



LEVEL: A computer program for solving the radial Schrödinger equation for bound and quasibound levels



Robert J. Le Roy

Department of Chemistry, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1

ARTICLE INFO

Article history:

Received 15 February 2016

Received in revised form

7 May 2016

Accepted 26 May 2016

Available online 1 June 2016

Keywords:

Radial Schroedinger solver

Diatom vib-rotational energies

Diatom Franck-Condon factors

Quasibound levels

Quasibound level widths

Centrifugal distortion constants

Diatom matrix elements

Diatom Einstein coefficients

ABSTRACT

This paper describes program **LEVEL**, which can solve the radial or one-dimensional Schrödinger equation and automatically locate either all of, or a selected number of, the bound and/or quasibound levels of any smooth single- or double-minimum potential, and calculate inertial rotation and centrifugal distortion constants and various expectation values for those levels. It can also calculate Franck–Condon factors and other off-diagonal matrix elements, either between levels of a single potential or between levels of two different potentials. The potential energy function may be defined by any one of a number of analytic functions, or by a set of input potential function values which the code will interpolate over and extrapolate beyond to span the desired range.

© 2016 Elsevier Ltd. All rights reserved.

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 computer program **LEVEL** described here is a robust and flexible tool for performing such calculations. Although the original version of this program was based upon the famous Franck–Condon intensity program of Zare [1–3], the present version has been considerably modified over several decades, and now incorporates several unique features. In particular, it can:

- (i) automatically locate and calculate the widths of quasibound (orbiting resonance or tunneling predissociation) levels;
- (ii) calculate diatomic molecule inertial rotation and centrifugal distortion constants for levels of a given potential;
- (iii) readily locate levels with dominant wave function amplitude over either well of an asymmetric double minimum potential;
- (iv) 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 (*via* the input data file) rotational selection rules between the levels of a single potential, or between two different potentials. Although the present version only calculates Einstein A coefficients using Hönl–London factors for the case of *singlet–singlet* electronic transitions, it may readily be generalized to treat other cases.

E-mail address: leroy@UWaterloo.ca

<http://dx.doi.org/10.1016/j.jqsrt.2016.05.028>

0022-4073/© 2016 Elsevier Ltd. All rights reserved.

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. Program operation is controlled by the contents of a data file that 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 Appendices A and B in a Supplementary Materials PDF document accompanying this paper. Appendix C there then presents and describes sample data files and (truncated) output files illustrating many of the capabilities of the code, while Appendix D presents an outline the structure of the program and the roles of its various subroutines. The version of the code described herein includes corrections and enhancements incorporated up to April 2016. Individuals currently utilizing older versions [4] will likely find it desirable to obtain the current version since it contains a number of corrections and additional functionality. The author would always appreciate having users inform him of any apparent errors or instabilities in the code, or of additional features that might be 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 determining the discrete eigenvalues and the associated eigenfunctions of the radial or (effective) one-dimensional Schrödinger equation

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

in which μ is the “effective” or reduced mass of the system, J the rotational quantum number, r the internuclear distance, 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 $\Omega \equiv \text{OMEGA}$ is the projection of the electronic angular momentum onto the internuclear axis. However, for the special case of a diatomic molecule rotating in two dimensions, a case invoked by setting the input parameter $\text{OMEGA} > 99$ (see the discussion of data input statement #6 in Appendix B this term becomes $[J^2 - 1/4] \hbar^2 / 2\mu r^2$. Moreover, in order to allow for the special properties of small molecular ions, the program normally defines the reduced mass appearing in Eq. (1) as Watson's “charge-modified reduced mass” [5],

$$\mu = \mu_W \equiv \frac{M_A M_B}{M_A + M_B - m_e Q}, \quad (2)$$

in which M_A and M_B are the atomic masses of the two atoms, m_e is the electron mass, and $Q \equiv \text{CHARGE}$ (see input READ statement #1 in Appendix B) is the (positive or

negative) integer net charge on the molecule. Alternately, it may use the usual two-body expression for the reduced mass, with the missing/added electron mass subtracted from/added to the mass of one atom or the other.

The core of the calculation is the solution of Eq. (1) to determine the eigenvalues E_{vJ} and eigenfunctions $\Psi_{vJ}(r)$ of the potential $V_J(r)$. This is done in subroutine SCHRQ, which is based on the famous Cooley–Cashion–Zare routine SCHR [1–3,6,7], but incorporates the ability to locate and calculate the widths of “quasibound” or tunneling-predissociation levels [8–11], metastable levels that lie above the dissociation limit, but have their dissociation 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 R_H (see READ #5 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 typically obtained using an R_H value that yields a minimum of 30–50 mesh points between adjacent wavefunction nodes in the classically allowed region. An appropriate mesh size may be estimated using the “particle-in-a-box”-based expression

$$R_H = \pi / \left(\text{NPN} \times [\mu \times \max\{E - V(r)\} / C_u]^{1/2} \right). \quad (3)$$

in which NPN is the selected minimum number of mesh points per wavefunction node (say 50), $\max\{E - V(r)\}$ is the maximum of the local kinetic energy (in cm^{-1}) for the levels under consideration (in general it is \lesssim the potential well depth), and $C_u = 16.857629206$ is the “inertial constant”, the value of $\hbar^2/2$ in units $[\text{amu cm}^{-1} \text{Å}^2]$ (see Section 3). A value of NPN that is too small will yield results that will not have the desired accuracy, while too large a value may require excessive computational effort or cause array dimensions to be exceeded. Thus, while Eq. (3) is a useful guide, a careful user should always perform test calculations using different R_H values in order to verify that the calculated results achieve the accuracy desired for the specific application.

The numerical integration of Eq. (1) is performed in the range from R_{MIN} to R_{MAX} (see READ statement #5) using the Numerov algorithm [6,12]. This is a three-point propagation scheme, so 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, $r = R_{\text{MAX}}$, is initialized at an arbitrary value (unity), while its value at the adjacent inner mesh point is defined using the first-order semiclassical wavefunction [13]

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

The Numerov procedure then propagates the wavefunction inward.

At short range, most realistic intermolecular potentials grow very steeply, causing the wavefunctions in the classically forbidden region to die off extremely rapidly with decreasing r . As a result, the wave function at the inner end of the chosen range of integration is normally initialized by placing a node at the lower bound of this range, the read-in distance R_{MIN} . This is effected by setting

$\Psi_{v,J}(r = R_{\text{MIN}}) = 0$, giving $\Psi_{v,J}(r = R_{\text{MIN}} + R_H)$ an arbitrary (non-zero) value, and allowing the Numerov scheme to proceed from there. This is the normal case for a diatomic molecule problem. Note that one should normally set $R_{\text{MIN}} > 0$, 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 precisely symmetric potential whose midpoint is located at R_{MIN} . For asymmetric levels that would have a node at that midpoint, the normal treatment described above will suffice. However, another approach must be implemented for symmetric levels whose wave functions would have zero slope at R_{MIN} . This option is built into subroutine SCHRQ, and is invoked by setting the control parameter `INNOD1` ≤ 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 either to add parameter `INNOD1` to `READ` #24 (see lines #431–465 of the MAIN program), or to recompile the code with the value of `INNOD1`, defined on line #446 of the code, reset to 0.

