA}^2]\hbar^2/2\mu r^2$ . For rotation constrained to a plane setting  $\text{OMEGA} \geq 99$  will cause the centrifugal potential to have the appropriate form:  $[J^2 - 1/4]\hbar^2/2\mu r^2$ .

**VLIM**: The absolute energy (in  $\text{cm}^{-1}$ ) of the potential asymptote. This sets the absolute energy scale seen in the output. For power series (GPEF- or Dunham-type) potentials (IPOTL=4), it is the energy at the potential function minimum  $r = r_e$ .

For a pointwise potential (if  $\text{NTP} > 0$ ), we must specify how the interpolation is to be done, and since **RMAX** often lies outside the range of the input turning points, we also must specify how the potential is to be extrapolated at large  $r$ . Skip down to **READ #9** if  $\text{NTP} \leq 0$ .

#6. **READ(5,\*) NUSE, IR2, ILR, NCN, CNN**

**NUSE**: Specifies how the interpolation is to be done. If  $\text{NUSE} > 0$  use  $\text{NUSE}$ -point piecewise polynomials; if  $\text{NUSE} \leq 0$  perform cubic spline interpolation. For highly precise and smooth input points, such as those generated from an RKR calculation,  $\text{NUSE} = 8, 10$  or  $12$  is usually most appropriate; for less precise or less dense points, such as those from *ab initio* calculations, low order piecewise polynomials ( $\text{NUSE} = 4$ ) or splines ( $\text{NUSE} \leq 0$ ) are usually best.

**IR2**: For very steep repulsive potential walls, better interpolation is often attained by actually interpolating over  $r^2 \times V(r)$ ; setting the integer  $\text{IR2} > 0$  causes this to be done (normally recommended). A comparison between results obtained with this option turned on *vs.* off (setting  $\text{IR2} \leq 0$  causes interpolation to be performed over  $V(r)$  itself) provides an indication of the magnitude of “interpolation noise” uncertainties in the final results.

**ILR**: Specifies how to extrapolate from the outermost read-in turning points to **RMAX**. For a long extrapolation, one of  $\text{ILR} = -1, 0$  or  $1$  is often most appropriate; however, if the outer turning points extend moderately close to the dissociation limit (at **VLIM**), one should set  $\text{ILR} \geq 2$  and specify the theoretically appropriate value of **NCN** ( $\geq 1$ ), and if it is available, also input an estimate of **CNN** (see below).

For  $\text{ILR} < 0$ , fit the last 3 points to:  $V(r) = \text{VLIM} - A \times \exp[-b(r - r_o)^2]$

For  $\text{ILR} = 0$ , fit the last 3 points to:  $V(r) = \text{VLIM} - A \times r^p \times \exp[-br]$ .

For  $\text{ILR} = 1$ , fit the last 2 points to:  $V(r) = \text{VLIM} - A/r^B$ .

For  $\text{ILR} = 2$  or  $3$ , respectively, fit the outermost 2 or 3 points to a sum of 2 or 3 inverse-power terms, with powers differing by 2:  $V(r) = \text{VLIM} - \sum_{m=0}^{\text{ILR}-1} C_{\text{NCN}+2m}/r^{\text{NCN}+2m}$ .

For  $\text{ILR} \geq 4$ , fit outermost  $\text{ILR}$  turning points to a sum of  $\text{ILR}$  inverse-power terms, with powers differing by 1:  $V(r) = \text{VLIM} - \sum_{m=0}^{\text{ILR}-1} C_{\text{NCN}+m}/r^{\text{NCN}+m}$ .

**NCN**: For inverse-power potential extrapolation with  $\text{ILR} \geq 2$ , **NCN** ( $> 0$ ) specifies the limiting inverse-power behaviour:  $V(r) \propto \text{VLIM} - \text{CNN}/r^{\text{NCN}}$ . Otherwise (for  $\text{ILR} \leq 1$ ) it is a dummy input variable.

**CNN**: For inverse-power potential extrapolation with  $\text{ILR} \geq 2$ , setting  $\text{CNN} \neq 0$  causes the leading inverse-power coefficient to be fixed at the read-in value  $\text{CNN} = C_{\text{NCN}} [\text{cm}^{-1} \text{\AA}^{\text{NCN}}]$  rather than be determined from a fit to the outermost turning points.

The input turning points may come from *ab initio* or RKR calculations, and may need to be converted to the units ( $\text{\AA}$  and  $\text{cm}^{-1}$ ) used inside the program.

#7. **READ(5,\*) RFACT, EFACT, VSHIFT**

**RFACT & EFACT**: are multiplicative factors required to convert units of the  $\text{NTP}$  input turning point distances  $\text{XI}(i)$  and energies  $\text{YI}(i)$  to  $\text{\AA}$  and  $\text{cm}^{-1}$ , respectively. If no conversion is required, read in factors of  $1.0\text{D}+00$ .

**VSHIFT**: An energy shift (in  $\text{cm}^{-1}$ ) to be added to the input potential point energies to make them consistent with **VLIM**. It addresses the fact that *ab initio* or RKR turning points may be expressed relative to an energy zero inconsistent with the user-specified asymptote energy **VLIM**.

Read in the actual turning points.

#8. READ(5,\*) (XI(I), YI(I), I= 1,NTP)

**XI(i) & YI(i)**: are the (distance, energy) input turning points defining the potential function.

If the input potential is defined by an analytic function (the  $NTP \leq 0$  case) using subroutine POTGEN, the parameters characterizing it are input via some or all of READ statements #9–16. For a user-supplied POTGEN function, the input parameters (if any) are up to them, but the calling sequence must match that expected by PREPOT (see discussion at the end of §2.5).

#9. READ(5,\*) IPOTL, MPAR, NSR, NCMM, NVARB, IBOB, DSCM, REQ

#10. IF(IPOTL.GE.4) READ(5,\*) (MMLR(I), CMM(I), I= 1,NCMM)

#11. IF(NVARB.GT.0) READ(5,\*) (PARM(I), I=1,NVARB)

**IPOTL**: an integer specifying the type of analytic function used for the potential.

IPOTL = 1 generates a Lennard-Jones ( $m = NSR, n = MPAR$ ) function using Eq. (9). In this case NCMM and NVARB are dummy variables.

IPOTL = 2 generates the power series potential of Eq. (10) using Seto's [25] rearranged form of Šurkus' [26] expansion variable  $z = z(r) = (r^{MPAR} - r_e^{MPAR}) / (a_S r^{MPAR} + b_S r_e^{MPAR})$ , with  $c_0$  input as  $c_0 = DSCM$ ,  $PARM(i) = c_i$  for  $i = 1$  to  $(NVARB - 2)$ ,  $a_S = PARM(NVARB - 1)$  and  $b_S = PARM(NVARB)$ . Note that Šurkus' case of  $MPAR < 0$  is accommodated by Seto's identity [25]:  $z(-MPAR, a_S, b_S) = z(MPAR, -b_S, -a_S)$ . Here NSR and NCMM are dummy variables.

- Dunham expansions are generated by setting  $MPAR = 1$ ,  $\alpha_S = 0.0$  &  $b_S = 1.0$ .
- SPF expansions are generated by setting  $MPAR = 1$ ,  $\alpha_S = 1.0$  &  $b_S = 0.0$ .
- Ogilvie–Tipping expansions are generated by setting  $MPAR = 1$ ,  $\alpha_S = b_S = 0.5$ .
- A simple harmonic oscillator potential is generated by setting  $MPAR = 1$ ,  $NVARB = 2$ ,  $a_S = 0.0$  and  $b_S = 1.0$ , while the harmonic force constant  $k = 2 DSCM / REQ^2$ .
- All of these polynomial-type potentials have an undefined (or at best, indirectly-defined) asymptote, so parameter VLIM defines the potential energy minimum.
- If  $MPAR = 0$ , the potential is generated as an NVARB-order polynomial in  $r$  whose constant coefficient is set by the input VLIM value.

IPOTL = 3 normally generates the EMO potential of Eq. (11), where the expansion-variable power  $p = MPAR \geq 1$  and the expansion coefficients are  $PARM(i) = \phi_{i-1}$  for  $i = 1$  to NVARB. The order of the exponent polynomial  $\phi(r) = \sum_{i=0}^N \phi_i [y_p(r)]^i$  is  $N = N_L = (NVARB - 1)$  for  $r \geq r_e$ , and  $N = N_S = NSR$  for  $r < r_e$ , where necessarily  $0 < NSR \leq (NVARB - 1)$ . For this case NCMM is a dummy variable.

If  $MPAR = -2$ , generate original Coxon-Hajigeorgiou “GMO” potential [48, 49], in which the exponent coefficient  $\phi(r)$  is a simple power series in  $(r - r_e)$  of order  $(NVARB - 1)$  with coefficients  $PARM(i) = \phi_{i-1}$  for  $i = 1$  to NVARB.

If  $MPAR = -1$ , generate Hua Wei's [33] 4-parameter Morse-like potential

$$V(r) = \mathfrak{D}_e ([1 - e^{-b(r-r_e)}] / [1 - C e^{-b(r-r_e)}])^2 \quad \text{where } b = PARM(1) \text{ and } C = PARM(2).$$

In this case NVARB = 2 and NCMM & NSR are dummy variables.

IPOTL = 4 generates an MLR or MLJ potential from Eqs. (14–16) [35, 50, 32, 34], where  $DSCM = \mathfrak{D}_e$ ,  $REQ = r_e$ , and the potential tail of Eq. (16) is a sum of of NCMM inverse-power terms  $C_m / r^m$ , with  $m = MMLR(j)$  and  $C_m = CMM(j)$  [ $\text{cm}^{-1} \text{Å}^m$ ] for  $j = 1$  to NCMM. Positive  $C_m$  values are attractive and negative values repulsive. If  $u_{LR}(r)$  consists of more than one inverse-power term, necessarily  $p > (MMLR(NCMM) - MMLR(1))$  [34].



For  $\text{MPAR} > 0$  and  $\text{NSR} > 0$ , integer  $\text{MPAR} = p$  defines the power  $p$  in the radial variable of Eq. (12), and the exponent coefficient in Eq. (14) is given by Eq. (15). The upper bound on the summation defining  $\phi(r)$  is defined as  $N = N_L = (\text{NVARB} - 1)$  for  $r \geq r_e$  and as  $N = N_S = \text{NSR}$  for  $r < r_e$ , where necessarily  $N_S \leq N_L$ , and the exponent expansion coefficients are  $\text{PARM}(i) = \phi_{i-1}$  for  $i = 1$  to  $\text{NVARB} = (N_L + 1)$ . This case can yield either an MLJ ( $\text{NCMM} = 1$ ) or an MRL ( $\text{NCMM} > 1$ ) form.

Setting  $\text{MPAR} < 0$  yields an MLJ potential ( $\text{NCMM} = 1$ ) with the power  $p$  defining the radial variable of Eq. (12) being  $p = |\text{MPAR}|$  and the exponent coefficient having the form [35]  $\phi(r) = f_S(r) \sum_{i=0}^{N_L} \phi_i [y_1(r)]^i + [1 - f_S(r)] \phi_\infty$ , where the switching function  $f_S(r) = 1/[e^{\alpha_S(r-r_S)} + 1]$  is defined by parameters  $\alpha_S = \text{PARM}(\text{NVARB} - 2)$  and  $r_S = \text{PARM}(\text{NVARB} - 1)$ , and at distances  $r < r_{\text{inn}} = \text{PARM}(\text{NVARB})$  the short-range repulsive potential is extrapolated linearly with the slope of the MLJ function at  $r = r_{\text{inn}}$ . For this model,  $\text{NSR}$  is a dummy variable, and the coefficients of the exponent expansion are  $\text{PARM}(i) = \phi_{i-1}$  for  $i = 1$  to  $(\text{NVARB} - 3) = (N_L + 1)$ .

Setting  $\text{MPAR} = 0$  yields a special version of the preceding ( $\text{MPAR} < 0$ ) case in which the expansion variable power  $p = 1$  and the expansion variable has the extra factor of 2 associated with the original Ogilvie-Tipping variable [29]:  $y_1(r) = 2(r - r_e)/(r + r_e)$ .

For  $\text{MPAR} > 0$  and  $\text{NSR} \leq 0$ , the power  $p = \text{MPAR}$  and  $\phi(r)$  is a simple polynomial in  $y_p$  of order  $(\text{NVARB} - 1)$  with the  $\text{NVARB}$  coefficients  $\phi = \text{PARM}(i)$  for  $i = 0$  to  $(\text{NVARB} - 1)$ .

[This is a legacy case, and *not* a recommended model.]

$\text{IPOTL} = 5$  generates the DELR potential of Eqs. (17–20) [12], where  $\text{DSCM} = \mathcal{D}_e$ ,  $\text{REQ} = r_e$ , the power defining the expansion variable  $y_p(r)$  is read in as  $\text{MPAR} = p$  and  $\text{NCMM}$  is the number of terms in the inverse-power sum of Eq. (18). The exponent coefficient  $\phi(r)$  has the same simple power-series form as that for an EMO potential ( $\text{IPOTL} = 3$ , above), with expansion coefficients  $\text{PARM}(i) = \phi_{i-1}$  for  $i = 1$  to  $(\text{NVARB} - 2) = (N_L + 1)$ , and as for most MLR and EMO cases, the exponent polynomial order for  $r < r_e$  is set by  $\text{NSR} \leq N_L$ . The damping function scaling parameters is input as  $\rho_d = \text{PARM}(\text{NVARB} - 1)$ , while setting  $\text{PARM}(\text{NVARB}) > 0$  selects the Tang-Toennies damping function of Eq. (21) and setting  $\text{PARM}(\text{NVARB}) < 0$  selects the Douketis *et al.* function of Eq. (22). The powers  $m = \text{MMLR}(i)$  and coefficients  $C_m = \text{CMM}(i) \text{ cm}^{-1} \text{ \AA}^m$ ] of Eq. (18) are input through `READ #10`, and in this model, positive values of  $C_m$  yield repulsive terms and negative values attractive ones.

$\text{IPOTL} = 6$  generates the generalized HFD( $m = \text{MMLR}(j)$  for  $j = 1$  to  $\text{NCMM}$ ) type potentials of Eq. (23), where  $A$  and  $\beta_1$  are defined (internally) by the input values of  $\mathcal{D}_e = \text{DSCM}$  and  $r_e = \text{REQ}$ , the damping function parameters are read in as  $\text{PARM}(i) = \alpha_i$  for  $i = 1 - 3$ , the other exponent coefficients are  $\beta_2 = \text{PARM}(4)$  and  $\gamma = \text{PARM}(5)$ , and the inverse-power coefficients  $C_m$  are input as dimensionless parameters (positive if attractive)  $\text{CMM}(j) = \overline{C}_m = C_m/[\mathcal{D}_e r_e^m]$  for  $j = 1$  to  $\text{NCMM}$ . For this case,  $\text{NVARB} = 5$ , while  $\text{MPAR}$  and  $\text{NSR}$  are dummy parameters.

$\text{IPOTL} = 7$  generates the Tiemann-type potential [41] of Eq. (24), which consists of a power series of order  $\text{IORD} = (\text{NVARB} - 4)$  in the variable  $\xi = (r - r_m)/(r + b r_m)$ , where  $b = \text{PARM}(\text{NVARB} - 2)$ , with expansion coefficients  $\text{PARM}(i) = a_{i-1}$  for  $i = 1$  to  $(\text{NVARB} - 3)$ . The true overall well depth is  $\mathcal{D}_e = \text{DSCM}$ , and the reference distance  $r_m$  (which for  $a_1 \neq 0$  is *not* identical to  $r_e$ ) is read in as `REQ`. For  $r < r_{\text{inn}} = \text{PARM}(\text{NVARB} - 1)$  the potential is extrapolated inward by smoothly attaching the function  $A + C_{12}/r^{12}$  to the power series

at the distance  $r_{\text{inn}}$ , while for  $r > r_{\text{out}} = \text{PARM}(\text{NVARB})$  it is extrapolated as a sum of NCMM inverse-power terms with powers  $m = \text{MMLR}(j)$  and coefficients  $C_m = \text{CMM}(j)$  for  $j = 1$  to NCMM, with the (internal) addition of one further inverse-power term  $C_{m_L}/r^{m_L}$  with  $m_L = [\text{MMLR}(\text{NCMM}) + 2]$ , whose coefficient  $C_{m_L}$  is defined (internally) to attach this long-range tail to the polynomial at  $r = r_{\text{out}}$ . For this model, MPAR and NSR are dummy parameters.