For special applications, if desired, a hard wall outer boundary condition may be imposed by setting the input integer `IV` (*i*) (see `READ` #25), which would otherwise represent the vibrational quantum number, equal to a large negative number whose absolute value would specify the radial mesh point at which 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*.

The Cooley procedure [6,7] for finding an eigenvalue of Eq. (1) is illustrated in Fig. 1. For any given trial energy the numerical integration proceeds inward from R_{MAX} and outward from R_{MIN} until the two solution segments meet at a chosen matching point r_x . The discontinuity in the slopes of the inward and outward trial functions at r_x is then used to estimate the energy correction required to converge on the eigenvalue closest to the given trial energy [14,6], and this process is repeated until the energy improvement is smaller than the chosen convergence criterion (parameter `EPS` of `Read` #5). 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 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 [6,14] used by SCHRQ tends to become unstable if the matching distance r_x lies in the well in which the wavefunction has very small amplitude. As a result, it is normally necessary to require r_x to lie in the well in which its amplitude is the largest. In the current version of the program, this choice is set by the internal control parameter `INNER` (\equiv `SINNER`), which is given a default

value in Line #408 of the code, but as needed, 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 (normally) performed just as routinely as those for a normal single-well potential.

In general, the outward and inward numerical integrations start at the input distances R_{MIN} and R_{MAX} (see `READ` #5), respectively, which should 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 R_{MIN} or larger R_{MAX} value may be needed in order to ensure that the desired accuracy is achieved. However, should R_{MIN} or R_{MAX} lie sufficiently far into the classically forbidden regions that $|V_J(r) - E|$ becomes extremely large, the integration algorithm itself can become numerically unstable for the given mesh size. For realistic diatomic molecule potential curves, this situation is only likely to occur near R_{MIN} . 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 R_{MIN} will cause these warning messages to disappear and (marginally) reduce the computational effort. For most diatomic molecules, a reasonable value of R_{MIN} 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, smaller values may be necessary.

The program internally defines the upper bound on the range of numerical integration as the smaller of, the read in (`READ` #5) value of R_{MAX} , and the largest distance consistent with the specified mesh and the internally defined (see Section 3) potential energy function and distance array dimension `NDIMR`. As with R_{MIN} , the choice of R_{MAX} is not critical, provided (for truly bound states) that 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} mentioned above is used here. However, due to the anharmonicity of typical molecular potential curves, the requisite values of R_{MAX} 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 estimated for each level using the semiclassical result of Eq. (4), which shows that the wavefunction dies off exponentially in the classically forbidden region with an exponent of

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

in which the turning point $r_2(v,J)$ marks the outer end of the classically accessible region at energy $E_{v,J}$. For each level it considers `S`, SCHRQ first locates $r_2(v,J)$, and then determines a value of $r_{\text{end}}(v,J)$ that 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,

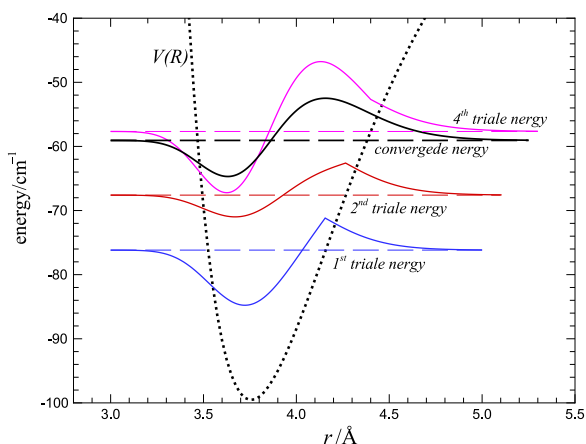


Fig. 1. Illustration of the Löwdin/Cooley eigenvalue convergence procedure [14,6] showing wavefunctions at the first four trial energies that converge on the $v = 1$ eigenvalue of a model potential. This calculation set r_x at the outer turning point for the given trial energy.

the employment of this procedure in the program can reduce the overall computation time by a factor of two or more.

2.2. Locating quasibound levels and determining their widths

Quasibound levels, also known as orbiting resonances, shape resonances, or tunneling-predissociation levels, are metastable eigenstates of Eq. (1) that lie at energies above the potential asymptote $V_J(r \rightarrow \infty)$, but below the maximum of a potential energy barrier that lies outside the main part of the potential well. Although they are part of the continuum of states with $E > V_J(r \rightarrow \infty)$, they are distinct in that their wavefunction amplitude at small distances, inside the potential barrier, is very much larger than those for neighbouring continuum states, so that they manifest themselves as distinct, sharp spectroscopic lines. The nature of the “inside-the-barrier” portion of the wavefunctions for such states is illustrated in Fig. 2, which also demonstrates the immense difference between the tunneling lifetimes of levels lying near the bottom of a barrier, and those lying near the top.

Following Refs. [8,9], program **LEVEL** locates quasibound levels by combining the standard $\Psi_{v,J}(r = R_{\text{MIN}}) = 0$ inner boundary condition with an outer boundary condition defined by the slope of the inward-increasing Airy function at the third (outermost) turning point. This procedure is the most accurate and efficient method for locating quasibound levels that has been proposed to date [8–10,15,16]. It is nearly exact for narrow (long-lived) states, while for the very broadest levels lying marginally below barrier maxima (such as the $v = 2$ level in Fig. 2), the difference with level energies obtained using alternative methods are at most a small fraction (say, $\sim 20\%$) of the width (FWHM) of that level. More accurate predictions for such short-lived states would require a detailed simulation of the actual observable property of interest, since different methods of observing a given quasibound level may yield estimated level energies differing by a fraction of the

level width. For example, if such levels are being observed spectroscopically, the precise positions of the peaks in the bound \rightarrow continuum spectrum should be calculated using a photodissociation simulation code [17,18].

Calculation of the width or tunneling lifetime of a quasibound level by program **LEVEL** is based on Eq. (4.5) of Connor and Smith [10]; a more transparent description of this procedure may be found in Section II.B of Ref. [11]. 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 appearing in Eq. (5) across the classically forbidden 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:

$$\sqrt{2\mu/\hbar^2} \int_{r_1(v, J)}^{r_2(v, J)} [E_{v, J} - V_J(r)]^{-1/2} dr, \quad (6)$$

together with (the energy derivative of) a phase correction factor that takes account of the proximity to the barrier maximum [9,10]. This procedure yields widths that 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 [9,10]. To obtain more accurate widths for those broad levels would again require one to perform a direct simulations of the process by which they are observed. The results in Fig. 2, in which the tunneling lifetimes are seen to range from *ca.* 1 fs to 900 years, illustrate the extreme range of tunneling lifetimes that may arise.

On a practical note, if the outer end of the specified numerical integration range, R_{MAX} , 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 of Eq. (5) over the barrier 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 R_{MAX} . 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 C).

2.3. Calculating diatomic molecule centrifugal distortion constants

The rotational sublevels of a given vibrational level of a molecule have traditionally been represented by the power series [19,20]

$$\begin{aligned} E_{v, J} &= G(v) + B_v[J(J+1) - \Omega^2] - D_v[J(J+1) - \Omega^2]^2 + H_v[J(J+1) - \Omega^2]^3 + \dots \\ &= \sum_{m=0} K_m(v)[J(J+1) - \Omega^2]^m \end{aligned} \quad (7)$$

If desired, program **LEVEL** will compute values of the inertial rotational constant $B_v = (\hbar^2/2\mu)\langle v, J = 0 | 1/r^2 | v, J = 0 \rangle$ and of the first six centrifugal distortion constants associated with this expansion ($-D_v$, H_v , L_v , M_v , N_v and

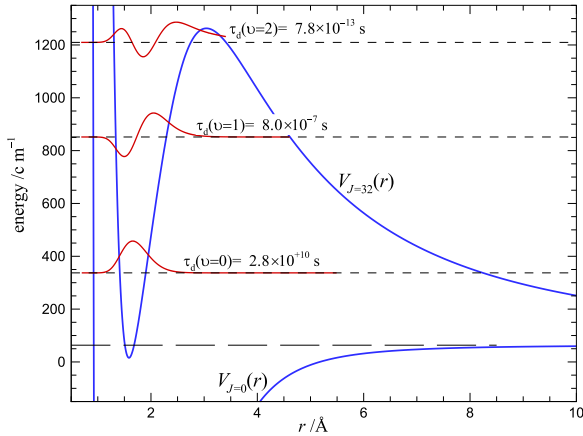


Fig. 2. Quasibound level wavefunctions (red curves) and the associated tunneling lifetimes for the three vibrational levels supported by the $J=32$ centrifugally distorted potential (upper blue curve) for the $(^1\Pi)$ state of CH^+ (adapted from Fig. 2 of Ref. [42]). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

O_v). These constants have their normal significance when generated for levels of a rotationless ($[J(J+1)-\Omega^2]=0$) potential [20], and are simply related to derivatives of the energy with respect to $[J(J+1)-\Omega^2]$ when calculated for vibration-rotational levels with $J > 0$. Calculation of these constants is invoked by setting input parameter `LCDC` > 0 (see `READ` #24), and is performed using a subroutine based on Tellinghuisen's reformulation [21] of the exact quantum mechanical method of Hutson [22], which is extended herein to higher order to allow the calculation of N_v and O_v . To ensure stable, fully converged evaluation of these properties, it is usually necessary to make the eigenvalue convergence parameter `EPS` of `READ` #5 and the value of the radial mesh `RH` distinctly smaller than would be necessary to achieve the normal degree of convergence for the eigenvalue themselves.

2.4. Calculating expectation values, matrix elements, and Einstein A coefficients

If requested, 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 user-specified radial variable $\text{RFN}(r)$:

$$M(r) = \sum_{i=0}^{\text{MORDR}} \text{DM}(i) \times \text{RFN}(r)^i. \quad (8)$$

Parameters `MORDR`, `IRFN` and `DREF` defining the extent of the power series and the nature of the radial variable $\text{RFN}(r)$ are input via `READ` #26, and the power series coefficients `DM(i)` are input via `READ` #27. In this last option, the nature of the radial variable $\text{RFN}(r)$ is specified by the choice of a value for input variable `IRFN` in the range $-4 \leq \text{IRFN} \leq 9$ (see comments regarding `READ` #26 in Appendix B), while setting `IRFN` ≥ 10 causes $M(r)$ to be defined by interpolating over and extrapolating beyond a set of read-in numerical values, and setting `IRFN` ≤ -10

causes $M(r)$ to be defined by a user-defined analytic radial function. For this last case, code for calculating the desired function should be inserted into the program in the manner illustrated by the example in lines #593–614 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` #28–31.

The conventional Franck-Condon factor $\text{FCF} = |\langle \Psi_{v'J'} | \Psi_{vJ} \rangle|^2$ is the square of the matrix element of the zeroth power of $\text{RFN}(r)$, and will be generated whenever any off-diagonal matrix elements are calculated (i.e., whenever input parameter `IXPCT` ≥ 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 A coefficient coupling the two levels in question. For cases in which a pointwise (`IRFN` ≥ 10) or analytic user-defined (`IRFN` ≤ -10) matrix element argument function $M(r)$ is chosen, `MORDR` and `DREF` 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 `DREF` are dummy variables and no `DM(i)` values are read in.

The Einstein 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 [23]

$$A = 3.1361891 \times 10^{-7} \frac{S(J', J'')}{2J' + 1} \nu^3 |\langle \Psi_{v'J'} | M(r) | \Psi_{v''J''} \rangle|^2 \quad (9)$$

Here: A has units s^{-1} , $M(r)$ is the dipole moment (or transition dipole) function in units debye, ν the emission frequency in 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 ± 1 . Note that while versions of this code prior to 7.7 used the $S(J', J'')$ expressions of Herzberg [20], subsequent versions use the corrected Hönl-London factors recommended by Hansson and Watson [24], which for Π - Σ or Σ - Π transitions are a factor of four larger than those listed by Herzberg [20] because of their use of a different definition of the perpendicular transition dipole moment function (see also Bernath [23]). To modify these selection rules and intensity factors, or to extend them to other cases, a user will need to modify lines #60–106 of subroutine `MATXEL`.