**MPAR & NSR:** Integers used to characterize particular potential forms (see above).

**NCMM:** Integer specifying how many inverse-power terms will be used to define the long-range part of the potential for cases  $\text{IPOTL} \geq 4$ .

**NVARB:** Integer specifying the number of (real) parameters  $\text{PARM}(i)$  read in to define the potential.

**IBOB:** An integer to specify whether (for  $\text{IBOB} > 0$ ) or not (for  $\text{IBOB} \leq 0$ ) atomic-mass-dependent Born-Oppenheimer breakdown correction terms are to be included in the rotationless (electronic) and/or centrifugal ( $[J(J+1) - \Omega^2]\hbar^2/2\mu r^2$ ) potential functions (see § 2.6).

**DSCM:** Normally (except for the  $\text{IPOTL} = 2$  case, above), the potential well depth  $\mathfrak{D}_e$  in  $\text{cm}^{-1}$ .

**REQ:** Normally (except for the  $\text{IPOTL} = 7$  case, see above) the equilibrium distance  $r_e$  in Å.

If atomic-mass-dependent Born-Oppenheimer breakdown (BOB) terms are to be incorporated into the potential energy and/or centrifugal potential functions ( $\text{IBOB} > 0$ ), use READS #12 and one or more (as needed) of #13–16. These BOB functions normally have the forms defined by Eqs. (25) and (26). If a given type of correction function is to be omitted, the associated expansion order (e.g., NU1, NU2, NT1 or NT2) should be set  $< 0$ , in which case the associated parameter READ statement (of #13–16) is omitted, while if no BOB corrections are considered ( $\text{IBOB} \leq 0$ ), omit all of READS #12–16.

#12. READ(5,\*) MN1R, MN2R, PAD, MAD, NU1, NU2, PNA, NT1, NT2

**MN1R & MN2R:** The integer mass numbers of the *reference* isotopes of atoms 1 & 2, respectively (or atoms  $A$  &  $B$ , respectively) whose masses  $M_a^{\text{ref}}$  appear in Eqs. (25) & (26) [51].

**NU1 & NU2:** For atoms  $a = 1$  & 2 (or  $A$  &  $B$ ), the orders  $N_{\text{ad}}^a$  of the polynomial expansions of Eq. (25). Set them  $< 0$  to neglect READS # 13 and/or 14 and omit such corrections.

**NT1 & NT2:** For atoms  $a = 1$  & 2, the orders  $N_{\text{na}}^a$  of the polynomial expansions of Eq. (26). Set them  $< 0$  to neglect READS # 15 and/or 16 and omit such corrections.

**PAD & PNA:** For  $\text{PAD} > 0$ , the (positive) integer  $p$  defining the polynomial expansion variable  $y_p(r)$  for the adiabatic potential energy BOB function of Eq. (25) is  $p = \text{PAD}$ , and that for the non-adiabatic centrifugal BOB function of Eq. (26) is  $p = \text{PNA}$ .

For  $\text{PAD} \leq 0$ , use the BOB radial forms of Coxon & Hajigeorgiou [52] which utilize an explicit switching function  $f_S(r) = 1/[e^{A_{\text{ada}}(r - R_{\text{ada}})} + 1]$ . In place of Eq. (25), the adiabatic correction to the potential is:  $m_e[1/M_a - 1/M_a^{\text{ref}}] \left\{ f_S(r) \sum_{j=0}^{\text{NU}a} \text{Ua}(j)(z)^j + [1 - f_S(r)/f_S(r_e)] \text{UaINF} \right\}$  for  $a = 1$  or 2, where  $z = 2(r - r_e)/r + r_e$ . In place of Eq. (26), the non-adiabatic centrifugal correction function has the form:  $g_a(r) = [m_e/M_a] \left\{ \sum_{j=0}^{\text{NT}a} \text{Ta}(j)[(r - r_e)/r_e]^j \right\}$  for  $a = 1$  or 2. For  $\text{PAD} \leq 0$ , use READS #13a–16a, as appropriate, rather than READS #13–16.

**MAD:** For the normal  $\text{PAD} > 0$  case, MAD is the (positive) integer ( $m = \text{MAD}$ ) in Eq. (25) which defines the limiting long-range inverse-power behaviour of the adiabatic potential correction function. This should normally be set equal to the power of the leading inverse-power term contributing to the long-range potential  $\text{MAD} = \text{MMLR}(1)$ . If  $\text{MAD} = 0$  and  $\text{UaINF} = 0$  the adiabatic potential correction function is collapsed to a simple power series in  $y_p(r)$ .

For cases in which  $IBOB > 0$  and  $PAD > 0$ , use READS #13–16.

#13. IF(NU1.GE.0) READ(5,\*) U1INF, (U1(I),I= 0,NU1)

#14. IF(NU2.GE.0) READ(5,\*) U2INF, (U2(I),I= 0,NU2)

#15. IF(NT1.GE.0) READ(5,\*) T1INF, (T1(I),I= 0,NT1)

#16. IF(NT2.GE.0) READ(5,\*) T2INF, (T2(I),I= 0,NT2)

**UaINF & Ua(i)**: for  $a = 1$  or  $2$ , are the (real\*8) expansion parameters defining the ‘adiabatic’ potential correction functions of Eq. (25):  $Ua(j) = u_j^a$  and  $UaINF = u_\infty^a$  in units  $\text{cm}^{-1}$ .

**TaINF & Ta(i)**: for  $a = 1$  or  $2$ , are the (real\*8) expansion parameters defining the centrifugal potential correction functions of Eq. (26):  $Ta(j) = t_j^a$  and  $TaINF = t_\infty^a$  are dimensionless parameters.

For the Coxon-Hajigeorgiou BOB functions [52] invoked by setting  $PAD \leq 0$ , use READS #13a–16a in place of #13–16. For this case, always set  $T1INF = T2INF = 0$ .

#13a. IF(NU1.GE.0) READ(5,\*) U1INF, (U1(I),I= 0,NU1)

#14a. IF(NU2.GE.0) READ(5,\*) U2INF, (U2(I),I= 0,NU2)

#15a. IF(NT1.GE.0) READ(5,\*) T1INF, (T1(I),I= 0,NT1)

#16a. IF(NT2.GE.0) READ(5,\*) T2INF, (T2(I),I= 0,NT2)

For a calculation involving only a single potential energy curve ( $NUMPOT = 1$  in READ #1), now proceed directly to READ #17. However, if  $NUMPOT = 2$  first repeat READS #5–16 to input the second potential function.

Now read the parameters controlling which levels are to be calculated, and what expectation values and or matrix elements are to be calculated (if any).

#17. READ(5,\*) NLEV1, AUTO1, LCDC, LXPCT, NJM, JDJR, IWR, LPRWF

**NLEV1**: If  $> 0$ , integer NLEV1 is the number of levels to be calculated, for each of which the quantum number specifications will be input via READ #18.

If  $\leq 0$ , the program automatically finds all vibrational levels from  $v = 0 - |\text{NLEV1}|$  associated with the rotational quantum number read in as IJ(1) (see below). If the input value of NLEV is very large and negative, the program will (attempt to) find *all* possible vibrational levels associated with the specified  $J = \text{IJ}(1)$ .

**AUTO1**: Integer AUTO1  $> 0$  (normal option) causes the program to (attempt to) automatically generate realistic trial eigenvalues for all desired levels, so only their quantum number labels need be input via (READ #18a). If this fails, setting AUTO1  $\leq 0$  will allow/require a trial energy  $GV(i)$  to be input (via READ #18b) for each specified level (using the NLEV1  $> 0$  option).

**LCDC**: If LCDC  $> 0$ , calculate the inertial rotational constant  $B_v$  and the first 6 centrifugal distortion constants  $\{-D_v, H_v, L_v, \dots \text{ etc.}\}$  for all of the levels specified by NLEV1. If LCDC  $\geq 2$  also write them in a compact format to channel–9.

**LXPCT**: An integer controlling what expectation values/matrix elements are to be calculated. For LXPCT = 0, no expectation values or matrix elements are calculated (and hence READS #19–24 are omitted). *Even* values of  $|\text{LXPCT}| \neq 0$  cause the results to be written in compact form to channels–7 or 8 (as appropriate, see below), as well as to channel–6; *Odd* values of LXPCT yield only channel–6 output, and *negative* values of LXPCT cause most of the (relatively wordy) print to channel–6 to be suppressed.

LXPCT = -1 causes the eigenvalues and (if appropriate) quasibound level widths to be written compactly to channel-7, and no expectation values or matrix elements are calculated (so READS #19-24 are omitted).

LXPCT = 1, 2 or -2 causes calculation of expectation values of the kinetic energy and of positive powers of the distance variable specified by READ #19 plus either #20 or #21-24 (as appropriate, see below). Write results to channel-6 if LXPCT = 1 or 2; also write them (compactly) to channel-7 if LXPCT = 2, and write them only (compactly) to channel-7 if LXPCT = -2.

$|\text{LXPCT}| \geq 3$  invokes the calculation of matrix elements coupling levels of Potential-1 to each other (if NUMPOT = 1) or to levels of Potential-2 (if NUMPOT = 2), as specified by READS #25 & 26. Write results to channel-6 if LXPCT > 0 and (compactly) to channel-8 if LXPCT =  $\pm 4$ .

$|\text{LXPCT}| \geq 5$  also causes the component radial moments comprising the overall matrix elements to be written to channel-7, while still writing the overall matrix elements for selection-rule allowed transitions to channel-8. For LXPCT =  $\pm 6$  write *only* the radial moment components, and omit the output to channel-8.

**NJM & JDJR:** If (integer) NJM > 0, for each (vibrational) level generated by the NLEV1 specification, automatically calculate eigenvalues (and if appropriate, expectation values and matrix elements) for all rotational sublevels  $J$  ranging from the input-specified (see below)  $J = \text{IJ}(i)$  to a maximum of  $J = \text{NJM}$  (or until that vibrational level energy predissociates above the potential barrier), with  $J$  increasing in steps of JDJR. To automatically generate all possible rotational levels, set JDJR = 1, IJ( $i$ ) = 0 and NJM very large (e.g., set NJM = 999).

**IWR:** An integer controlling the printout of diagnostics and calculation details inside SCHRQ.

If IWR  $\neq 0$  print warning and error messages inside subroutine SCHRQ, as appropriate. Unless one is troubleshooting, normally set IWR = -1.

If IWR  $\geq 1$  also print final eigenvalue and node count for every level determined.

If IWR  $\geq 2$  also print end-of-range wave function amplitudes.

If IWR  $\geq 3$  also print intermediate trial eigenvalues as the iterative convergence proceeds.

**LPRWF:** If LPRWF > 0 write to channel-6 the wavefunction at every {LPRWF}<sup>th</sup> mesh point.

If LPRWF < 0 write wavefunction compactly to channel-10 at every |LPRWF|<sup>th</sup> mesh point.

If LPRWF = 0, no wavefunction printout.

Read quantum numbers specifying which vibration-rotation levels (of Potential-1) are to be determined.

#18 a. IF(AUTO1.GT.0) READ(5,\*) (IV(I),IJ(I),I= 1,max{ 1, NLEV1})

#18 b. IF(AUTO1.LE.0) READ(5,\*) (IV(I),IJ(I),GV(I),I= 1,max{1, NLEV1})

**IV(i) & IJ(i):** For NLEV1 > 0 these are the vibrational [ $v = \text{IV}(i)$ ] and rotational [ $J = \text{IJ}(i)$ ] quantum numbers of the levels to be determined; if NJM > IJ( $i$ ) the program also automatically calculates rotational levels for that  $v = \text{IV}(i)$  for  $J = \text{IJ}(i)$  to NJM in steps of JDJR.

For NLEV1  $\leq 0$ , read one {IV( $i$ ),IJ( $i$ )} pair. The value of IV(1) is ignored, but  $J = \text{IJ}(1)$  is the rotational quantum number for which all vibrational levels up to  $v = |\text{NLEV1}|$  are to be determined.

**GV(i):** If AUTO1  $\leq 0$ , READ #18b is used in place of #18a, and GV( $i$ ) is the trial energy read in for each level  $v = \text{IV}(i)$ ,  $J = \text{IJ}(i)$ . This option presumes NLEV1 > 0.

If expectation values or matrix elements are to be calculated (i.e., if LXPCT  $\neq 0$  or -1), READS #19-24 specify the desired arguments. However, if LXPCT = 0 or -1 the data input is now finished.

#19. READ(5,\*) MORDR, IRFN, RREF

**MORDR**: is an integer specifying the highest power of the chosen radial function or distance coordinate  $\text{RFN}(r)$  whose expectation values or matrix elements are to be calculated (see Eq. (7)). The current program version is dimensioned for  $\text{MORDR} \leq 20$ . To calculate only Franck-Condon factors (when  $|\text{LXPCT}| \geq 3$ ), input  $\text{MORDR} = -1$ .

**IRFN & RREF**: are integer and real variables, respectively, specifying the definition of the radial function or distance coordinate  $\text{RFN}(r)$ .

If  $\text{IRFN} \leq -10$ ,  $\text{RFN}$  is generated by user-supplied code inserted at line #498 of the main program. In this case  $\text{MORDR}$  and  $\text{RREF}$  are dummy variables, and  $\text{READS}$  #20-24 are ignored.

If  $\text{IRFN} = -4$ ,  $\text{RFN}(r) = r$  and the matrix element operator is a polynomial in  $r$  of order  $\text{MORDR}$ , with coefficients  $\text{DM}(i)$ , premultiplying the derivative operator  $\frac{d}{dr}$  which acts on the wavefunction for ‘Potential-2’.

If  $\text{IRFN} = -3$ ,  $\text{RFN}(r) = 1/r^3$ .

If  $\text{IRFN} = -2$ ,  $\text{RFN}(r) = 1/r^2$ .

If  $\text{IRFN} = -1$ , use a Dunham-type expansion coordinate  $\text{RFN}(r) = (r - r_{\text{ref}})/r_{\text{ref}}$ .

If  $\text{IRFN} = 0$ , the function  $\text{RFN}(r) = r$ , the distance coordinate itself.

If  $\text{IRFN} = 1 - 9$ , use the Šurkus-type variable  $\text{RFN}(r) = y_p(r) = (r^p - r_{\text{ref}}^p)/(r^p + r_{\text{ref}}^p)$ , with  $p = \text{IRFN}$ .

For  $\text{IRFN} = -1$  or  $1 - 9$ , a positive input value of  $\text{RREF}$  specifies it as the reference distance  $r_{\text{ref}}$  for these cases (usually set =  $r_e$ ). However, if the input value of  $\text{RREF} \leq 0.0$ , the program internally (iteratively) determines a value of  $r_{\text{ref}}$  such that the expectation value of  $\text{RFN}(r)$  is identically zero for the first vibration-rotation level considered.

If  $\text{IRFN} \geq 10$ ,  $\text{RFN}(r)$  is a function defined by reading in and interpolating over (and extrapolating beyond) input values of some known radial (e.g., a dipole or transition moment) function. This reading and interpolation/extrapolation is performed by the same subroutine package PREPOT used for treating a numerical input potential (see  $\text{READS}$  #5–8). In this case  $\text{MORDR}$  and  $\text{RREF}$  are dummy variables, and  $\text{READ}$  #20 is ignored.

#20. `IF(DABS(IRFN).LE.9) READ(5,*) (DM(J), J= 0,MORDR)`

**DM(j)**: Coefficients of the power series in  $\text{RFN}(r)$  defining the argument of the overall expectation values or matrix elements:  $M(r) = \sum_{j=0}^{\text{MORDR}} \text{DM}(j) \times \text{RFN}(r)^j$ .

If the expectation value or matrix element radial function argument is to be defined by interpolating over and extrapolating beyond a set of read-in points ( $\text{IRFN} \geq 10$ ), use the same read sequence, options and procedures employed for treating a numerical input potential. Most input parameters here have definitions essentially equivalent to those associated with  $\text{READ}$  #5–8.