2.5. Automatic location of any specified vibrational level (s) of a given potential

For each potential function considered, **LEVEL** begins by examining the input instructions to determine the highest vibrational level of interest, `VMAX`. It then uses subroutine `ALF` to locate all vibrational levels up to that maximum v , and calculates the first seven rotational constants ($K_{(v)}$ for $m=1,7$) for each. If a user wishes to have the code find *all* bound (and quasibound), for potentials with a barrier levels, input parameter `NLEV1` (see `READ` #24) should be set equal to some negative number whose magnitude is much larger than the actual possible number of levels (say, `NLEV1` = -999). The actual level energy

search begins by locating the potential energy function minimum (the lowest minimum for a double-well potential). A simple three-point approximation is used to estimate the harmonic force constant there, and the harmonic oscillator formula is employed to estimate the zero-point energy of that well. This yields a trial energy for subroutine SCHRQ that almost always leads to convergence on the eigenvalue of the $v = 0$ level of that potential. ALF then calls subroutine SCECOR, which evaluates the semiclassical phase integral

$$v(E) + 1/2 = \frac{1}{\pi} \sqrt{\frac{2\mu}{\hbar^2}} \int_{r_1(v)}^{r_2(v)} (E - V(r))^{1/2} dr \quad (10)$$

and its energy derivative

$$\frac{d(v(E) + 1/2)}{dE} = \frac{1}{2\pi} \sqrt{\frac{2\mu}{\hbar^2}} \int_{r_1(v)}^{r_2(v)} (E - V(r))^{-1/2} dr \quad (11)$$

over the well at the energy of the current level, and since the inverse of the latter provides a good estimate of the local vibrational level spacing $\Delta G_{v+1/2}$, its addition to the energy of the current level yields a good trial eigenvalue for the next higher vibrational level. This procedure is normally iterated until all levels have been found.

The above procedure is satisfactory for a single-well potential, or for levels lying above the barrier maximum in a double-well potential, but for levels in a double minimum potential lying below such a barrier maximum and above the minimum of the shallower well, the level search procedure is more complicated. In practise, all but the levels lying closest to the barrier maximum of an asymmetric double-minimum potential are predominantly either “inner-well” or “outer-well” levels, with large wavefunction amplitude over one well, and negligible wavefunction amplitude over the other. Thus, the procedure begins by determining values of the phase integrals of Eqs. (10) and (11) over both wells, at the current energy. Whichever of the two $\{v + 1/2\}$ integral values is closest to having precisely half-integer value identifies the current level as being an “inner-well” or “outer-well” state [25]. The inverse of the second integral for that well then provides an estimate of the distance to the next higher level associated with that well. At the same time, the product of the inverse of the integral of Eqs. (11) for the *other* well with the difference between the value of the its $\{v + 1/2\}$ integral and the next larger half-integer number provides a semiclassical estimate of the distance to the next higher level centred over the second well. The lower of these two energies then defines our estimate for the trial eigenvalue and “well-identity” for the next higher level. That “well-identity” then allows the assignment of the subroutine SCHRQ control parameter *INNER* that assigns the inward/outward wavefunction matching point r_x (see Fig. 1) to the well over which the wavefunction is expected to have maximum amplitude in the subsequent calculation. Of course, at energies very near the maximum of that interwell barrier, the levels often have no unique “well identity”, and the above approach may break down. When this occurs, the code reverts to a “brute force” procedure of systematically changing the trial energy by small

increments until convergence on the desired “next” level is achieved.

Once the above process is complete, the stored values of the vibrational energies and rotational constants (see Section 2.3) allow the code readily to generate an accurate trial energy for any ro-vibrational levels of interest. If this procedure fails (e.g., if centrifugal distortion of a two-well potential causes the level of interest to shift to the other well), a call to subroutine ALF using the appropriate centrifugally distorted potential will normally allow the code to climb up the well to locate and converge on the level of interest.

2.6. Defining the rotationless potential $V(r)$

The potential function package that 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 #6–23; 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* read-in turning points $\{X_I(i), Y_I(i)\}$ input via READ #9, or (by setting *NTP* ≤ 0) by an analytic function. In the former case, interpolation over the input 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 *NUSE*-point piecewise polynomials (typically *NUSE*=8 or 10), while for *NUSE* ≤ 0 the interpolation uses a cubic spline function. If the specified 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 $X_I(2)$, the potential is extrapolated inward with an exponential function fitted to the first three turning points. Similarly, if *RMAX* > $X_I(NTP - 1)$ the potential for $r > X_I(NTP - 1)$ 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 #7 (see Appendix B).

To define the potential by an analytic function, rather than by an array of points, the integer input parameter *NTP* of READ #6 should be set ≤ 0. The program then *skips* READS #7–9 and proceeds instead to #10–23 (see Appendix B), where it reads values of the parameters defining the chosen analytic potential. The present version of the code allows use of the following eight families of analytic potential energy functions.

2.6.1. Lennard–Jones(*m*,*n*) potentials

Lennard–Jones(*m*,*n*) potentials have the form

$$V_{LJ}(r) \equiv \mathfrak{D}_e \left[\left(\frac{n}{m-n} \right) \left(\frac{r_e}{r} \right)^m - \left(\frac{m}{m-n} \right) \left(\frac{r_e}{r} \right)^n \right], \quad (12)$$

in which \mathfrak{D}_e is the well depth and r_e the equilibrium internuclear distance.

2.6.2. “Generalized polynomial expansion function” (GPEF) potentials

The GPEF functional form [26] is

$$V_{\text{GPEF}}(r) \equiv \beta_0 z^2 \left(1 + \sum_{i=1}^{N_\beta} \beta_i z^i \right). \quad (13)$$

These functions are Dunham-like [19] polynomial expansions based on Seto's modification [27] of the dimensionless radial variable

$$z = \frac{r^q - r_e^q}{a_s r^q + b_s r_e^q} \quad (14)$$

that was introduced by Šurkus et al. [26], and include the familiar Dunham [19], Simons–Parr–Finlan [28], and Ogilvie–Tipping [29] expansions as special cases that are invoked by particular choices for its parameters q , a_s , and b_s (see discussion of READ #10 in Appendix B).

2.6.3. “Extended morse oscillator” or EMO potentials

EMO potentials have the functional form [27,30–32]:

$$V_{\text{EMO}}(r) \equiv \mathfrak{D}_e \left[1 - e^{\beta(r)(r-r_e)} \right]^2, \quad (15)$$

in which \mathfrak{D}_e and r_e are the well depth and equilibrium internuclear distance, and

$$\beta(r) = \beta_{\text{EMO}}(r) \equiv \sum_{i=0}^{N_\beta} \beta_i [y_q^{\text{ref}}(r)]^i \quad (16)$$

is a simple polynomial in the Šurkus-type variable [26]:

$$y_q^{\text{ref}}(r) = y_q(r; r_{\text{ref}}) \equiv \frac{r^q - r_{\text{ref}}^q}{r^q + r_{\text{ref}}^q}, \quad (17)$$

in which q is a selected small positive integer and r_{ref} a chosen expansion centre. That expansion centre has often been set equal to the equilibrium distance r_e , but there are sometimes substantial advantages associated with allowing it to have larger values [33,34]. Truncating the exponent expansion at the constant term yields the familiar simple Morse potential [35].

Another special-case Morse-type function allowed by POTGEN is Hua Wei's 4-parameter potential [36]:

$$V(r) = \mathfrak{D}_e \left([1 - e^{-b(r-r_e)}] / [1 - C e^{-b(r-r_e)}] \right)^2. \quad (18)$$

2.6.4. “Morse/long-range” or MLR potentials

MLR potentials [37,33,34] are expressed as

$$V_{\text{MLR}}(r) \equiv \mathfrak{D}_e \left[1 - \frac{u_{\text{LR}}(r)}{u_{\text{LR}}(r_e)} e^{-\beta(r)y_p^{\text{re}}(r)} \right]^2. \quad (19)$$

These functions have the same Morse-like [35] structure as the EMO potentials of Eq. (15), but the radial variable in the exponent is now a dimensionless variable with the structure of $y_q^{\text{ref}}(r)$ of Eq. (17), but with the expansion centre fixed at the equilibrium distance $r_{\text{ref}} = r_e$, and the exponential term has a pre-factor defined by an attractive long-range potential tail function $u_{\text{LR}}(r)$ whose form is

given by theory [38–41] to be

$$u_{\text{LR}}(r) = \sum_{i=1}^{\text{last}} D_{m_i}(\rho r) \frac{C_{m_i}}{r^{m_i}}, \quad (20)$$

in which the $D_{m_i}(r)$ are “damping functions” that approach 1 as $r \rightarrow \infty$, and (normally, see below) approach zero as $r \rightarrow 0$ sufficiently rapidly to suppress the singularities in the inverse-power terms [34], while ρ is a system-dependent range-scaling parameter (see below).

The two defining features of the MLR potential form are: (i) since the pre-factor to the exponential term in Eq. (19) is equal to 1 at the equilibrium distance, r_e , and $y_p^{\text{re}}(r_e) = 0$, then $V_{\text{MLR}}(r_e) = 0$ and (ii) the requirement that

$$\lim_{r \rightarrow \infty} \beta(r) = \beta_\infty = \ln \left\{ \frac{2\mathfrak{D}_e}{u_{\text{LR}}(r_e)} \right\}, \quad (21)$$

which means that at large distances Eq. (19) becomes

$$\begin{aligned} V_{\text{MLR}}(r) &\simeq \mathfrak{D}_e - u_{\text{LR}}(r) + \frac{1}{4\mathfrak{D}_e} [u_{\text{LR}}(r)]^2 \\ &\simeq \mathfrak{D}_e - \sum_{i=1}^{\text{last}} \frac{C_{m_i}}{r^{m_i}} + \frac{1}{4\mathfrak{D}_e} \left(\sum_{i=1}^{\text{last}} \frac{C_{m_i}}{r^{m_i}} \right)^2. \end{aligned} \quad (22)$$

The final form for Eq. (22) shows that although the MLR function does take on the theoretically predicted inverse-power-sum form at long range, unless $m_{\text{last}} < 2m_1$, the limiting long-range behaviour of the quadratic term in Eqs. (19) and (22) will change the long-range behaviour specified by Eq. (20). This problem is most serious for cases in which $m_1 = 3$ [33], but can also cause difficulties for systems with larger m_1 values [42]. However, it is resolved here by the fact that, as necessary, the code internally modifies the input C_{m_i} coefficients, and/or adds additional terms, so as to precisely cancel the effect of those spurious non-physical quadratic terms, as illustrated by the discussion of the coefficients $C_6^{\text{adj}} = C_6 + (C_3)^2 / 4\mathfrak{D}_e$ and $C_9^{\text{adj}} = C_3 C_6^{\text{adj}} / 2\mathfrak{D}_e$ for the $m_1 = 3$ case in Ref. [33].

The exponent coefficient function $\beta(r)$ has normally been written as a constrained polynomial expansion defined in terms of two Šurkus-type variables [26] that both have the form of Eq. (17), but are defined by different integers, p and q [33,34]:

$$\beta(r) = \beta_{\text{PE-MLR}}(r) \equiv y_p^{\text{ref}}(r) \beta_\infty + [1 - y_p^{\text{ref}}(r)] \sum_{i=0}^{N_\beta} \beta_i [y_q^{\text{ref}}(r)]^i. \quad (23)$$

Potentials using this type of exponent coefficient function are called polynomial-exponent MLR (PE-MLR) functions.

As an alternative to Eq. (23), $\beta(r)$ may be represented as a natural cubic spline in the independent variable $y_q^{\text{ref}}(r)$, passing through a specified set of input points, as described in Refs. [43,44],

$$\beta(r) = \beta_{\text{SE-MLR}}(r) \equiv \sum_{k=1}^{N_\beta} S_k(y_q^{\text{ref}}(r)) \beta_k, \quad (24)$$

in which the spline “basis functions” $S_k(y_q^{\text{ref}}(r))$ are completely defined by the chosen mesh of values of $y_q^{\text{ref}}(r_j)$ [45,46]. In this case, the connection to the long-range behaviour of Eq. (22) is provided by the constraint of fixing $\beta_{N_\beta} = \beta_\infty$.

Potentials using this second type of exponent coefficient function are called spline-exponent MLR (SE-MLR) functions.

Program **LEVEL** offers two choices regarding the form of the damping functions $D_m(r)$ in Eq. (20): one is based on a form proposed by Douketis et al. [47],

$$D_m^{\text{DS}(s)}(\rho r) = \left[1 - \exp \left\{ -\frac{b^{\text{ds}}(s) \cdot (\rho r)}{m} - \frac{c^{\text{ds}}(s) \cdot (\rho r)^2}{\sqrt{m}} \right\} \right]^{m+s}, \quad (25)$$

and the other is based on a form proposed by Tang and Toennies [48],

$$D_m^{\text{TT}(s)}(\rho r) = 1 - e^{-b^{\text{tt}}(s) \cdot \rho r} \sum_{k=0}^{m-1+s} \frac{(b^{\text{tt}}(s) \cdot \rho r)^k}{k!}, \quad (26)$$

in both of which ρ is a system-dependent range-scaling parameter [47,34], while $b^{\text{ds}}(s)$, $c^{\text{ds}}(s)$ and $b^{\text{tt}}(s)$ are previously optimized [34] system-independent parameters whose values are built into the code.

Both of these generalized damping function forms [34] have the property that at very small r

$$D_m^s(r)/r^m \simeq r^s \quad (27)$$

for all values of m and s . The original Tang–Toennies model [48] corresponds to the case $s = +1$, and that would present no problem if $u_{\text{LR}}(r)$ were an additive contribution to the potential (as it is for potential function forms of Sections 2.6.5–2.6.8, discussed below). However, it cannot be used in the MLR potential form, as it requires $s \leq 0$ to prevent the repulsive potential wall from turning over and approaching zero as $r \rightarrow 0$ [34,44]. The original Douketis et al. [47] model corresponds to the case $s = 0$, and while that presents no formal problems for either MLR potentials or the functions described below, the results of Ref. [34] suggest that either $s = -1$ or $s = -1/2$ might be a better choice, since they seem to lead to more realistic physical behaviour at very small r [34].