#21. `READ(5,*) NRFN, RFLIM`

#22. `READ(5,*) NUSEF, ILRF, NCNF, CNNF`

#23. `READ(5,*) RFACTF, MFACTF`

#24. `READ(5,*) (XIF(I), YIF(I), I= 1,NRFN)`

**NRFN** is the number of known function values  $\{\text{XIF}(i), \text{YIF}(i)\}$  to be read in,

**RFLIM** is the limiting asymptotic value imposed when extrapolating beyond the range of the input values, and

**NUSEF** specifies how the interpolation is to be performed, while **ILRF**, **NCNF** and **CNNF** define the manner in which it extrapolates to large  $r$  (as in  $\text{READ}$  #6).

**RFACTF** & **MFACTF** convert units of input distances  $XIF(i)$  and ordinates  $YIF(i)$ , respectively, to Å and whatever units are required for the expectation value/matrix element argument  $M(r)$  (debye, for a dipole or transition moment function).

For matrix element calculations, couple each level of Potential-1, generated as specified by READs #17 & 18, to all rotation levels of the NLEV2 vibrational levels  $v = IV2(i)$  allowed by the rotational selection rules  $\Delta J = J2DL$  to  $J2DU$  in steps of  $J2DD$  (e.g., for  $P$  &  $R$  transitions:  $J2DL = -1$ ,  $J2DU = +1$  &  $J2DD = +2$ ). If  $NUMPOT = 2$  these are levels of Potential-2 and no constraints are imposed, but if  $NUMPOT = 1$  the matrix elements couple levels of Potential-1, and to avoid redundancy the program considers only emission from (rotational sublevels of) these NLEV2 vibrational levels into lower ( $v''$ ,  $J''$ ) levels generated as per READs #17 & 18. Integer  $AUTO2 > 0$  causes LEVEL to automatically generate trial eigenvalues for all desired levels (preferred option), so only their vibrational quantum number labels need be input (READ #26a). If this fails, setting  $AUTO2 \leq 0$  will require a trial pure vibrational energy  $GV(i)$  to be read in (READ #26b) for each specified level.

#25. READ(5,\*) NLEV2, AUTO2, J2DL, J2DU, J2DD

#26 a. IF(AUTO2.GT.0) READ(5,\*) (IV2(I), I= 1,NLEV2)

#26 b. IF(AUTO2.GT.0) READ(5,\*) (IV2(I), GV(I), I= 1,NLEV2)

## 5 Program Updates

This section briefly describes changes in the program's capabilities since Version 7.0 (January 2000). I am grateful to users for bringing a number of the program malfunctions corrected here to my attention.

### Major Updates in Versions 7.1 to 7.6

- The ability to search for the very highest bound level of a potential (e.g., for the one bound level of ground state  $He_2$ ) may now be explicitly selected by the user. This option is invoked by setting the following parameters in the input data file: the number of bound levels to be searched for should be set at  $NLEV1 = 1$  and the vibrational index read in for that level should be set at  $IV(1) = 999$ . This extension was required because the extreme case of a very weakly-bound level in a shallow potential was sometimes by missed by the general purpose 'level-finder' subroutine ALF.
- The variety of analytic potentials which can be generated by subroutine POTGEN in the PREPOT package has been further extended and generalized.
- The ability to calculate matrix elements of a function incorporating a radial first derivative operator may now be invoked by setting input parameter (READ #19)  $IRFN = -4$ . The matrix elements will then be of the operator  $f(r) \frac{d}{dr}$  where  $f(r)$  is a polynomial in  $r$  of order  $MORDR$ , with coefficients  $DM(i)$ .
- Parameter  $VSHIFT$  is now input through Read #7 (rather than READ #5).
- The expansion variable used in the MLJ (now MLR) and EMO analytic potential functions was generalized to the form  $y_p(r) = [(r)^p - (r_e)^p]/[(r)^p + (r_e)^p]$ , where  $p$  is a small integer whose value is input as the parameter  $p = MPAR$ .

### Updates New to Version 7.7 (1 June 2005)

- Experience showed that for potentials with unusual shapes or levels very near dissociation the code was not always sufficiently robust to generate eigenvalues for any specified quantum number, so parameters  $AUTO1$  and  $AUTO2$  (see READs #17 & 25) now give the user a choice of whether or not to rely on the code to automatically find all specified levels, or (if that fails) to read in a trial energy for each desired level.
- The choice of radial expansion variable used to define the argument in calculating expectation values or matrix elements has been expanded to include the generalized Šurkus-type variable  $y_p(r) = (r^p - r_e^p)/(r^p + r_e^p)$ .
- The Hönl-London factors used in the calculation of absolute Einstein emission coefficients for  $\Delta\Lambda = \pm 1$  transitions (see Eq. (8)) has been corrected, following Hansson and Watson [24].
- The atomic masses stored in data subroutine MASSES have been updated to the 2003 values of Ref. [47] and the physical constants have been updated to the 2005 values of Ref. [46].

- The analytic EMO and MLJ potential functions were generalized to allow different exponent expansion powers  $N_L$  and  $N_S = \text{NSR}$  for  $r > r_e$  and  $r < r_e$ , respectively, and the structure of the input data file was changed slightly by the addition of parameter `NSR` to `READ #9`. The ability to generate the Tiemann-type analytic potential of Eq. (24) was also added.
- The electronic angular momentum projection quantum number `OMEGA` is now input through `READ #5` (rather than #17 and 25), as it is a property of that state, and is more logically read in when the potential is being specified.

## Updates New to Version 8.0 (2 April 2007)

- **Coding errors** affecting the calculation of tunneling level widths, which had crept into LEVEL 7.5 (June 2002!) and persisted in distributed version of the code till March 11, 2007, have been corrected. The errors made the results a factor of 2-3 too small for very small widths, a factor of 2 too big for very large widths, and (fortunately!) accidentally became only a few % for widths in the range  $10^{-3}$  to  $5 \text{ cm}^{-1}$ .
- The ability to generate MLR and DELR analytic potential functions was added, the ordering and ‘IPOTL’ labeling of the various forms is changed, and the parameters used to define the nature of a particular analytic potential function (`READ #9`) was modified by addition of parameter `NCMM` and removal of `NCN`.
- The formulation of the `IPOTL = 4` option to generate an  $\text{MLJ}_p(N_S, N_L)$  potential has been extended to allow use of the Ogilvie-Tipping expansion variable  $y_{\text{OT}}(r) = 2(r - r_e)/(r + r_e)$  in the exponent expansion for  $\phi(r)$ . This option is invoked by setting parameter `MPAR = 0`.
- Illustrative sample data files for the various allowed types of analytic potential are included in the manual and in the distributed sample data file.
- The description of the calling sequence for a user-generate potential energy function subroutine on p. 8 is corrected.
- The description of the introduction of a user-implemented analytic matrix element argument, invoked by setting `IRFN ≤ -10`, has been clarified.
- For the cases of a user-supplied or pointwise matrix element argument function, ( $|\text{IRFN}| ≥ 10$ ), `MORDR` and `RREF` are now dummy variables, and the expansion coefficients `DM(i)` are no longer read in.
- An error in the definition of the Dunham or generalized Surkus variable used in matrix element calculations, which was introduced with version 7.7, has been corrected.

## Acknowledgements

I am pleased to acknowledge the foresight and stimulation provided by Professor George Burns of the University of Toronto who instigated my work on this type of computational research tool back in 1966. I must also acknowledge the inspiration and assistance provided by R.N. Zare’s seminal 1964 Franck-Condon intensity factor paper and the associated reports [1, 2, 3]. I thank J.Y. Seto for assistance in developing the level-finder subroutine `ALF`, and J.Y. Seto and G.T. Kraemer for assistance in developing the atomic mass database subroutine `MASSES`. I am also grateful to many users for comments which have identified errors and suggested useful extensions of the code, with particular thanks to R.W. Field, P.F. Bernath, P. Leyland, A.J. Ross, A.S. Dickinson and G.C. McBane. This work has been supported by the Natural Sciences and Engineering Research Council of Canada.

## References

- [1] R. N. Zare and J. K. Cashion, *The IBM Share Program D2 NU SCHR 1072 for Solution of the Schrödinger Radial Equation, by J.W. Cooley: Necessary and Useful Modifications for its Use on an IBM 7090*, University of California Lawrence Radiation Laboratory Report UCRL-10881 (1963).

- [2] R. N. Zare, University of California Lawrence Radiation Laboratory Report UCRL-10925 (1963).
- [3] R. N. Zare, *J. Chem. Phys.* **40**, 1934 (1964).
- [4] a) R. J. Le Roy, LEVEL 1.0, University of Wisconsin Theoretical Chemistry Institute Report WIS-TCI-429G (1971). b) R. J. Le Roy, LEVEL 2.0 University of Waterloo Chemical Physics Research Report CP-58 (1976). c) *ibid*, LEVEL 3.0, CP-110 (1979). d) *ibid*, LEVEL 4.0, CP-230 (1983). e) *ibid*, LEVEL 5.0, CP-330 (1991). f) *ibid*, LEVEL 5.1, CP-330R (1992). g) *ibid*, LEVEL 5.2 CP-330R<sup>2</sup> (1993). h) *ibid*, LEVEL 5.3, CP-330R<sup>3</sup> (1994). i) *ibid*, LEVEL 6.0, CP-555 (1995). j) *ibid*, LEVEL 6.1, CP-555R (1996) k) *ibid*, LEVEL 7.0 – 7.7, CP-642 – 642R<sup>3</sup>, CP-655 & CP-661 (2000 – 2005).
- [5] R. J. Le Roy, LEVEL 8.0: *A Computer Program for Solving the Radial Schrödinger Equation for Bound and Quasibound Levels*, University of Waterloo Chemical Physics Research Report CP-663 (2007); see <http://leroy.uwaterloo.ca/programs/>.
- [6] J. K. G. Watson, *J. Mol. Spectrosc.* **80**, 411 (1980).
- [7] J. W. Cooley, *Math. Computations* **15**, 363 (1961).
- [8] J. Cashion, *J. Chem. Phys.* **39**, 1872 (1963).
- [9] R. J. Le Roy and R. B. Bernstein, *J. Chem. Phys.* **54**, 5114 (1971).
- [10] R. J. Le Roy and W.-K. Liu, *J. Chem. Phys.* **69**, 3622 (1978).
- [11] J. N. L. Connor and A. D. Smith, *Mol. Phys.* **43**, 397 (1981).
- [12] Y. Huang and R. J. Le Roy, *J. Chem. Phys.* **119**, 7398 (2003); erratum: *ibid* **126** 169904 (2007).
- [13] K. Smith, *The Calculation of Atomic Collision Processes* (Wiley-Interscience, New York, 1971), chapter 4.
- [14] M. S. Child, *Semiclassical Mechanics with Molecular Applications* (Clarendon Press, Oxford, 1991).
- [15] P.-O. Löwdin, *An Elementary Iteration-Variation Procedure for Solving the Schrödinger Equation*, Technical Note No. 11, Quantum Chemistry Group, Uppsala University, Uppsala, Sweden (1958).
- [16] M. Hehenberger, P. Froelich, and E. Brändas, *J. Chem. Phys.* **65**, 4559 (1976).
- [17] M. Hehenberger, *J. Chem. Phys.* **67**, 1710 (1977).
- [18] R. J. Le Roy, *Comp. Phys. Comm.* **52**, 383 (1989).
- [19] R. J. Le Roy and G. T. Kraemer, BCONT 2.2: *A Computer Program for Calculating Bound→Continuum Transition Intensities for Diatomic Molecules*, University of Waterloo Chemical Physics Research Report CP-650R<sup>2</sup> (2004); see <http://leroy.uwaterloo.ca/programs/>.
- [20] G. Herzberg, *Spectra of Diatomic Molecules* (Van Nostrand, New York, 1950).
- [21] J. Tellinghuisen, *J. Mol. Spectrosc.* **122**, 455 (1987).
- [22] a) J. M. Hutson, *J. Phys. B. (At. Mol. Phys.)* **14**, 851–857 (1981); b) J. M. Hutson, QCPE Bulletin, **2**, no. 2, Program #435, Quantum Chemistry Program Exchange, Indiana University, Bloomington, Indiana.
- [23] P. F. Bernath, *Spectra of Atoms and Molecules*, 2nd ed. (Oxford University Press, New York, 2005).
- [24] A. Hansson and J. K. G. Watson, *J. Mol. Spectrosc.* **233**, 169 (2005).
- [25] J. Y. Seto, *Direct Fitting of Analytic Potential Functions to Diatomic Molecule Spectroscopic Data*, M.Sc. Thesis, Department of Chemistry, University of Waterloo (2000).
- [26] A. A. Šurkus, R. J. Rakauskas, and A. B. Bolotin, *Chem. Phys. Lett.* **105**, 291 (1984).
- [27] J. L. Dunham, *Phys. Rev.* **41**, 721 (1932).
- [28] G. Simons, R. G. Parr, and J. M. Finlan, *J. Chem. Phys.* **59**, 3229 (1973).
- [29] J. F. Ogilvie, *Proc. Roy. Soc. (London)* **A 378**, 287 (1981).
- [30] E. G. Lee *et al.*, *J. Mol. Spectrosc.* **194**, 197 (1999).
- [31] J. Y. Seto *et al.*, *J. Chem. Phys.* **110**, 11756 (1999).



- [32] Y. Huang, *Determining Analytical Potential Energy Functions of Diatomic Molecules by Direct Fitting*, M.Sc. Thesis, Department of Chemistry, University of Waterloo (2001).
- [33] W. Hua, Phys. Rev. A **42**, 2524 (1990).
- [34] R. J. Le Roy and R. D. E. Henderson, Mol. Phys. **105**, 663 (2007).
- [35] P. G. Hajigeorgiou and R. J. Le Roy, J. Chem. Phys. **112**, 3949 (2000).
- [36] R. J. Le Roy, Y. Huang, and C. Jary, J. Chem. Phys. **125**, 164310 (2006), N2.
- [37] K. T. Tang and J. P. Toennies, J. Chem. Phys. **80**, 3726 (1984).
- [38] C. Douketis, J. M. Hutson, B. J. Orr, and G. Scoles, Mol. Phys. **52**, 763 (1984).
- [39] J. Hepburn, G. Scoles, and R. Penco, Chem. Phys. Lett. **36**, 451 (1975).
- [40] C. Douketis *et al.*, J. Chem. Phys. **76**, 3057 (1982).
- [41] C. Samuelis *et al.*, Phys. Rev. A **63**, 012710 (2000).
- [42] R. J. Le Roy, in *Molecular Spectroscopy*, edited by R. N. Barrow, D. A. Long, and D. J. Millen (Chemical Society of London, London, 1973), Vol. 1, Specialist Periodical Report 3, pp. 113–176.
- [43] R. J. Le Roy, J. Chem. Phys. **73**, 6003 (1980).
- [44] R. J. Le Roy, J. Chem. Phys. **101**, 10217 (1994).
- [45] R. J. Le Roy and Y. Huang, J. Mol. Struct. (Theochem) **591**, 175 (2002).
- [46] P. J. Mohr and B. N. Taylor, Rev. Mod. Phys. **77**, 1 (2005).
- [47] G. Audi, A. H. Wapstra, and C. Thibault, Nucl. Phys. **A 729**, 337 (2003).
- [48] J. A. Coxon and P. G. Hajigeorgiou, J. Mol. Spectrosc. **139**, 84 (1990).
- [49] J. A. Coxon and P. G. Hajigeorgiou, J. Mol. Spectrosc. **150**, 1 (1991).
- [50] J. Y. Seto, R. J. Le Roy, J. Vergès, and C. Amiot, J. Chem. Phys. **113**, 3067 (2000).
- [51] R. J. Le Roy, J. Mol. Spectrosc. **194**, 189 (1999).
- [52] J. A. Coxon and P. G. Hajigeorgiou, J. Phys. Chem. **A 110**, 6261 (2006).
- [53] S. D. Peyerimhoff and R. J. Buenker, Chem. Phys. **57**, 279 (1981).
- [54] R. J. Le Roy *et al.*, J. Chem. Phys. **123**, 204304/1 (2005).