The long-range interaction energy in an MLR potential is not restricted to having the form of Eq. (20), but could take on any form predicted by theory. In particular, in the present version of **LEVEL** it may be represented as one of the roots of a 2×2 or 3×3 diagonalization arising from the type of two-state [33,49] or three-state [50,51] coupling encountered for alkali dimers at the first $nS+nP$ asymptote [44]. A more detailed discussion of various features of the MLR potential function forms may be found in Ref. [44] and in Appendix B.

2.6.5. The “double-exponential long-range” or DELR potential

DELR potentials [11] have the functional form

$$V_{\text{DELR}}(r) \equiv Ae^{-2\beta(r) \cdot (r-r_e)} - Be^{-\beta(r) \cdot (r-r_e)} + u_{\text{LR}}(r), \quad (28)$$

in which the exponent coefficient $\beta(r)$ has exactly the same form as that for the EMO potential (see Eq. (16)), and $u_{\text{LR}}(r)$, the function chosen to represent the long-range region, is assumed to have the form of Eq. (20) or any of the coupled-state functions mentioned at the end of preceding subsection. The pre-exponential factors A and B are defined in terms of the well depth and the properties of

$u_{\text{LR}}(r)$ at r_e , that is,

$$A = \mathfrak{D}_e + V_{\text{LR}}(r_e) + V'_{\text{LR}}(r_e)/\beta_0, \quad (29)$$

$$B = 2\mathfrak{D}_e + 2V_{\text{LR}}(r_e) + V'_{\text{LR}}(r_e)/\beta_0 s, \quad (30)$$

in which $V'_{\text{LR}}(r_e) \equiv dV_{\text{LR}}(r)/dr|_{r=r_e}$.

An illustration of how a DELR function maybe used to represent a potential energy function with a barrier lying outside the main well is provided by Fig. 3 of Ref. [44].

2.6.6. The generalized “HFD” potential function

The present generalization of the “Hartree–Fock/Dispersion” or HFD potential function [52,53] is written as

$$V_{\text{HFD}}(r) \equiv A_{\text{HFD}} \left(\frac{r}{r_e} \right)^\gamma e^{(-\beta_1 r - \beta_2 r^2)} - D_{\text{HFD}}(r) \sum_m \frac{C_m}{r^m}, \quad (31)$$

in which

$$D_{\text{HFD}}(r) = \exp \left\{ -\alpha_1 [\alpha_2/r - 1]^{\alpha_3} \right\} \quad \text{for } r < \alpha_2 \\ = 1 \quad \text{for } r \geq \alpha_2, \quad (32)$$

is a global damping function applied to all inverse-power terms. The input parameters are the long-range coefficients $\{C_m\}$ and damping function parameters $\{\alpha_i\}$ defining the attractive inverse-power-sum term, the power γ and the quadratic exponent coefficient β_2 , while the values of exponential-term parameters A_{HFD} and β_1 are derived internally from the fact that the potential well has a depth of \mathfrak{D}_e at the equilibrium internuclear distance r_e .

2.6.7. The generalized Tang–Toennies (gTT) potential function form

Generalized Tang–Toennies-type potentials consist of the sum of a repulsive exponential term, whose exponent and pre-factor may consist of multiple terms, plus a sum of attractive inverse-power terms that are damped by the same $s = +1$ version of the generalized TT-type damping function of Eq. (26) that is a key feature of the original TT form [48,54]. This structure has been used in a number of published potentials [55–57], and the extended version of this form available in **LEVEL** incorporates most variants found in the recent literature:

$$V_{\text{gTT}}(r) \equiv \left(\beta_5 + \beta_6 r + \frac{\beta_7}{r} + \beta_8 r^2 + \beta_9 r^3 \right) \\ \times e^{-\{\beta_1 r + \beta_2 r^2 + \beta_3/r + \beta_4/r^2\}} - u_{\text{LR}}(r), \quad (33)$$

in which the attractive outer-wall function $u_{\text{LR}}(r)$ has the form of Eq. (20) with the damping functions defined by $s = 1$ Eq. (26). However, while the “basic” Tang–Toennies function [48,54] constrains the argument of the damping function to be identical to the leading term in the exponent of Eq. (33), $\beta_1 r$, in the present implementation these terms are independent; the damping function argument is ρr with the shape parameter of Eq. (26) fixed at $b^{\text{tt}}(s) = 1$. Moreover, while the original Tang–Toennies model only allowed for even powers m_i in Eq. (20), with $m_i \geq 6$, and used the individual damping functions of Eq. (26) with $s = +1$. LEVEL will accept any user-supplied set of powers $\{m_i\}$ and either choice of damping function model. The input parameters for this model are then the set of powers $\{m_i\}$ and coefficients $\{C_{m_i}\}$ for the attractive outer branch of

the potential, and the values of the parameters ρ and $\beta_1 - \beta_9$, while the well depth \mathfrak{D}_e and equilibrium distance r_e are *properties*, rather than defining parameters of the potential.

2.6.8. The “X-expansion” or “Hannover polynomial potential” (HPP) form

The HPP or “X-expansion” form that was introduced by Tiemann and co-workers [58] represents the main part of the potential by a GPEF-like power series

$$V_{\text{HPP}}(r) \equiv \beta_0 + \beta_1 \xi + \beta_2 \xi^2 + \beta_3 \xi^3 + \beta_4 \xi^4 + \dots \quad (34)$$

in which $\xi \equiv \xi(r, b, r_m) = (r - r_m)/(r + br_m)$. The possible presence of a non-zero linear term ($\beta_1 \neq 0$) in Eq. (34) means that when $\beta_1 \neq 0$ the parameter r_m (which is input as $r_e = \text{REQ}$) only approximately corresponds to the equilibrium distance. At a specified small distance, this power series is smoothly joined to an exponential, and beyond a specified large distance, it is attached to an inverse-power sum with the form of Eq. (20).

2.6.9. Other choices

Note that apart from the simple GPEF polynomial potentials of Section 2.6.2, all of the above analytic potentials are defined relative to the absolute energy at the asymptote, which is specified by input parameter VLIM (see READ #6). For the HPP potential form this is effected by internally setting $a_0 = \text{VLIM} - \mathfrak{D}_e$, while the GPEF potentials of Eq. (13) are defined with VLIM set at the potential minimum.

Users may also readily introduce their own analytic potential functional forms, simply by replacing subroutine POTGEN with their own potential routines and making input parameter NTP of READ #6 a negative integer. To retain consistency with the rest of the present code, such a user-prepared POTGEN subroutine should have the argument list:

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

The first argument, parameter LNPT , is an integer which is fixed in program **LEVEL** as $\text{LNPT} = 1$ [59]. The other input quantities are the integer NPP specifying the size of the various radial arrays, the integers specifying the atomic numbers ($\text{IAN1}/\text{IAN2}$) and mass numbers ($\text{IMN1}/\text{IMN2}$) of the isotopologue of interest (required for calculating Born–Oppenheimer breakdown terms, see below), the absolute energy VLIM (in cm^{-1}) at the potential asymptote, the radial distances array $R(i)$ (in Å) at which potential values are to be generated, and the squared inverse distance array $\text{RM2}(i) = 1/R(i)^2$. The subroutine should return the desired NPP -point array of potential function values $\text{VV}(i)$ (in units cm^{-1}), 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 #24), 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 may be used in a near-dissociation theory [60–62] algorithm to estimate the number and energies of missing levels. If a user-specified 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., $\text{NCN} = 99$).