## Appendix: Program Structure, Illustrative Cases, Sample data and output files, and timing

### A. Program Structure

The present section lists the names and outlines the functions of the various subroutines used by LEVEL, and indicates their hierarchy. In particular, the level of indentation in this list indicates which subroutines call which others; unless stated otherwise, each subroutine is called exclusively by the immediately preceding routine having one lower level of indentation.

**LEVEL**: The main program which reads most input data, and calls the potential preparation, eigenfunction determination, and overlap integral routines.

**MASSSES**: A data subroutine containing accurate atomic masses and other properties of all stable atomic isotopes. For normal cases, its presence obviates the need for a user to look up and type precise particle masses into the input data file.

**ALF**: For any smooth single-minimum, shelf-state or double-minimum potential, ALF (Automatic Level Finder) uses multiple calls to subroutine SCHRQ (see below) to determine the vibrational energies of all levels from  $v = 0$  up to some maximum  $v$  specified by its input parameter KVMAX. If fewer than the specified number of levels can be found, appropriate warnings are printed.

**SCHRQ**: Solves the Schrödinger equation to determine the eigenvalue and (unit normalized) eigenfunction of the vibrational level lying closest to the input trial energy.

**QBOUND**: For quasibound levels (those lying above the potential asymptote, but behind a potential barrier), applies the Airy function boundary condition at the third turning point to initiate the inward inward integration of the wave function for such levels [9, 10].

**WIDTH**: Calculates the tunneling predissociation lifetime or width of a quasibound level [10, 11, 12].

**LEVQAD**: Called by WIDTH to evaluate the near-turning-point contributions to the semiclassical quadratures over the potential well and the barrier to tunneling which are required for calculating the predissociation rate.

**CDJOEL**: Calculates the diatomic molecule centrifugal distortion constants of Eq. (6). The required input is the effective (centrifugally-distorted, if appropriate) radial potential, and the eigenvalue and eigenfunction of the level in question (as calculated by SCHRQ).

**LEVXPC**: Calculates the desired diagonal expectation values of powers of the specified distance coordinate or (interpolated) radial function  $RFN(r)$  (see READ #16 in Section 4).

**MATXEL**: Calculates the desired off-diagonal matrix elements of powers of the specified radial function or distance coordinate, and the radiative lifetime or Einstein emission coefficient  $\mathcal{A} \text{ s}^{-1}$ .

**PREPOT**: The subroutine which oversees reading of the parameters defining the potential and generating the required potential array at the  $N$  distances specified by the input distance array.

**GENINT**: Uses piecewise polynomial or cubic spline functions to interpolate over a set of read-in turning points to yield the potential array at the equally-spaced radial mesh required by SCHRQ. If necessary it extrapolates beyond range of the given points with analytic functions (see options for parameter ILR of READ #6).

**PLYINTRP**: Performs piecewise polynomial interpolation on a given array of points and returns the function value and (if desired) its derivatives at a specified point. It is used by GENINT to interpolate for the potential for cases with NUSE > 0. This routine is also used by SPLINT to defining the first derivatives at the inner and outer ends of the interpolation region,  $R = XI(1)$  and  $XI(NTP)$ , respectively.

**SPLINT**: Calls subroutine SPLINE to generate the  $4 \times (NTP - 1)$  coefficients required to define the cubic spline through the  $NTP$  given turning points, and then uses these coefficients to generate the desired interpolated function values at the specified integration mesh points.

**SPLINE**: Generates the coefficients defining the cubic spline function through the given  $NTP$  turning points.

**POTGEN**: For the analytic potential function specified by input parameter IPOTL (see READ #9), generates the required array of potential function values on the specified radial grid.

**DAMPF**: Generates the damping functions of Eqs. (21) and (22) utilized in defining the DELR potential function of Eq. (17).

## B. Illustrative Cases

The running time for this program will depend entirely on the complexity of the calculation being performed and the type of computer being used; CPU requirements may range from a fraction of a second to a few minutes. This Appendix presents sample data files and the resulting output for representative cases which illustrate some of the types of problems to which the program may be applied.

**Case 1**: This data file (see Appendix C) consists of five separate data sets which illustrate a variety of the capabilities of the program, including the fact that it can treat several independent problems in a single run by simply putting the input data for several cases into the same file. Parts of the resulting output are listed in Appendix D. The total CPU time required for these 5 cases on an old (circa 1999) workstation was 1.2 sec.

- a) Determine eigenvalues for vibrational levels  $v = 56 - 75$  of an *ab initio* potential for the  ${}^3\Pi_g$  state of  $Cl_2$  which has asymmetric double minima [53], and calculate expectation values of various powers of  $r$ . The output expectation values show that the wave function amplitude hops back and forth between the two wells with increasing energy. This case illustrates the ability of the program to automatically find arbitrary levels of a double minimum potential. The “ALF ERROR” warning messages seen in the output merely indicate that the automatic level-finder subroutine “ALF” encounters some internal challenges (which it overcomes!) in climbing a ladder of levels which hop from one well to the other.
- b) For a Lennard-Jones(12,6) potential, find all the vibrational levels and calculate the associated values of the centrifugal distortion constants. This is a model system with fictitious particle masses, so those masses and the chosen particle names ‘L1’ and ‘J2’ are input by READ #2.
- c) For the same simple model Lennard-Jones(12,6) potential of Case 1.b), calculate all possible infrared matrix elements involving levels with  $v \leq 2$  and  $J(\text{upper}) \leq 1$ .
- d) For the same model Lennard-Jones(12,6) potential of Case 1.b), locate all (four) vibrational levels of the centrifugally-distorted potential associated with  $J = 18$ . This demonstrates that the procedure for automatically finding all vibrational levels works for a potential with a barrier (here, centrifugal in origin) protruding above its dissociation limit. This example invokes the highest print level inside SCHRQ (by setting IWR = 3, see READ #17) in order to illustrate the progress of the iterative eigenvalue convergence procedure. In this case convergence details for each level are presented twice (though only one iteration is required the second time) because of the internal program logic.
- e) To illustrate use of one of the more sophisticated analytic potential forms and incorporation of Born-Oppenheimer breakdown (BOB) contributions to the radial and centrifugal potentials, this case calculates properties of levels of the ground state of  ${}^{109}\text{AgD}$  using the analytic  $\text{EMO}_p$

potential and BOB correction functions determined from a combined isotopologue analysis of data for the AgH system in which  $^{107}\text{AgH}$  was the “reference isotopologue” (see Ref. [54]). Note that in such cases, calculation of the rotational constants is based on a centrifugal potential which includes the BOB correction function of Eq. (26).

**Case 2.** Illustrative input data files are provided for eigenvalue calculations using each of the  $\text{IPOTL} \geq 4$  analytic potential energy functions described in §2.6 (Appendix E). The total CPU time required for these 6 cases on an old (circa 1999) workstation was 5.2 sec.

**Case 3.** This is an illustrative “production run” which generates the predicted frequencies and Einstein emission coefficients for some 156 135 lines in the  $B(^3\Pi_{0u}^+) - X(^1\Sigma_g^+)$  spectrum of  $\text{Br}_2$ . This case again exploits the capability of the  $\text{NLEV1} < 0$  option for automatically finding the first  $|\text{NLEV1}| + 1$  levels of a given potential. It also illustrates a use of the  $\text{NJM} > 0$  option for finding many (or all) rotational sublevels for each  $v$ , and of the application of rotational selection rules in calculating transitions between two different electronic states. The input data file is listed in Appendix F, parts of the channel-6 output in Appendix G and a portion of the associated channel-8 output is in Appendix H. The CPU time required for this case on an old (circa 1999) workstation was 77 sec.

The largest piece of output for this case consists of a listing of transition frequencies and Einstein  $\mathcal{A}$ -coefficients for some 156 135 transitions (yielding that number of lines of text) written to channel-8. A small segment of this output file is shown in Appendix H. Since the associated “standard” output to channel-6 is also moderately lengthy (some 1655 lines), the listing of it presented in Appendix G is also considerably truncated. However, this output illustrates some of the (often not serious) warning messages which may be generated by the program. In particular, beginning at  $v' = 28$ ,  $J' = 127$  (and seen in the truncated listing of Appendix G for  $v = 34$ ) warnings occur for quasibound levels for which  $\text{RMAX}$  is smaller than the outermost turning point at which the program was attempting to apply the Airy function boundary condition (see Section 2.B). The first of these warnings is printed once each iteration as  $\text{SCHRQ}$  converges on the associated eigenvalue; as it indicates, in this case the Airy function boundary condition is replaced by use of the WKB wave function initialization of Eq. (3). The second type of warning seen here describes the approximation used to estimate that portion of the exponent integral determining the semiclassical tunneling probability (Eq. (4) with  $r_{\text{end}}$  replaced by  $r_3$ ) which is due to the potential barrier at distances beyond  $\text{RMAX}$ . When this situation arises the associated tunneling level widths are usually extremely small, and although the resulting predicted width may be slightly in error the calculated eigenvalue is usually extremely accurate.

A different type of problem gives rise to the lines beginning “ $\text{SCHRQ}$  has a ...” and “CAUTION for ...” seen here for  $v = 34$ ,  $J = 127$ . This is the highest quasibound level for this  $v$ ; it lies very close to the centrifugal barrier maximum and the Airy function boundary condition was unable to achieve full convergence to  $\text{EPS}$  for this particular level. In spite of this problem, however, the last eigenvalue change of “ $\text{DE} = 1.63\text{D-}03$ ”  $\text{cm}^{-1}$  is considerably smaller than the width ( $\text{FWHM} = 0.90 \text{ cm}^{-1}$ ) of this tunneling predissociation level, so the lack of full convergence has negligible effect on the real accuracy of the resulting eigenvalue.

Another type of warning message is that seen at  $v = 34$  for  $J = 128$ , and for  $v = 40$  for  $J \geq 92$  (“... find onee turn point: ...”). It appears when the automatic search for ever higher rotational sublevels goes past the top of the (centrifugal) barrier maximum. In this case the code tries one more time to place a level marginally below that barrier, a step which sometimes fails (as it did for  $v = 34$ ,  $J = 128$ ) and sometimes succeeds (as it did for  $v = 40$ ,  $J = 92$ ).

## C. Input data file for Case 1

Note that entries after the “%” sign on each line of the input data file are comments identifying the variables, and are ignored by the program.

```

17 35 17 35 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 1.a: Levels of double minimum potential for 3(PI)g Cl2'
0.001 1.6 99. 1.d-06 % RH RMIN RMAX EPS
29 0 0 95440.DO % NTP LPPOT IOMEG VLIM
0 0 2 1 0.D5 % NUSE IR2 ILR NCN CNM
0.5291772108D0 8065.5444D0 0.d0 % RFACT EFACT VSHIFT
3.34 10.608 3.40 9.683 3.43 9.221 3.49 8.758
3.57 8.295 3.66 8.132 3.76 8.051 3.86 8.132 3.94 8.268
4.03 8.350 4.17 8.595 4.26 8.704 4.31 8.758 4.46 8.432
4.56 8.134 4.66 7.887 4.80 7.615 4.91 7.425 5.00 7.343
5.17 7.125 5.37 6.962 5.51 6.908 5.66 6.880 5.80 6.908
5.94 6.989 6.00 7.044 6.20 7.125 6.40 7.261 6.60 7.425
15 1 0 1 0 1 -0 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
56 0 57 0 58 0 59 0 60 0 61 0 62 0 63 0 64 0 65 0 66 0
67 0 68 0 69 0 70 0 % IV(i) IJ(i)
3 0 0.DO % MORDR IRFN RREF
1.d0 -2.d-1 3.d-2 -4.d-3 % {DM(j)}

0 30 0 30 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'L1' 30.34373256D0 % NAME1 MASS1
'J2' 30.34373256D0 % NAME2 MASS2
'Case 1.b: For a model L.J.(12,6) get all vib. levels and their CDCs'
0.0020 0.6 99. 1.d-6 % RH RMIN RMAX EPS
0 0 0 0.DO % NTP LPPOT IOMEG VLIM
1 6 12 0 0 0 1.D3 1.DO % IPOTL MPAR NSR NVARB NCMM IBOB DSCM REQ
-998 1 2 -1 0 1 -1 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 0 % IV(1) IJ(1)

0 30 0 30 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'L1' 30.34373256D0 % NAME1 MASS1
'J2' 30.34373256D0 % NAME2 MASS2
'Case 1.c: For same model L.J.(12,6) get radial matrix elements for v < 3'
0.0020 0.6 99. 1.d-6 % RH RMIN RMAX EPS
0 0 0 0.DO % NTP LPPOT IOMEG VLIM
1 6 12 0 0 0 1.D3 1.DO % IPOTL MPAR NSR NVARB NCMM IBOB DSCM REQ
-2 1 0 5 1 1 -1 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 0 % IV(1) IJ(1)
4 -1 1.DO % MORDR IRFN RREF
1.d0 -0.2d0 0.03d0 -0.004d0 0.0005d0 % {DM(j)}
3 1 -1 +1 2 % NLEV2 AUTO2 J2DL J2DU J2DD
0 1 2 % IV2(1) IV2(2) IV2(3)

-1 0 -1 0 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'L1' 30.34373256D0 % NAME1 MASS1
'J2' 30.34373256D0 % NAME2 MASS2
'Case 1.d: For the same L.J.(12,6), find levels of potential with a barrier'
0.0020 0.6 99. 1.d-6 % RH RMIN RMAX EPS
0 0 0 0.DO % NTP LPPOT IOMEG VLIM
1 6 12 0 0 0 1.D3 1.DO % IPOTL MPAR NSR NVARB NCMM IBOB DSCM REQ
-999 1 0 -1 0 1 5 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 18 % IV(1) IJ(1)

47 109 1 2 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 1.e: X-state {109}AgD using EM03(7,7) potential [JCP 123, 204304(2005)]'
0.0010 0.8 9.0 1.d-6 % RH RMIN RMAX EPS
0 0 0 19250.0d0 % NTP LPPOT IOMEG VLIM
3 3 7 0 8 1 19250.d0 1.6179162d0 % IPOTL MPAR NSR NCMM NVARB IBOB DSCM REQ
1.54358095D+00 3.73860D-02 1.66424D-01 9.8030D-02 1.7089D-01 6.0200D-02
1.4000D-01 2.2400D-01
107 1 3 3 -1 3 3 -1 2 % MN1R MN2R PAD MAD NU1 NU2 PNA NT1 NT2
0.0D+0 0.0D+0 1.175D+01 1.756D+01 -1.270D+01 % U2INF CA2(0) CA2(1) ...
0.0D+0 0.0D+0 1.5D-04 9.3D-04 % G2INF GA2(0) GA2(1) ...
-15 1 2 -1 0 1 -1 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 0 % IV(1) IJ(1)

```

## D. Channel-6 Output file for Case 1

```

Case 1.a: Levels of double minimum potential for 3(PI)g Cl2
=====
Generate ZMU= 17.48442634000(u) & BZ= 1.037181808D+00((1/cm-1)(1/Ang**2))
      from atomic masses: 34.96885268000 & 34.96885268000(u)
Integrate from RMIN= 1.600 to RMAX= 81.60 with mesh RH= 0.001000(Angst)

Potential #1 for Cl( 35)-Cl( 35)
=====
State has OMEGA= 0 and energy asymptote: Y(lim)= 95440.0000(cm-1)
Perform cubic spline interpolation over the 29 input points
To make input points Y(i) consistent with Y(lim), add Y(shift)= 0.0000
Scale input points: (distance)* 5.291772108D-01 & (energy)* 8.065544400D+03
      to get required internal units [Angstroms & cm-1 for potentials]
-----
      R(i)      Y(i)      R(i)      Y(i)      R(i)      Y(i)
-----
3.340000 10.60800000  4.170000  8.59500000  5.370000  6.96200000
3.400000  9.68300000  4.260000  8.70400000  5.510000  6.90800000
3.430000  9.22100000  4.310000  8.75800000  5.660000  6.88000000
3.490000  8.75800000  4.460000  8.43200000  5.800000  6.90800000
3.570000  8.29500000  4.560000  8.13400000  5.940000  6.98900000
3.660000  8.13200000  4.660000  7.88700000  6.000000  7.04400000
3.760000  8.05100000  4.800000  7.61500000  6.200000  7.12500000
3.860000  8.13200000  4.910000  7.42500000  6.400000  7.26100000
3.940000  8.26800000  5.000000  7.34300000  6.600000  7.42500000
4.030000  8.35000000  5.170000  7.12500000
-----
Extrapolate to X .le. 1.7992 with
Y= -5091495.587 +5.609801D+06 * exp(-4.542065D-02*X)
Extrapolate to X .GE. 3.3867 using
Y= 95440.0000 - [ 1.128770D+05/X**1 +1.377826D+05/X**3]
-----

Calculate properties of the single potential described above
Potential-1 uses inner boundary condition of zero value at RMIN

Eigenvalue convergence criterion is EPS= 1.0D-06(cm-1)
Airy function at 3-rd turning point is quasibound outer boundary condition

State-1 electronic angular momentum OMEGA= 0
      yields centrifugal potential [J*(J+1) - 0.00]/r**2

Solve for the 15 vibration-rotation levels of Potential-1:
(v,J) = ( 56, 0) ( 57, 0) ( 58, 0) ( 59, 0) ( 60, 0) ( 61, 0)
      ( 62, 0) ( 63, 0) ( 64, 0) ( 65, 0) ( 66, 0) ( 67, 0)
      ( 68, 0) ( 69, 0) ( 70, 0)

Matrix element arguments are powers of the distance r (in Angstroms)
Coefficients of expansion for radial matrix element/expectation value argument:
      1.000000D+00 -2.000000D-01 3.000000D-02 -4.000000D-03

*** ALF ERROR ***
Attempt to find next higher vibrational level fails!
Use of differences to estimate the energy for the next
vibrational level (v= 38) failed after 1 attempt.

*** ALF ERROR ***
Attempt to find next higher vibrational level fails!
Use of differences to estimate the energy for the next
vibrational level (v= 48) failed after 1 attempt.

*** ALF ERROR ***
Attempt to find next higher vibrational level fails!
Use of differences to estimate the energy for the next
vibrational level (v= 53) failed after 1 attempt.

*** ALF ERROR ***
Attempt to find next higher vibrational level fails!
Use of differences to estimate the energy for the next
vibrational level (v= 63) failed after 1 attempt.

*** ALF WARNING ***
Potential found to have a second minimum.
The highest calculated level is E(v= 70) = 70022.038
E(v= 56, J= 0)= 68214.438 <M(r)>= 0.4721236994 <KE>= 4580.196
      <X** 1>= 3.63045127 <X** 2>= 13.66002892 <X** 3>= 52.89672851
-----
E(v= 57, J= 0)= 68387.998 <M(r)>= 0.4695211055 <KE>= 4617.063
      <X** 1>= 3.64732349 <X** 2>= 13.79516352 <X** 3>= 53.71727532
-----
E(v= 58, J= 0)= 68476.426 <M(r)>= 0.6824612177 <KE>= 1651.393
      <X** 1>= 2.04384519 <X** 2>= 4.18845602 <X** 3>= 8.60585634
-----
E(v= 59, J= 0)= 68559.695 <M(r)>= 0.4669189762 <KE>= 4652.829
      <X** 1>= 3.66408683 <X** 2>= 13.93033264 <X** 3>= 54.54340916
-----
E(v= 60, J= 0)= 68729.539 <M(r)>= 0.4643045293 <KE>= 4686.656
      <X** 1>= 3.68083020 <X** 2>= 14.06617695 <X** 3>= 55.37868484
-----
E(v= 61, J= 0)= 68897.547 <M(r)>= 0.4617011736 <KE>= 4718.810
      <X** 1>= 3.69738156 <X** 2>= 14.20154715 <X** 3>= 56.21723206
-----
E(v= 62, J= 0)= 69063.700 <M(r)>= 0.4591313915 <KE>= 4749.186
      <X** 1>= 3.71357576 <X** 2>= 14.33535705 <X** 3>= 57.05354179
-----
E(v= 63, J= 0)= 69214.948 <M(r)>= 0.6781183394 <KE>= 1985.878
      <X** 1>= 2.07712475 <X** 2>= 4.36259569 <X** 3>= 9.33364518

```

```

-----
E(v= 64, J= 0)= 69228.156 <M(r)>= 0.4588457946 <KE>= 4751.128
<X** 1>= 3.71260667 <X** 2>= 14.36531491 <X** 3>= 57.39807953
-----
E(v= 65, J= 0)= 69390.488 <M(r)>= 0.4540380380 <KE>= 4803.188
<X** 1>= 3.74528392 <X** 2>= 14.60094614 <X** 3>= 58.73339038
-----
E(v= 66, J= 0)= 69551.030 <M(r)>= 0.4516314964 <KE>= 4825.287
<X** 1>= 3.75994565 <X** 2>= 14.72704336 <X** 3>= 59.54766856
-----
E(v= 67, J= 0)= 69709.530 <M(r)>= 0.4496202586 <KE>= 4837.877
<X** 1>= 3.77158473 <X** 2>= 14.83463802 <X** 3>= 60.27548393
-----
E(v= 68, J= 0)= 69864.123 <M(r)>= 0.4631813280 <KE>= 4633.896
<X** 1>= 3.66813652 <X** 2>= 14.22170340 <X** 3>= 57.46061757
-----
E(v= 69, J= 0)= 69900.180 <M(r)>= 0.6579325797 <KE>= 2352.457
<X** 1>= 2.22764739 <X** 2>= 5.25807534 <X** 3>= 13.57005040
-----
E(v= 70, J= 0)= 70022.038 <M(r)>= 0.4464976857 <KE>= 4867.346
<X** 1>= 3.78784877 <X** 2>= 15.00980662 <X** 3>= 61.55668996
-----

```

Find the 15 vibration-rotation levels:

v	J	E(v)	v	J	E(v)	v	J	E(v)
56	0	68214.43842	61	0	68897.54686	66	0	69551.03001
57	0	68387.99804	62	0	69063.70032	67	0	69709.52996
58	0	68476.42603	63	0	69214.94841	68	0	69864.12275
59	0	68559.69491	64	0	69228.15645	69	0	69900.18008
60	0	68729.53947	65	0	69390.48802	70	0	70022.03777

Case 1.b: For a model L.J.(12,6) get all vib. levels and their CDCs

```

-----
Generate ZMU= 15.17186628000(u) & BZ= 9.000000000D-01((1/cm-1)(1/Ang**2))
from atomic masses: 30.34373256000 & 30.34373256000(u)
Integrate from RMIN= 0.600 to RMAX= 99.00 with mesh RH= 0.002000(Angst)

```

Potential #1 for L1( 30)-J2( 30)

State has OMEGA= 0 and energy asymptote: Y(lim)= 0.0000(cm-1)

Lennard-Jones(12, 6) potential with De= 1000.000(cm-1) Re = 1.000000(A)

Calculate properties of the single potential described above  
 Potential-1 uses inner boundary condition of zero value at RMIN

Eigenvalue convergence criterion is EPS= 1.0D-06(cm-1)  
 Airy function at 3-rd turning point is quasibound outer boundary condition

State-1 electronic angular momentum OMEGA= 0  
 yields centrifugal potential [J\*(J+1) - 0.00]/r\*\*2

For J= 0, try to find the first 400 vibrational levels of Potential-1

```

ALF finds all J= 0 vib. levels below vD= 6.631 estimated by N-D theory
The highest calculated level is E(v= 6) =-0.77249285
E(v= 0, J= 0)= -811.519 Bv= 1.0540426 -Dv= -4.1381D-05 Hv= -6.5120D-09
Lv= -1.9007D-12 Mv= -7.2230D-16 Nv= -3.1926D-19 Ov= -1.5585D-22
E(v= 1, J= 0)= -507.170 Bv= 0.9332450 -Dv= -6.0794D-05 Hv= -1.4820D-08
Lv= -6.5533D-12 Mv= -3.7208D-15 Nv= -2.4359D-18 Ov= -1.7523D-21
E(v= 2, J= 0)= -287.838 Bv= 0.8013476 -Dv= -9.2177D-05 Hv= -3.5187D-08
Lv= -2.3947D-11 Mv= -2.0879D-14 Nv= -2.1041D-17 Ov= -2.3384D-20
E(v= 3, J= 0)= -141.503 Bv= 0.6566553 -Dv= -1.4627D-04 Hv= -9.2303D-08
Lv= -1.0432D-10 Mv= -1.5291D-13 Nv= -2.6179D-16 Ov= -4.9802D-19
E(v= 4, J= 0)= -54.890 Bv= 0.4974391 -Dv= -2.4978D-04 Hv= -2.9569D-07
Lv= -6.5212D-10 Mv= -1.9135D-12 Nv= -6.6507D-15 Ov= -2.5885D-17
E(v= 5, J= 0)= -13.337 Bv= 0.3221808 -Dv= -4.9470D-04 Hv= -1.4933D-06
Lv= -9.1824D-09 Mv= -7.7465D-11 Nv= -7.8195D-13 Ov= -8.8619D-15
E(v= 6, J= 0)= -0.772 Bv= 0.1290036 -Dv= -1.6478D-03 Hv= -3.8715D-05
Lv= -2.0424D-06 Mv= -1.4479D-07 Nv= -1.1986D-08 Ov= -1.0942D-09

```

v	E(v)	v	E(v)	v	E(v)	v	E(v)
0	-811.5192	2	-287.8379	4	-54.8902	6	-0.7725
1	-507.1696	3	-141.5034	5	-13.3368		

An n= 6 N-D theory extrapolation from last 2 levels implies vD = 6.631

Case 1.c: For same model L.J.(12,6) get radial matrix elements for v < 3

```

-----
Generate ZMU= 15.17186628000(u) & BZ= 9.000000000D-01((1/cm-1)(1/Ang**2))
from atomic masses: 30.34373256000 & 30.34373256000(u)
Integrate from RMIN= 0.600 to RMAX= 99.00 with mesh RH= 0.002000(Angst)

```

Potential #1 for L1( 30)-J2( 30)

State has OMEGA= 0 and energy asymptote: Y(lim)= 0.0000(cm-1)

Lennard-Jones(12, 6) potential with De= 1000.000(cm-1) Re = 1.000000(A)

```

-----
Calculate properties of the single potential described above
Potential-1 uses inner boundary condition of zero value at RMIN

Eigenvalue convergence criterion is EPS= 1.0D-06(cm-1)
Airy function at 3-rd turning point is quasibound outer boundary condition

State-1 electronic angular momentum OMEGA= 0
      yields centrifugal potential [J*(J+1) - 0.00]/r**2
For J= 0, try to find the first 3 vibrational levels of Potential-1
      and automatically increment J in steps of 1 to a maximum value of 1

Matrix element argument expansion variable is: X = (r - RREF)/RREF
      where reference length is held fixed at RREF = 1.0000000000(Angstroms)
Coefficients of expansion for radial matrix element/expectation value argument:
      1.000000D+00 -2.000000D-01 3.000000D-02 -4.000000D-03 5.000000D-04

Using the rotational selection rule: delta(J)= -1 to 1 with increment 2
calculate matrix elements for coupling to the 3 vibrational levels of
Potential-2: v = 0 1 2
-----
Coupling E(v= 0, J= 1)= -809.4112 to E(v= 0, J= 0)= -811.5192
      Moment matrix elements: <X** 0>= 0.9999997392 <X** 1>= 0.0315663983
      <X** 2>= 0.0043724231 <X** 3>= 0.0004274505 <X** 4>= 0.0000681864
FCF= 1.0000D+00 <M>= 9.93816D-01 d(E)= -2.11 A(Einst)= 9.6706D-07 s-1
+++++
For vibrational level v = 0 of Potential-1
      J E J E J E J E
-----
0 -811.519 1 -809.411
+++++
Coupling E(v= 1, J= 0)= -507.1696 to E(v= 0, J= 1)= -809.4112
      Moment matrix elements: <X** 0>= 0.0007038706 <X** 1>= 0.055644965
      <X** 2>= 0.0061799007 <X** 3>= 0.0011668795 <X** 4>= 0.0002051809
FCF= 4.9543D-07 <M>=-1.02282D-02 d(E)= -302.24 A(Einst)= 9.0586D-04 s-1
+++++
Coupling E(v= 1, J= 1)= -505.3034 to E(v= 0, J= 0)= -811.5192
      Moment matrix elements: <X** 0>= -0.0007036644 <X** 1>= 0.0554599642
      <X** 2>= 0.0061672593 <X** 3>= 0.0011650604 <X** 4>= 0.0002048749
FCF= 4.9514D-07 <M>=-1.16152D-02 d(E)= -306.22 A(Einst)= 4.0496D-04 s-1
+++++
Coupling E(v= 1, J= 1)= -505.3034 to E(v= 0, J= 2)= -805.1964
      Moment matrix elements: <X** 0>= 0.0014096829 <X** 1>= 0.0556429498
      <X** 2>= 0.0062065908 <X** 3>= 0.0011721096 <X** 4>= 0.0002064159
FCF= 1.9872D-06 <M>=-9.53729D-03 d(E)= -299.89 A(Einst)= 5.1293D-04 s-1
+++++
Coupling E(v= 1, J= 1)= -505.3034 to E(v= 1, J= 0)= -507.1696
      Moment matrix elements: <X** 0>= 0.9999989173 <X** 1>= 0.1072309514
      <X** 2>= 0.0227481313 <X** 3>= 0.0044941830 <X** 4>= 0.0010331538
FCF= 1.0000D+00 <M>= 9.79218D-01 d(E)= -1.87 A(Einst)= 6.5155D-07 s-1
+++++
For vibrational level v = 1 of Potential-1
      J E J E J E J E
-----
0 -507.170 1 -505.303
+++++
Coupling E(v= 2, J= 0)= -287.8379 to E(v= 0, J= 1)= -809.4112
      Moment matrix elements: <X** 0>= -0.0001445686 <X** 1>= -0.0146868141
      <X** 2>= 0.0021494006 <X** 3>= 0.0006270858 <X** 4>= 0.0002038462
FCF= 2.0900D-08 <M>= 2.85487D-03 d(E)= -521.57 A(Einst)= 3.6268D-04 s-1
+++++
Coupling E(v= 2, J= 0)= -287.8379 to E(v= 1, J= 1)= -505.3034
      Moment matrix elements: <X** 0>= 0.0012364336 <X** 1>= 0.0838602021
      <X** 2>= 0.0215270661 <X** 3>= 0.0065378301 <X** 4>= 0.0019424342
FCF= 1.5288D-06 <M>=-1.49150D-02 d(E)= -217.47 A(Einst)= 7.1749D-04 s-1
+++++
Coupling E(v= 2, J= 1)= -286.2356 to E(v= 0, J= 0)= -811.5192
      Moment matrix elements: <X** 0>= 0.0001452962 <X** 1>= -0.0147714152
      <X** 2>= 0.0021295981 <X** 3>= 0.0006226650 <X** 4>= 0.0002028174
FCF= 2.1111D-08 <M>= 3.16108D-03 d(E)= -525.28 A(Einst)= 1.5140D-04 s-1
+++++
Coupling E(v= 2, J= 1)= -286.2356 to E(v= 0, J= 2)= -805.1964
      Moment matrix elements: <X** 0>= -0.0002889435 <X** 1>= -0.0146592358
      <X** 2>= 0.0021551720 <X** 3>= 0.0006307829 <X** 4>= 0.0002051064
FCF= 8.3488D-08 <M>= 2.70514D-03 d(E)= -518.96 A(Einst)= 2.1384D-04 s-1
+++++
Coupling E(v= 2, J= 1)= -286.2356 to E(v= 1, J= 0)= -507.1696
      Moment matrix elements: <X** 0>= -0.0012355622 <X** 1>= 0.0834456953
      <X** 2>= 0.0214393061 <X** 3>= 0.0065170119 <X** 4>= 0.0019370216
FCF= 1.5266D-06 <M>=-1.73066D-02 d(E)= -220.93 A(Einst)= 3.3767D-04 s-1
+++++
Coupling E(v= 2, J= 1)= -286.2356 to E(v= 1, J= 2)= -501.5723
      Moment matrix elements: <X** 0>= 0.0024778195 <X** 1>= 0.0841137737
      <X** 2>= 0.0216303216 <X** 3>= 0.0065743346 <X** 4>= 0.0019558701
FCF= 6.1396D-06 <M>=-1.37213D-02 d(E)= -215.34 A(Einst)= 3.9306D-04 s-1
+++++
Coupling E(v= 2, J= 1)= -286.2356 to E(v= 2, J= 0)= -287.8379
      Moment matrix elements: <X** 0>= 0.9999970447 <X** 1>= 0.2089561346
      <X** 2>= 0.0661950523 <X** 3>= 0.0214010286 <X** 4>= 0.0074711130
FCF= 9.9999D-01 <M>= 9.60110D-01 d(E)= -1.60 A(Einst)= 3.9644D-07 s-1
+++++
For vibrational level v = 2 of Potential-1
      J E J E J E J E
-----
0 -287.838 1 -286.236
+++++