2.7. Born–Oppenheimer breakdown (BOB) 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 input parameter IBOB has a value greater than zero. These (optional) functions are defined as in Ref. [63]. 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. (17), namely,

$$\left(\frac{M_a - M_a^{\text{ref}}}{M_a} \right) \left(y_{p_{\text{ad}}}^{r_e}(r) u_{\infty}^a + \left[1 - y_{p_{\text{ad}}}^{r_e}(r) \right] \sum_{j=0}^{N_{\text{ad}}^a} u_j^a [y_{q_{\text{ad}}}^{r_e}(r)]^j \right), \quad (35)$$

in which M_a is the mass of the specific isotope of atom- a , M_a^{ref} is the mass of the chosen reference isotope of that species [63], $y_{p_{\text{ad}}}^{r_e}(r)$ and $y_{q_{\text{ad}}}^{r_e}(r)$ have the form of Eq. (17), but with $r_{\text{ref}} = r_e$, and the integers p_{ad} and q_{ad} are chosen in the manner discussed in Refs. [63,33,42].

The centrifugal BOB correction is a multiplicative factor $[1 + g_A(r) + g_B(r)]$ that is applied to the centrifugal contribution to the overall potential function $V_j(r)$ of Eq. (1), in which 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. (35):

$$g_a(r) = \frac{M_a^{\text{ref}}}{M_a} \left(y_{q_{\text{na}}}^{\text{eq}}(r) t_{\infty}^a + \left[1 - y_{q_{\text{na}}}^{\text{eq}}(r) \right] \sum_{j=0}^{N_{\text{na}}^a} t_j^a [y_{q_{\text{na}}}^{\text{eq}}(r)]^j \right). \quad (36)$$

Note, however, that this expression has no separate integer p_{na} , since no general theoretical prediction regarding the limiting long-range behaviour of these functions is available. However, the algebraic structure of Eqs. (23) and (35) has been retained in order to allow for the treatment of molecular ions, for which t_{∞}^a would be non-zero [63].

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 cm^{-1} , respectively. One exception to this rule is that the transition dipole function $M(r)$ of Eqs. (8) used for calculating the Einstein coefficients of Eq. (9), as defined by the expansion coefficients $\text{DM}(i)$ of READ #27 (see Appendix B) is assumed to be in debye (where 1 debye = $3.335640952 \times 10^{-30}$ C m = 0.393430295 au). Note, however, that in the $\text{IRFN} \geq 10$ option that defines the matrix element argument by numerically interpolating over a set of read-in points (see READS #28–31), the main (Channel-6) output describing the read-in transition moment function values being interpolated over will (incorrectly) refer to their units as “ cm^{-1} ” rather than debye, since the interpolation is done by the subroutine package that was set up to deal with a pointwise input potential. Note too, that if a 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` #8 and 30) to convert them to the appropriate units.

Values of the physical constants appear in the program in two places. The first is the factor $\mathcal{C}_u \equiv \hbar^2/2 = 16.857629206 \text{ amu} \cdot \text{\AA}^2 \text{ cm}^{-1}$ [64] appearing in the radial Schrödinger equation of Eq. (1) and in Eq. (3), whose units effectively define the units of the input/output variables. The second is in the collections of terms defining the numerical factor in Eq. (9) used in calculating the Einstein coefficient for the rate of spontaneous emission, as described above. Current values of these constants are based upon the 2012 CODATA recommended values of Ref. [64]. In addition, the masses of all stable isotopes of all atoms taken from The AME2012 Atomic Mass Evaluation of Ref. [65] are stored in data subroutine `MASSSES`.

The array dimension limits, which a user might wish to change, are set in `PARAMETER` statements in the main driver routine and in subroutines `GENINT` and `SPLINT`. In the former, `NDIMR` (currently 250 001) 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 safely be 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 that may be saved when applying the `NJM > 0` option to generate automatically many J sublevels for a given v (see `READ` #24). The two other array-size parameters set inside the code are `NTPMX` (currently 2000, set in `PREPOT`), which specifies the maximum number of potential function points (or radial matrix element argument $M(r)$ values, for `IRFN > 10`) that may be read in to be interpolated over, and `MAXSP` (currently $8000 = 4 \times \text{NTPMX}$, set in `SPLINT`), the number of spline coefficients required for interpolation over the read-in function values.

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` #6 & 24) 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>`, in which `<filename>.5` is the name

of the user-created input data file (note that `<filename>` may be any name, and is normally chosen to identify a specific case). For the above shell, the standard output from Channel-6 will be written to file `<filename>.6`, and the Channel-7, Channel-8, Channel-9, and Channel-10 outputs will be written to files `<filename>.7`, ..., `<filename>.10`, respectively.

4. Concluding remarks *mea culpa*

In conclusion, I should bring the reader's attention to a capability that is *not incorporated* into the present version of **LEVEL**. In 2008 and 2011 Meshkov, Stolyarov and I published two papers describing the use of an "adaptive mapping procedure" that replaces numerical integration to solve Eq. (1) on the infinite domain $r \in [0, \infty)$ by integration using a dimensionless radial variable, such as $y_q^{\text{ref}}(r)$ of Eq. (17), on the finite domain $y_q^{\text{ref}} \in [-1, +1]$ [66,67]. Use of this procedure allows one to readily obtain accurate solutions for levels lying extremely close to dissociation, such as sometimes observed for excited states of electronically excited alkali dimers, or to calculate the s -wave scattering length associated with any given potential energy function using a single-pass finite-domain calculation. Unfortunately, the different array structures required by this approach means that it has had to be incorporated into a separate, albeit parallel version of the code, which (up till now, anyway) has always lagged behind the most current "conventional" version of **LEVEL**, and has not yet been posted for public use. However, until a properly documented "adaptive mapping procedure" version **LEVEL** can be posted for public use, I will continue to provide an (incompletely documented) version of such a code to individuals who request it.