```



```

Find 3 Potential-1 vibrational levels with J= 0
  v   E(v)       v   E(v)       v   E(v)       v   E(v)
  0 -811.5192    1 -507.1696    2 -287.8379
An n= 6 N-D theory extrapolation from last 2 levels implies vD = 6.812
=====

```

Case 1.d: For the same L.J.(12,6), find levels of potential with a barrier

```

=====
Generate ZMU= 15.17186628000(u) & BZ= 9.000000000D-01((1/cm-1)(1/Ang**2))
        from atomic masses: 30.34373256000 & 30.34373256000(u)
Integrate from RMIN= 0.600 to RMAX= 99.00 with mesh RH= 0.002000(Angst)

```

Potential #1 for L1( 0)-J2( 0)

State has OMEGA= 0 and energy asymptote: Y(lim)= 0.0000(cm-1)

Lennard-Jones(12, 6) potential with De= 1000.000(cm-1) Re = 1.000000(A)

Calculate properties of the single potential described above  
 Potential-1 uses inner boundary condition of zero value at RMIN

Eigenvalue convergence criterion is EPS= 1.0D-06(cm-1)  
 Airy function at 3-rd turning point is quasibound outer boundary condition

State-1 electronic angular momentum OMEGA= 0  
 yields centrifugal potential [J\*(J+1) - 0.00]/r\*\*2

For J= 18, try to find the first 400 vibrational levels of Potential-1

ITER	ETRIAL	J= 18 F(E)	ETRIAL= DF(E)	-4.7291361D+02 D(E)	INNER= 0 M	R(M)	WF(1st) /WF(M)	WF(NEND) /WF(M)	R(NEND)	NBEG	ITP1
1	-4.7291361D+02	7.96D-03	1.18D+02	1.87D+01	256	1.110	7.4D-27	2.9D-11	2.31	1	178
2	-4.5421107D+02	-1.22D-03	1.76D+02	-1.92D+00	260	1.118	1.0D-26	2.7D-11	2.34	1	177
3	-4.5613214D+02	-2.14D-05	1.70D+02	-3.51D-02	260	1.118	9.9D-27	2.5D-11	2.34	1	177
4	-4.5616719D+02	-6.83D-09	1.70D+02	-1.12D-05	260	1.118	9.9D-27	2.5D-11	2.34	1	177
5	-4.5616720D+02	-4.82D-15	1.70D+02	-7.90D-12	260	1.118	9.9D-27	2.5D-11	2.34	1	177

E(v= 0, J= 18)= -456.1672 5 Iterations R(M)= 1.118 WF(NBEG)/WF(M)= 9.9D-27  
 WF(NEND)/WF(M)= 2.5D-11

ITER	ETRIAL	J= 18 F(E)	ETRIAL= DF(E)	-1.9055580D+02 D(E)	INNER= 0 M	R(M)	WF(1st) /WF(M)	WF(NEND) /WF(M)	R(NEND)	NBEG	ITP1
1	-1.9055580D+02	-4.75D-03	2.74D+02	-4.82D+00	333	1.264	-2.4D-26	4.1D-11	2.94	1	164
2	-1.9537280D+02	-3.48D-04	2.20D+02	-4.40D-01	331	1.260	-2.1D-26	3.1D-11	2.94	1	164
3	-1.9581283D+02	-2.19D-06	2.17D+02	-2.80D-03	331	1.260	-2.1D-26	3.0D-11	2.94	1	164
4	-1.9581563D+02	-8.87D-11	2.17D+02	-1.14D-07	331	1.260	-2.1D-26	3.0D-11	2.94	1	164

E(v= 1, J= 18)= -195.8156 4 Iterations R(M)= 1.260 WF(NBEG)/WF(M)=-2.1D-26  
 WF(NEND)/WF(M)= 3.0D-11

ITER	ETRIAL	J= 18 F(E)	ETRIAL= DF(E)	-2.4929311D+01 D(E)	INNER= 0 M	R(M)	WF(1st) /WF(M)	WF(NEND) /WF(M)	R(NEND)	NBEG	ITP1
1	-2.4929311D+01	-1.62D-03	3.08D+02	-1.46D+00	424	1.446	3.0D-26	6.3D-11	4.59	1	158
2	-2.6389747D+01	-5.65D-05	2.74D+02	-5.73D-02	422	1.442	2.8D-26	6.8D-11	4.53	1	158
3	-2.6447084D+01	-7.47D-08	2.73D+02	-7.60D-05	422	1.442	2.8D-26	6.7D-11	4.53	1	158
4	-2.6447160D+01	-1.51D-13	2.73D+02	-1.54D-10	422	1.442	2.8D-26	6.7D-11	4.53	1	158

E(v= 2, J= 18)= -26.4472 4 Iterations R(M)= 1.442 WF(NBEG)/WF(M)= 2.8D-26  
 WF(NEND)/WF(M)= 6.7D-11

ITER	ETRIAL	J= 18 F(E)	ETRIAL= DF(E)	5.5586943D+01 D(E)	INNER= 0 M	R(M)	WF(1st) /WF(M)	WF(NEND) /WF(M)	R(NEND)	NBEG	ITP1
1	5.5586943D+01	3.15D-03	2.82D+02	3.11D+00	575	1.748	-1.8D-26	2.4D-01	2.40	1	156
2	5.8698635D+01	-2.80D-04	3.97D+02	-1.96D-01	595	1.788	-2.3D-26	3.8D-01	2.29	1	156
3	5.8502801D+01	-1.00D-05	3.87D+02	-7.20D-03	594	1.786	-2.3D-26	3.7D-01	2.30	1	156
4	5.8495601D+01	-3.37D-07	3.87D+02	-2.42D-04	594	1.786	-2.3D-26	3.7D-01	2.30	1	156
5	5.8495358D+01	-1.12D-08	3.87D+02	-8.06D-06	594	1.786	-2.3D-26	3.7D-01	2.30	1	156
6	5.8495350D+01	-3.74D-10	3.87D+02	-2.68D-07	594	1.786	-2.3D-26	3.7D-01	2.30	1	156

E(v= 3, J= 18)= 58.4954 6 Iterations R(M)= 1.786 WF(NBEG)/WF(M)=-2.3D-26  
 WF(NEND)/WF(M)= 3.7D-01  
 Lifetime= 5.035D-12(s) Width= 1.054D+00 dG/dv= 41.60 V(max)= 64.01

ALF finds all J= 18 vib. levels below vD= 3.328 estimated by N-D theory  
 The highest calculated level is E(v= 3) = 58.495350

ITER	ETRIAL	J= 18 F(E)	ETRIAL= DF(E)	-4.5616720D+02 D(E)	INNER= 0 M	R(M)	WF(1st) /WF(M)	WF(NEND) /WF(M)	R(NEND)	NBEG	ITP1
1	-4.5616720D+02	-4.82D-15	1.70D+02	-7.90D-12	260	1.118	9.9D-27	2.5D-11	2.34	1	177

E(v= 0, J= 18)= -456.1672 1 Iterations R(M)= 1.118 WF(NBEG)/WF(M)= 9.9D-27  
 WF(NEND)/WF(M)= 2.5D-11

ITER	ETRIAL	J= 18 F(E)	ETRIAL= DF(E)	-1.9581563D+02 D(E)	INNER= 0 M	R(M)	WF(1st) /WF(M)	WF(NEND) /WF(M)	R(NEND)	NBEG	ITP1
1	-1.9581563D+02	-8.87D-11	2.17D+02	-1.14D-07	331	1.260	-2.1D-26	3.0D-11	2.94	1	164

E(v= 1, J= 18)= -195.8156 1 Iterations R(M)= 1.260 WF(NBEG)/WF(M)=-2.1D-26

WF(NEND)/WF(M)= 3.0D-11

Solve for	v=	J=	ETRIAL=	INNER=	WF(1st)	WF(NEND)					
ITER	ETRIAL	F(E)	DF(E)	D(E)	M	R(M)	/WF(M)	/WF(M)	R(NEND)	NBEG	ITP1
1	-2.6447160D+01	-1.51D-13	2.73D+02	-1.54D-10	422	1.442	2.8D-26	6.7D-11	4.53	1	158

E(v= 2,J= 18)= -26.4472 1 Iterations R(M)= 1.442 WF(NBEG)/WF(M)= 2.8D-26  
WF(NEND)/WF(M)= 6.7D-11

Solve for	v=	J=	ETRIAL=	INNER=	WF(1st)	WF(NEND)					
ITER	ETRIAL	F(E)	DF(E)	D(E)	M	R(M)	/WF(M)	/WF(M)	R(NEND)	NBEG	ITP1
1	5.8495350D+01	-3.74D-10	3.87D+02	-2.68D-07	594	1.786	-2.3D-26	3.7D-01	2.30	1	156

E(v= 3,J= 18)= 58.4954 1 Iterations R(M)= 1.786 WF(NBEG)/WF(M)=-2.3D-26  
WF(NEND)/WF(M)= 3.7D-01

Lifetime= 5.035D-12(s) Width= 1.054D+00 dG/dv= 41.60 V(max)= 64.01

Find	v	E(v)	v	E(v)	v	E(v)	v	E(v)
4	0	-456.1672	1	-195.8156	2	-26.4472	3	58.4954

Case 1.e: X-state {109}AgD using EM03(7,7) potential [JCP 123, 204304(2005)]

Generate ZMU= 1.97752904167(u) & BZ= 1.173076604D-01((1/cm-1)(1/Ang\*\*2))  
from atomic masses: 108.90475200000 & 2.01410177780(u)  
Integrate from RMIN= 0.800 to RMAX= 9.00 with mesh RH= 0.001000(Angst)

Potential #1 for Ag(109)- D( 2)

State has OMEGA= 0 and energy asymptote: Y(lim)= 19250.0000(cm-1)

BOB adiabatic potential correction for atom-2 of mass 2.01410177780  
consists of mass factor [1- MASS( 1 H)/MASS( 2 H)] multiplying all of:  
u2INF= 0.000000 times y3= [(r\*\*3 - Re\*\*3)/(r\*\*3 + Re\*\*3)]  
plus [1 - y3] times an order 3 polynomial in  
y3=[(r\*\*3 - Re\*\*3)/(r\*\*3 + Re\*\*3)] with the 4 coefficients:  
0.000000000E+00 11.7500000 17.5600000 -12.7000000

BOB centrifugal correction for atom-2 of mass 2.01410177780  
consists of mass factor [MASS( 1 H)/MASS( 2 H)] multiplying all of:  
q2INF= 0.000000 times y3= [(r\*\*3 - Re\*\*3)/(r\*\*3 + Re\*\*3)]  
plus [1 - y3] times an order 2 polynomial in y3 with the 3 coefficients:  
0.000000000E+00 0.150000000E-03 0.930000000E-03

Potential is an EM0\_3 with De= 19250.0000 Re= 1.617916200  
Exponent factor is order- 7 power series in y=(r\*\*3 - Re\*\*3)/(r\*\*3 + Re\*\*3)  
with 8 coefficients: 1.543580950D+00 3.738600000D-02 1.664240000D-01  
9.803000000D-02 1.708900000D-01 6.020000000D-02 1.400000000D-01  
2.240000000D-01

Calculate properties of the single potential described above  
Potential-1 uses inner boundary condition of zero value at RMIN

Eigenvalue convergence criterion is EPS= 1.0D-06(cm-1)  
Airy function at 3-rd turning point is quasibound outer boundary condition

State-1 electronic angular momentum OMEGA= 0  
yields centrifugal potential [J\*(J+1) - 0.00]/r\*\*2  
For J= 0, try to find the first 25 vibrational levels of Potential-1  
E(v= 0, J= 0)= 621.527 Bv= 3.2205459 -Dv= -8.7936D-05 Hv= 1.3523D-09  
Lv= -3.8730D-14 Mv= 2.8933D-19 Nv= -2.8833D-23 Ov= -1.2833D-27  
E(v= 1, J= 0)= 1837.802 Bv= 3.1483027 -Dv= -8.7227D-05 Hv= 1.2955D-09  
Lv= -3.9617D-14 Mv= 8.9664D-20 Nv= -3.7762D-23 Ov= -1.4750D-27

..... omit results for v= 2-22 .....

E(v= 23, J= 0)= 18975.036 Bv= 1.0378809 -Dv= -2.7179D-04 Hv= -9.5322D-08  
Lv= -7.6990D-11 Mv= -8.2964D-14 Nv= -1.0641D-16 Ov= -1.5297D-19  
E(v= 24, J= 0)= 19156.482 Bv= 0.7945156 -Dv= -4.1659D-04 Hv= -3.2127D-07  
Lv= -5.8062D-10 Mv= -1.4623D-12 Nv= -4.4539D-15 Ov= -1.5359D-17

Find	v	E(v)	v	E(v)	v	E(v)	v	E(v)
25	0	621.5267	7	8412.7004	14	14454.0900	21	18377.7702
	1	1837.8022	8	9386.7439	15	15159.4225	22	18712.4681
	2	3019.7070	9	10325.0747	16	15820.8804	23	18975.0358
	3	4167.2795	10	11227.0834	17	16436.2800	24	19156.4822
	4	5280.4968	11	12092.0119	18	17003.0435		
	5	6359.2672	12	12918.9344	19	17518.1136		
	6	7403.4212	13	13706.7366	20	17977.8324		

An n=99 N-D theory extrapolation from last 2 levels implies vD = 25.437  
with the 1 missing level(s) predicted to be:

v	E(v)	v	E(v)	v	E(v)	v	E(v)
25	19241.7789						

## E. Input file for Case 2

Note that entries after the “%” sign on each line of the input data file are comments identifying the variables, and are ignored by the program.

```

7 14 7 14 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 2.a: MLR4(5,8) potential for {14,14}N2(X) from JCP 125, 164310 (2006)'
0.0002 0.6 99. 1.d-8 % RH RMIN RMAX EPS
-1 00 0 0.d0 % NTP LPPOT IOMEG VLIM
4 4 6 2 9 1 79845.d0 1.097679d0 % IPOTL MPAR NSR NCMM NVARB IBOB DSCM REQ
6 1.160d+05 8 6.38d+05 % MMLR(i) CMM(i) for i=1,2
-2.34414547D+00 -9.724690D-01 -1.561777D+00 -1.1360D+00 -1.3963D+00
-8.19D-01 -4.5D-01 -3.36D+00 2.1D+00 % PARM(i)
14 14 6 6 1 1 6 -1 -1 % MN1R MN2R PAD MAD NU1 NU2 PNA NT1 NT2
0.d0 0.d0 -14.1d0 % U1INF U1(i)
0.d0 0.d0 -14.1d0 % U2INF U2(i)
-20 1 2 0 0 1 -1 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 0 % IV(i) IJ(i)

37 85 37 85 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 2.b: Seto et al. [JCP 113, 3067 (2000)] MLJ3(14,14) potential for Rb2(X)'
0.0010 2.6 99. 1.d-8 % RH RMIN RMAX EPS
-1 00 0 3993.53d0 % NTP LPPOT IOMEG VLIM
4 -1 14 1 18 0 3993.53d0 4.209951249d0 % IPOTL MPAR NSR NCMM NVARB IBOB DSCM REQ
6 2.261d+07 % MMLR(1) CMM(1)
-5.8905396d0 1.2123488d1 1.4375360d1 3.2361700d1 7.3117000d1 -4.0307000d2
1.6591600d3 -9.6104000d3 -7.3975000d4 4.5020000d5 3.1900000d5 -7.0460000d6
1.9519000d7 -2.2980000d7 1.0300000d7 % \phi(i)-{i=0,14} & Asw Rsw Rinn
1.4d0 13.7d0 0.d0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
-999 1 2 0 0 1 -1 0 % IV(i) IJ(i)
0 0

1 1 9 19 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 2.c: Coxon-Hajigeorgiou MLJ1(15,15) pot1 for HF(X) [JPC A110, 6261(2006)]'
0.0010 0.3 10. 1.d-8 % RH RMIN RMAX EPS
-1 00 0 0 0 49361.6d0 % NTP LPPOT IOMEG VLIM
4 0 15 1 19 1 49361.6d0 % IPOTL MPAR NSR NCMM NVARB IBOB DSCM REQ
0.916838964172d0 % MMLR(i) CMM(i) for i=1
6 37425.d+0 % \phi(i)'s Asw Rsw R{inner}
-3.96723772059d0 0.7963341064d0 0.161009057d0 0.55528385d0 0.3846146d0
0.6309591d0 0.050499d0 -3.689526d0 9.18380d0 16.05901d0
-79.79459d0 53.5575d0 165.63607d0 -354.0128d0 262.6046d0 -70.11957d0
3.7d0 3.0d0 0.58d0 % MN1R MN2R PAD MAD NU1 NU2 PNA NT1 NT2
1 19 0 0 9 -1 1 6 3 % U1INF U1(0) U1(1)
49996.03436d0 0.d0 1.1840711d5 -2.870925d5 3.68864d5 -1.20288d5 -0.50663d5
-12.8253d5 55.5406d5 -73.7058d5 31.2377d5 8.0d0 2.5d0 % u{H}
0.d0 0.d0 -0.111718d0 -0.718557d0 0.064555d0 -0.24251d0
-0.080841d0 -0.145375d0 3.0d0 4.5d0 % q{H}
0.d0 0.d0 -4.01777d0 9.33428d0 -5.46394d0 3.0d0 4.5d0 % q{F}
-99 1 2 0 0 1 -1 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 0 % IV(i) IJ(i)

3 7 3 7 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 2.d: Huang/Le Roy [JCP 119, 7398 (2003)] DELR potential for Li2(B)'
0.0025 1.5 70. 1.d-8 % RH RMIN RMAX EPS
-1 00 1 0.d0 14903.983468d0 2984.444d0 % NTP LPPOT IOMEG VLIM
5 3 5 4 12 1 2984.444d0 2.93617142d0 % IPOTL MPAR NSR NCMM NVARB IBOB DSCM REQ
3 1.788d5 6 -6.97586d6 8 -1.378d8 10 -3.445d9 % MMCM(j) CMM(j)
0.970911966d0 0.2075358d0 0.1751542d0 0.188843d0 0.15648d0 0.252d0
-2.185d0 6.91598d0 -9.6903477d0 4.7186d0
0.4647d0 -1.d0 % \phi(i)'s , RHOD & PARM_4_IDF
7 7 3 3 1 1 3 -1 -1 % MN1R MN2R PAD MAD NU1 NU2 PNA NT1 NT2
1.05578d0 0.7478d0 -0.472d0 % U1INF U1(0) U1(1)
1.05578d0 0.7478d0 -0.472d0 % U1INF U1(0) U1(1)
-99 1 2 0 0 1 -1 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 0 % IV(i) IJ(i)

10 20 36 84 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 2.e: Aziz HFD-B(6,8,10,12) for {20}Ne{84}Kr [JCP 91, 6348 (1989)]'
0.010 2.0 99. 1.d-08 % RH RMIN RMAX EPS
-1 00 0 49.75d0 % NTP LPPOT IOMEG VLIM
6 0 0 4 5 0 49.75d0 3.621d0 % IPOTL MPAR NSR NCMM NVARB IBOB DSCM REQ
6 1.16162375d0 8 0.39191564d0 % MMLR(i) CMM(i) for i=1,2
10 0.35284381d0 12 0.29724436d0 % alpha_i (i=1-3) beta2 gamma
1.d0 0.9365d0 2.d0 0.71720676d0 0.d0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
-99 1 2 0 0 1 -1 0 % IV(i) IJ(i)
0 0

20 40 20 40 0 1 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 2.f: "Tiemann-type" potential for Ca2(X) [Phys.Rev. A66, 042503(2002)]'
0.0005 3.0 99. 1.d-08 % RH RMIN RMAX EPS
-1 0 0 1102.096077d0 % NTP LPPOT IOMEG VLIM
7 0 0 2 24 0 1102.096077d0 4.277277d0 % IPOTL MPAR NSR NCMM NVARB IBOB DSCM REQ
6 -1.074d7 8 -2.4505d8 % MMLR(j) CMM(j) j=1,NCMM
0.00042747d0
-0.254083092764773077d01 0.379611002601149221d04 0.382070302022495241d03
-0.274390396954679318d04 -0.322736334190800926d04 0.363113805693018548d03
0.634370542189755270d04 -0.740151835960846893d04 -0.190738913003729067d05
0.542347392433017594d05 0.440392304373011066d05 -0.155387944954526116d06
-0.836628381353236182d05 0.213831067083156871d06 0.155922449222826835d06
-0.156260872999483137d06 -0.146711120959219668d06 0.277542999772230869d06
0.712908015579339117d05 -0.126115550408998979d04
-0.5929d0 3.66d0 10.d0 % b R{inn} R{out}
-99 1 2 0 0 1 -1 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 0 % IV(i) IJ(i)

```

## F. Input file for Case 3

Note that entries after the “%” sign on each line of the input data file are comments identifying the variables, and are ignored by the program.

```

35 79 35 79 0 2 % IAN1 IMN1 IAN2 IMN2 CHARGE NUMPOT
'Case 3: Predict emission for B-X Br2 based on Gerstenkorn (1987) constants'
0.0015 1.85 15.0 1.d-4 % RH RMIN RMAX EPS
123 0 0 19742.072d0 % NTP LPPOT OMEGA VLIM
10 0 4 5 1.8D+05 % NUSE IR2 ILR CN CN
1.D0 1.D0 15902.4802d0 % RFACT EFACT VSHIFT & B-state turning points
2.32483352887883 3834.348717 2.32492725139916 3831.646866
2.32505070817846 3828.090070 2.32520875281731 3823.540508
2.32540641247184 3817.856430 2.32564885907544 3810.893306
..... skip 56 lines listing 112 more turning points .....
6.87678817820732 3817.856430 7.25492791418183 3823.540508
7.69918940093639 3828.090070 8.22965127226713 3831.646866
8.87579778305112 3834.348717

47 0 0 16056.926D0 % NTP2 LPPOT2 OMEGA2 VLIM2
10 0 0 6 0.D0 % NUSE2 IR22 ILR2 CN2 CNN2
1.D0 1.D0 0.d0 % RFACT EFACT VSHIFT2
2.05649830399176 4483.356304 2.06283295581950 4189.629584
2.06951602200031 3893.539190 2.07659093472317 3595.104297
2.08411094486653 3294.343327 2.09214251282019 2991.273984
..... skip 18 lines listing 36 more turning points .....
2.56851940726203 3294.343327 2.58514008764203 3595.104297
2.60140908192717 3893.539190 2.61738272682198 4189.629584
2.63310768612594 4483.356304

-40 1 0 -4 130 1 -1 0 % NLEV1 AUTO1 LCDC LXPCT NJM JDJR IWR LPRWF
0 0 % IV(1) IJ(1)
1 0 1.D0 % MORDR IRFN RREF
-0.219 0.265 % DM(0) DM(1)
15 1 -1 +1 2 % NLEV2 AUTO2 J2DL J2DU J2DD
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 % {IV2(i)}

```

## G. Channel-6 Output file for Case 3

```

Case 3: Predict emission for B-X Br2 based on Gerstenkorn (1987) constants
=====
Generate ZMU= 39.45916855000(u) & BZ= 2.340730602D+00((1/cm-1)(1/Ang**2))
from atomic masses: 78.91833710000 & 78.91833710000(u)
Integrate from RMIN= 1.850 to RMAX= 15.00 with mesh RH= 0.001500(Angst)

Potential #1 for Br( 79)-Br( 79)
=====
State has OMEGA= 0 and energy asymptote: Y(lim)= 19742.0720(cm-1)
Perform 10-point piecewise polynomial interpolation over 123 input points
Beyond read-in points extrapolate to limiting asymptotic behaviour:
Y(r) = Y(lim) - ( 0.1800000D+06)/r** 5
To make input points Y(i) consistent with Y(lim), add Y(shift)= 15902.4802
Scale input points: (distance)* 1.00000000D+00 & (energy)* 1.00000000D+00
to get required internal units [Angstroms & cm-1 for potentials]
-----
R(i) Y(i) R(i) Y(i) R(i) Y(i)
-----
2.32483353 3834.3487 2.42191370 1696.7527 3.22507343 1821.0440
2.32492725 3831.6469 2.43001839 1568.5519 3.26089292 1941.3789
..... skip 37 lines of the turning point listing .....
2.40751394 1941.3789 3.15509529 1568.5519 8.22965127 3831.6469
2.41443578 1821.0440 3.18985642 1696.7527 8.87579778 3834.3487
-----
Extrapolate to X .le. 2.3249 with
Y= 14763.263 +3.552602D+09 * exp(-5.797858D+00*X)
Function for X .GE. 8.230 generated by 4-point inverse-power interpolation
with leading term 1/r**5 relative to dissociation limit YLIM= 19742.072
and (dimensionless) leading coefficient fixed as C5= 180000.00
-----

Get matrix elements between levels of Potential-1 (above) & Potential-2 (below)
-----
For Potential #2:
=====
State has OMEGA= 0 and energy asymptote: Y(lim)= 16056.9260(cm-1)
Perform 10-point piecewise polynomial interpolation over 47 input points
To make input points Y(i) consistent with Y(lim), add Y(shift)= 0.0000
Scale input points: (distance)* 1.00000000D+00 & (energy)* 1.00000000D+00
to get required internal units [Angstroms & cm-1 for potentials]
-----
R(i) Y(i) R(i) Y(i) R(i) Y(i)
-----
2.05649830 4483.3563 2.18626063 646.2909 2.42013527 966.1727
2.06283296 4189.6296 2.19810575 485.5309 2.43259119 1125.2912
2.06951602 3893.5392 2.21248391 324.2269 2.44438269 1283.8592

```

```

2.07659093 3595.1043 2.22316187 227.1840 2.45563605 1441.8751
2.08411094 3294.3433 2.23177061 162.3804 2.46644318 1599.3372
2.09214251 2991.2740 2.24253421 97.4903 2.47687343 1756.2436
2.10077036 2685.9133 2.25851483 32.5137 2.49680783 2068.3827
2.11010533 2378.2776 2.28102606 0.0000 2.51575473 2378.2776
2.12029713 2068.3827 2.30436129 32.5137 2.53392724 2685.9133
2.13155647 1756.2436 2.32199230 97.4903 2.55147764 2991.2740
2.13767849 1599.3372 2.33440900 162.3804 2.56851941 3294.3433
2.14419655 1441.8751 2.34467357 227.1840 2.58514009 3595.1043
2.15117974 1283.8592 2.35784040 324.2269 2.60140908 3893.5392
2.15871971 1125.2912 2.37638051 485.5309 2.61738273 4189.6296
2.16694249 966.1727 2.39240504 646.2909 2.63310769 4483.3563
2.17602973 806.5054 2.40683303 806.5054
-----
Extrapolate to X .le. 2.0628 with
Y= -2292.367 +1.200354D+10 * exp(-6.996051D+00*X)
Function for X .GE. 2.6174 generated as
Y= 16056.9260 - ( 1.657906D+06 ) * r** 22.046446 * exp[-( 9.991684*r)]
-----
Potential-1 uses inner boundary condition of zero value at RMIN
Eigenvalue convergence criterion is EPS= 1.0D-04(cm-1)
Airy function at 3-rd turning point is quasibound outer boundary condition
State-1 electronic angular momentum OMEGA= 0
yields centrifugal potential [J*(J+1) - 0.00]/r**2
For J= 0, try to find the first 41 vibrational levels of Potential-1
and automatically increment J in steps of 1 to a maximum value of 130
Matrix element arguments are powers of the distance r (in Angstroms)
Coefficients of expansion for radial matrix element/expectation value argument:
-2.190000D-01 2.650000D-01
Potential-2 uses inner boundary condition of zero value at RMIN
Using the rotational selection rule: delta(J)= -1 to 1 with increment 2
calculate matrix elements for coupling to the 15 vibrational levels of
Potential-2: v = 0 1 2 3 4 5 6 7 8 9 10 11 12 13
14
State-2 electronic angular momentum OMEGA= 0
yields centrifugal potential [J*(J+1) - 0.00]/r**2
-----
For vibrational level v = 0 of Potential-1
J E J E J E J E J E
0 15985.810 27 16030.648 54 16161.756 81 16378.536 108 16679.981
1 15985.929 28 16033.967 55 16168.262 82 16388.199 109 16692.754
2 15986.166 29 16037.406 56 16174.885 83 16397.977 110 16705.642
3 15986.522 30 16040.962 57 16181.627 84 16407.872 111 16718.643
4 15986.997 31 16044.637 58 16188.485 85 16417.882 112 16731.758
5 15987.590 32 16048.430 59 16195.461 86 16428.009 113 16744.987
6 15988.302 33 16052.342 60 16202.554 87 16438.251 114 16758.330
7 15989.132 34 16056.372 61 16209.765 88 16448.609 115 16771.787
8 15990.082 35 16060.520 62 16217.093 89 16459.083 116 16785.357
9 15991.150 36 16064.786 63 16224.538 90 16469.673 117 16799.041
10 15992.336 37 16069.170 64 16232.100 91 16480.378 118 16812.838
11 15993.641 38 16073.673 65 16239.780 92 16491.198 119 16826.749
12 15995.065 39 16078.293 66 16247.576 93 16502.135 120 16840.772
13 15996.607 40 16083.032 67 16255.489 94 16513.186 121 16854.909
14 15998.268 41 16087.889 68 16263.520 95 16524.353 122 16869.159
15 16000.048 42 16092.864 69 16271.667 96 16535.635 123 16883.522
16 16001.946 43 16097.957 70 16279.932 97 16547.032 124 16897.998
17 16003.963 44 16103.167 71 16288.313 98 16558.545 125 16912.586
18 16006.098 45 16108.496 72 16296.810 99 16570.172 126 16927.287
19 16008.352 46 16113.943 73 16305.425 100 16581.914 127 16942.101
20 16010.724 47 16119.507 74 16314.156 101 16593.771 128 16957.027
21 16013.215 48 16125.189 75 16323.004 102 16605.743 129 16972.065
22 16015.824 49 16130.990 76 16331.968 103 16617.830 130 16987.216
23 16018.552 50 16136.907 77 16341.049 104 16630.031
24 16021.398 51 16142.943 78 16350.246 105 16642.347
25 16024.363 52 16149.096 79 16359.560 106 16654.777
26 16027.446 53 16155.367 80 16368.990 107 16667.322
+++++
..... skip analogous eigenvalue summaries for vibrational levels v'= 1-33 .....
.....
*** For J=100 E= 19743.24 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J=100 E= 19743.22 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J=100 E= 19743.22 R(3-rd) beyond range so tunneling calculation uses
pure centrifugal potential with J(app)= 99.17 for R > R(max)= 15.00
*** For J=101 E= 19748.69 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J=101 E= 19748.67 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J=101 E= 19748.67 R(3-rd) beyond range so tunneling calculation uses
pure centrifugal potential with J(app)= 100.17 for R > R(max)= 15.00
*** For J=102 E= 19754.16 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J=102 E= 19754.15 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J=102 E= 19754.15 R(3-rd) beyond range so tunneling calculation uses
pure centrifugal potential with J(app)= 101.18 for R > R(max)= 15.00
*** For J=103 E= 19759.66 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J=103 E= 19759.65 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J=103 E= 19759.65 R(3-rd) beyond range so tunneling calculation uses
pure centrifugal potential with J(app)= 102.19 for R > R(max)= 15.00
For J=127 ETRY= 19891.7867 > VMAX= 19891.7746 find onee turn point: R= 2.37
*** SCHRQ has a convergence problem, so for IT= 7 cut DE= -3.22D-03 in HALF

```

```

*** SCHRQ has a convergence problem, so for IT= 8 cut DE= 1.80D-03 in HALF
*** SCHRQ has a convergence problem, so for IT=10 cut DE= -3.51D-03 in HALF
*** SCHRQ has a convergence problem, so for IT=12 cut DE= -3.40D-03 in HALF
*** SCHRQ has a convergence problem, so for IT=14 cut DE= -3.39D-03 in HALF
*** CAUTION for v= 34 J=127 SCHRQ doesn't converge by ITER=15 DE= 1.69D-03
*** SCHRQ has a convergence problem, so for IT= 6 cut DE= -3.22D-03 in HALF
*** SCHRQ has a convergence problem, so for IT= 7 cut DE= 1.80D-03 in HALF
*** SCHRQ has a convergence problem, so for IT= 9 cut DE= -3.51D-03 in HALF
*** SCHRQ has a convergence problem, so for IT=11 cut DE= -3.40D-03 in HALF
*** SCHRQ has a convergence problem, so for IT=13 cut DE= -3.39D-03 in HALF
*** SCHRQ has a convergence problem, so for IT=15 cut DE= -3.39D-03 in HALF
*** CAUTION for v= -1 J=127 SCHRQ doesn't converge by ITER=15 DE=-1.70D-03
    
```

For vibrational level v = 34		of Potential-1							
J	E	J	E	J	E	J	E	J	E
0	19443.803	26	19466.021	52	19529.969	78	19632.179	104	19765.173
1	19443.867	27	19467.723	53	19533.224	79	19636.793	105	19770.719
2	19443.994	28	19469.486	54	19536.537	80	19641.452	106	19776.284
3	19444.185	29	19471.312	55	19539.905	81	19646.158	107	19781.868
4	19444.439	30	19473.198	56	19543.330	82	19650.908	108	19787.469
5	19444.756	31	19475.147	57	19546.811	83	19655.702	109	19793.083
6	19445.138	32	19477.157	58	19550.347	84	19660.539	110	19798.709
7	19445.582	33	19479.228	59	19553.939	85	19665.420	111	19804.345
8	19446.090	34	19481.360	60	19557.585	86	19670.343	112	19809.988
9	19446.662	35	19483.553	61	19561.286	87	19675.307	113	19815.634
10	19447.297	36	19485.806	62	19565.042	88	19680.311	114	19821.282
11	19447.995	37	19488.120	63	19568.851	89	19685.356	115	19826.927
12	19448.756	38	19490.495	64	19572.714	90	19690.440	116	19832.565
13	19449.581	39	19492.930	65	19576.629	91	19695.561	117	19838.191
14	19450.468	40	19495.425	66	19580.598	92	19700.721	118	19843.802
15	19451.419	41	19497.979	67	19584.619	93	19705.916	119	19849.390
16	19452.433	42	19500.594	68	19588.692	94	19711.148	120	19854.948
17	19453.510	43	19503.267	69	19592.816	95	19716.413	121	19860.467
18	19454.650	44	19506.000	70	19596.992	96	19721.713	122	19865.936
19	19455.852	45	19508.792	71	19601.218	97	19727.044	123	19871.337
20	19457.117	46	19511.643	72	19605.494	98	19732.407	124	19876.643
21	19458.445	47	19514.552	73	19609.820	99	19737.800	125	19881.820
22	19459.836	48	19517.520	74	19614.196	100	19743.222	126	19886.844
23	19461.289	49	19520.546	75	19618.620	101	19748.672		
24	19462.804	50	19523.629	76	19623.092	102	19754.148		
25	19464.381	51	19526.770	77	19627.612	103	19759.649		

..... skip analogous eigenvalue summaries for vibrational levels v'= 35-39 .....

```

*** For J= 74 E= 19742.97 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J= 74 E= 19742.96 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J= 74 E= 19742.96 R(3-rd) beyond range so tunneling calculation uses
    pure centrifugal potential with J(app)= 72.87 for R > R(max)= 15.00
*** For J= 75 E= 19746.03 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J= 75 E= 19746.02 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J= 75 E= 19746.02 R(3-rd) beyond range so tunneling calculation uses
    pure centrifugal potential with J(app)= 73.89 for R > R(max)= 15.00
*** For J= 76 E= 19749.10 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J= 76 E= 19749.09 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J= 76 E= 19749.09 R(3-rd) beyond range so tunneling calculation uses
    pure centrifugal potential with J(app)= 74.90 for R > R(max)= 15.00
*** For J= 77 E= 19752.19 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J= 77 E= 19752.18 R(3-rd) > RMAX & E < V(N) so try WKB B.C. @ RMAX
*** For J= 77 E= 19752.18 R(3-rd) beyond range so tunneling calculation uses
    pure centrifugal potential with J(app)= 75.92 for R > R(max)= 15.00
For J= 93 ETRY= 19800.6818 > VMAX= 19800.4705 find onee turn point: R= 2.35
For J= 93 ETRY= 19800.6892 > VMAX= 19800.4705 find onee turn point: R= 2.35
For J= 93 ETRY= 19800.6892 > VMAX= 19800.4705 find onee turn point: R= 2.35
For J= 93 ETRY= 19800.6892 > VMAX= 19800.4705 find onee turn point: R= 2.35
    
```

For vibrational level v = 40		of Potential-1							
J	E	J	E	J	E	J	E	J	E
0	19616.873	19	19626.163	38	19652.625	57	19694.708	76	19749.092
1	19616.922	20	19627.136	39	19654.470	58	19697.304	77	19752.181
2	19617.020	21	19628.157	40	19656.358	59	19699.935	78	19755.283
3	19617.168	22	19629.225	41	19658.289	60	19702.600	79	19758.395
4	19617.364	23	19630.341	42	19660.263	61	19705.297	80	19761.516
5	19617.609	24	19631.503	43	19662.279	62	19708.027	81	19764.642
6	19617.904	25	19632.713	44	19664.337	63	19710.788	82	19767.771
7	19618.247	26	19633.970	45	19666.437	64	19713.579	83	19770.899
8	19618.639	27	19635.273	46	19668.577	65	19716.401	84	19774.022
9	19619.081	28	19636.623	47	19670.759	66	19719.250	85	19777.138
10	19619.571	29	19638.019	48	19672.980	67	19722.128	86	19780.239
11	19620.109	30	19639.461	49	19675.242	68	19725.033	87	19783.320
12	19620.697	31	19640.949	50	19677.543	69	19727.963	88	19786.373
13	19621.333	32	19642.482	51	19679.882	70	19730.918	89	19789.385
14	19622.017	33	19644.061	52	19682.260	71	19733.896	90	19792.337
15	19622.750	34	19645.685	53	19684.676	72	19736.896	91	19795.203
16	19623.531	35	19647.353	54	19687.130	73	19739.918	92	19797.968
17	19624.360	36	19649.067	55	19689.620	74	19742.958		
18	19625.238	37	19650.824	56	19692.146	75	19746.017		

```

Find 41 Potential-1 vibrational levels with J= 0
v      E(v)      v      E(v)      v      E(v)      v      E(v)
-----
0 15985.8099  11 17599.2328  22 18743.5516  33 19404.8742
    
```

1	16150.1164	12	17723.5256	23	18822.9413	34	19443.8033
2	16311.0671	13	17843.8623	24	18898.2751	35	19479.6567
3	16468.6043	14	17960.1999	25	18969.6005	36	19512.5503
4	16622.6712	15	18072.4993	26	19036.9760	37	19542.6009
5	16773.2113	16	18180.7267	27	19100.4706	38	19569.9259
6	16920.1678	17	18284.8542	28	19160.1627	39	19594.6434
7	17063.4841	18	18384.8610	29	19216.1394	40	19616.8729
8	17203.1035	19	18480.7338	30	19268.4954		
9	17338.9707	20	18572.4682	31	19317.3318		
10	17471.0312	21	18660.0691	32	19362.7547		

An n= 5 N-D theory extrapolation from last 2 levels implies vD = 59.899

## G. Channel-3 Output file for Case 3

Case 3: Predict emission for B-X Br2 based on Gerstenkorn (1987) constants

Note that (v',J') & (v'',J'') strictly label the upper and lower levels, resp.,  
and E(lower)=E"  
but E(2)-E(1) is: (energy of State-2 level) - (energy of State-1 level)

dJ(J'')	Band		E(lower)	E(2)-E(1)	A(Einstein)	F-C Factor	<v'j' M v''j">
	v'	v''					
P( 1)	0	- 0	162.54	-15823.27	7.51655D-05	3.26301D-10	7.77793D-06
P( 1)	0	- 1	485.69	-15500.12	1.98479D-03	9.03473D-09	4.12243D-05
P( 1)	0	- 2	806.67	-15179.14	2.56290D-02	1.22429D-07	1.52860D-04
P( 1)	0	- 3	1125.45	-14860.36	2.15688D-01	1.08216D-06	4.57791D-04
P( 1)	0	- 4	1442.03	-14543.78	1.33029D+00	7.01608D-06	1.17424D-03
P( 1)	0	- 5	1756.40	-14229.41	6.41088D+00	3.55737D-05	2.66365D-03
P( 1)	0	- 6	2068.54	-13917.27	2.51331D+01	1.46863D-04	5.45244D-03
P( 1)	0	- 7	2378.43	-13607.38	8.24025D+01	5.07536D-04	1.02119D-02
P( 1)	0	- 8	2686.06	-13299.75	2.30522D+02	1.49801D-03	1.76763D-02
P( 1)	0	- 9	2991.42	-12994.39	5.58667D+02	3.83405D-03	2.84932D-02
P( 1)	0	- 10	3294.49	-12691.32	1.18684D+03	8.61073D-03	4.30264D-02
P( 1)	0	- 11	3595.25	-12390.56	2.23108D+03	1.71302D-02	6.11533D-02
P( 1)	0	- 12	3893.69	-12092.12	3.73965D+03	3.04191D-02	8.21222D-02
P( 1)	0	- 13	4189.78	-11796.03	5.62413D+03	4.85201D-02	1.04526D-01
P( 1)	0	- 14	4483.52	-11502.29	7.62833D+03	6.98793D-02	1.26426D-01
R( 0)	0	- 0	162.38	-15823.55	2.50467D-05	3.26174D-10	7.77641D-06
P( 2)	0	- 0	162.87	-15823.06	5.01114D-05	3.26320D-10	7.77817D-06
R( 0)	0	- 1	485.53	-15500.40	6.61386D-04	9.03137D-09	4.12166D-05
P( 2)	0	- 1	486.02	-15499.91	1.32322D-03	9.03524D-09	4.12255D-05
R( 0)	0	- 2	806.50	-15179.43	8.54047D-03	1.22386D-07	1.52833D-04
P( 2)	0	- 2	806.99	-15178.94	1.70863D-02	1.22436D-07	1.52864D-04
R( 0)	0	- 3	1125.29	-14860.64	7.18760D-02	1.08180D-06	4.57715D-04
P( 2)	0	- 3	1125.77	-14860.15	1.43794D-01	1.08221D-06	4.57804D-04
R( 0)	0	- 4	1441.87	-14544.06	4.43318D-01	7.01388D-06	1.17406D-03
P( 2)	0	- 4	1442.36	-14543.57	8.86871D-01	7.01642D-06	1.17427D-03
R( 0)	0	- 5	1756.24	-14229.69	2.13646D+00	3.55633D-05	2.66326D-03
P( 2)	0	- 5	1756.72	-14229.21	4.27395D+00	3.55753D-05	2.66372D-03
R( 0)	0	- 6	2068.38	-13917.55	8.37592D+00	1.46823D-04	5.45170D-03
P( 2)	0	- 6	2068.86	-13917.07	1.67555D+01	1.46870D-04	5.45257D-03
R( 0)	0	- 7	2378.27	-13607.66	2.74622D+01	5.07408D-04	1.02106D-02
..... Omit 156070 intermediate lines from this listing .....							
P( 92)	40	- 12	4558.90	-15236.31	6.64841D+01	6.27579D-04	1.09187D-02
R( 90)	40	- 13	4823.74	-14971.46	6.84556D+00	6.84007D-05	3.61670D-03
P( 92)	40	- 13	4852.01	-14943.19	6.11369D+00	6.07875D-05	3.40892D-03
R( 90)	40	- 14	5114.60	-14680.60	3.70554D+01	3.84150D-04	-8.66590D-03
P( 92)	40	- 14	5142.74	-14652.46	3.84472D+01	3.96498D-04	-8.80435D-03
R( 91)	40	- 0	846.98	-18950.99	8.39767D+01	4.87753D-04	8.89452D-03
P( 93)	40	- 0	877.16	-18920.80	8.52871D+01	4.92358D-04	8.93667D-03
R( 91)	40	- 1	1167.42	-18630.54	2.10874D+02	1.27296D-03	1.44598D-02
P( 93)	40	- 1	1197.49	-18600.48	2.12678D+02	1.27614D-03	1.44783D-02
R( 91)	40	- 2	1485.68	-18312.29	1.08096D+02	6.78836D-04	1.06238D-02
P( 93)	40	- 2	1515.63	-18282.34	1.07380D+02	6.70336D-04	1.05574D-02
R( 91)	40	- 3	1801.73	-17996.24	2.39523D+00	1.54965D-05	-1.62327D-03
P( 93)	40	- 3	1831.56	-17966.41	2.76649D+00	1.78031D-05	-1.73947D-03
R( 91)	40	- 4	2115.57	-17682.40	1.03528D+02	7.02541D-04	-1.09574D-02
P( 93)	40	- 4	2145.27	-17652.70	1.04870D+02	7.07512D-04	-1.09965D-02
R( 91)	40	- 5	2427.17	-17370.80	4.55941D+01	3.22713D-04	-7.46819D-03
P( 93)	40	- 5	2456.75	-17341.21	4.45962D+01	3.13842D-04	-7.36500D-03
R( 91)	40	- 6	2736.53	-17061.44	1.63941D+01	1.20264D-04	4.60057D-03
P( 93)	40	- 6	2765.99	-17031.98	1.74563D+01	1.27340D-04	4.73392D-03
R( 91)	40	- 7	3043.63	-16754.34	8.11570D+01	6.22525D-04	1.05187D-02
P( 93)	40	- 7	3072.96	-16725.00	8.14359D+01	6.21159D-04	1.05076D-02
R( 91)	40	- 8	3348.45	-16449.52	6.05804D+00	4.87387D-05	2.95411D-03
P( 93)	40	- 8	3377.66	-16420.31	5.45703D+00	4.36755D-05	2.79608D-03
R( 91)	40	- 9	3650.98	-16146.99	4.46439D+01	3.71936D-04	-8.24583D-03
P( 93)	40	- 9	3680.06	-16117.91	4.60071D+01	3.81203D-04	-8.34820D-03
R( 91)	40	- 10	3951.19	-15846.78	4.35551D+01	3.79843D-04	-8.37720D-03
P( 93)	40	- 10	3980.15	-15817.82	4.27258D+01	3.70610D-04	-8.27501D-03
R( 91)	40	- 11	4249.08	-15548.89	4.25959D+00	3.84497D-05	2.69541D-03
P( 93)	40	- 11	4277.91	-15520.06	4.87948D+00	4.38321D-05	2.87733D-03
R( 91)	40	- 12	4544.62	-15253.35	5.56357D+01	5.28869D-04	1.00258D-02
P( 93)	40	- 12	4573.32	-15224.64	5.60365D+01	5.29892D-04	1.00359D-02
R( 91)	40	- 13	4837.80	-14960.17	6.82654D+00	6.82839D-05	3.61565D-03
P( 93)	40	- 13	4866.37	-14931.60	6.15591D+00	6.12800D-05	3.42476D-03
R( 91)	40	- 14	5128.60	-14669.37	2.97279D+01	3.08674D-04	-7.77062D-03
P( 93)	40	- 14	5157.04	-14640.93	3.09159D+01	3.19393D-04	-7.90463D-03