Acknowledgements

I am pleased to acknowledge the foresight and stimulation provided by the late Professor George Burns of the University of Toronto who instigated my work on this type of computational research tool back in 1966, and the grace, advice, and financial support provided by the late Professor Richard Bernstein, then of the University of Wisconsin, Madison, who patiently supported my meanderings in this area. 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 [1–3]. I thank J.Y. Seto for assistance in developing the initial version of the level-finder subroutine, `ALF`, and J. Y. Seto, G.T. Kraemer and K. Slaughter for assistance in developing and updating the atomic mass database subroutine `MASSSES`. I am also grateful to many users for comments that identified errors and suggested useful extensions of the code, with particular thanks to R.W. Field, P.F. Bernath, A.J. Ross, A.S. Dickinson, J.O. Hornkohl and G.C. McBane. Finally, I am also grateful to my colleague Professor F.R.M. McCourt for his thorough and insightful comments on this paper. Finally, I am pleased to thank Natural Sciences and Engineering Research Council of

Canada which has generously supported my work in this area for the past four decades.

Appendix A. Supplementary data

Supplementary material associated with this paper consist of: a text file containing the full standalone FORTRAN source code for program **LEVEL**, a text file containing the sample data files that are presented in Appendix C, and a PDF document containing: (i) Appendix A, a structured listing of the READ statements in the program, (ii) Appendix B full definitions of all of the parameters and quantities read in the input data file, and descriptions of the associated program options, (iii) Appendix C, commented listings of illustrative sample input and output files, and (iv) Appendix D, a structured listing of the subroutines comprising the code with brief descriptions of their functions. Note that the equation and reference numbering appearing in this Supplemental Material document refer to those in this journal article, so for completeness we hereby cite nine papers referenced there that are associated with particular applications of this code, but had not been referred to above [68–76].

Anyone who wishes to be registered with the author as a user of this code, eligible to be sent any future bug fixes or updates, should fill in the online form at the [www address](http://www.sciencide2.uwaterloo.ca/~rleroy/LEVEL16) (<http://www.sciencide2.uwaterloo.ca/~rleroy/LEVEL16>)".

Supplementary data associated with this paper can be found in the online version at <http://dx.doi.org/10.1016/j.jqsrt.2016.05.028>.

References

- Zare RN, Cashion JK. 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.
- Zare RN. Programs for Calculating the Relative Intensities in the Vibrational Structure of Electronic Band Systems, University of California Lawrence Radiation Laboratory Report UCRL-10925; 1963.
- Zare RN. *J Chem Phys* 1964;40:1934.
- (a) RJ Le Roy. LEVEL 1.0. University of Wisconsin Theoretical Chemistry Institute Report WIS-TCI-429G; 1971; (b) Le Roy RJ. LEVEL 2.0. University of Waterloo Chemical Physics Research Report CP-58; 1976; (c) Le Roy RJ. LEVEL 3.0. CP-110; 1979; (d) Le Roy RJ. LEVEL 4.0. CP-230; 1983; (e) Le Roy RJ. LEVEL 5.0. CP-330; 1991; (f) Le Roy RJ. LEVEL 5.1. CP-330R; 1992; (g) Le Roy RJ. LEVEL 5.2. CP-330 R²; 1993; (h) Le Roy RJ. LEVEL 5.3. CP-330 R³; 1994; (i) Le Roy RJ. LEVEL 6.0. CP-555; 1995; (j) Le Roy RJ. LEVEL 6.1. CP-555R; 1996; (k) Le Roy RJ. LEVEL 7.0–7.7. CP-642 – 642 R³, CP-655 & CP-661; 2000–2005; (l) Le Roy RJ. LEVEL 8.0. CP-663; 2007–2014; (m) Le Roy RJ. LEVEL 8.2. CP-668; May 2014 [see <http://www.leroy.uwaterloo.ca/programs/>].
- Watson JKG. *J Mol Spectrosc* 1980;80:411.
- Cooley JW. *Math Comput* 1961;15:363.
- Cashion J. *J Chem Phys* 1963;39:1872.
- Le Roy RJ, Bernstein RB. *J Chem Phys* 1971;54:5114.
- Le Roy RJ, Liu W-K. *J Chem Phys* 1978;69:3622.
- Connor JNL, Smith AD. *Mol Phys* 1981;43:397.
- Huang Y, Le Roy RJ. *J Chem Phys* 2003;119:7398 [erratum: Huang Y, Le Roy RJ. *J Chem Phys* 2007;126:169904].
- Smith K. The calculation of atomic collision processes. New York: Wiley-Interscience; 1971 [Chapter 4].
- Child MS. *Semiclassical mechanics with molecular applications*. Oxford: Clarendon Press; 1991.
- Löwdin P-O. An elementary iteration-variation procedure for solving the Schrödinger Equation. Technical note No. 11, Quantum Chemistry Group, Uppsala University, Uppsala, Sweden; 1958.
- Hehenberger M, Froelich P, Brändas E. *J Chem Phys* 1976;65:4559.
- Hehenberger M. *J Chem Phys* 1977;67:1710.
- Le Roy RJ. *Comput Phys Commun* 1989;52:383.
- Le Roy RJ, Kraemer GT. BCONT 2.2: a computer program for calculating bound → continuum transition intensities for diatomic molecules. University of Waterloo Chemical Physics Research Report CP-650 R²; 2004 [see <http://www.leroy.uwaterloo.ca/programs/>].
- Dunham JL. *Phys Rev* 1932;41:721.
- Herzberg G. *Spectra of diatomic molecules*. New York: van Nostrand; 1950.
- Tellinghuisen J. *J Mol Spectrosc* 1987;122:455.
- Hutson JM. *J Phys At Mol Phys* 1981;14:851.
- Bernath PF. *Spectra of atoms and molecules*, 2nd ed. New York: Oxford University Press; 2005.
- Hansson A, Watson JKG. *J Mol Spectrosc* 2005;233:169.
- An equally valid approach would be to calculate the expectation value $\langle r \rangle$ and see whether it lies in the inner or outer well.
- Šurkus AA, Rakauskas RJ, Bolotin AB. *Chem Phys Lett* 1984;105:291.
- Seto JY. Direct fitting of analytic potential functions to diatomic molecule spectroscopic data [M.Sc. thesis]. Department of Chemistry, University of Waterloo; 2000.
- Simons G, Parr RG, Finlan JM. *J Chem Phys* 1973;59:3229.
- Ogilvie JF. *Proc R Soc Lond A* 1981;378:287.
- Lee EG, Seto JY, Hirao T, Bernath PF, Le Roy RJ. *J Mol Spectrosc* 1999;194:197.
- Seto JY, Morbi Z, Charron F, Lee SK, Bernath PF, Le Roy RJ. *J Chem Phys* 1999;110:11756.
- Huang Y. Determining analytical potential energy functions of diatomic molecules by direct fitting [M.Sc. thesis]. Department of Chemistry, University of Waterloo; 2001.
- Le Roy RJ, Dattani N, Coxon JA, Ross AJ, Crozet P, Linton C. *J Chem Phys* 2009;131:204309.
- Le Roy RJ, Haugen CC, Tao J, Li H. *Mol Phys* 2011;109:435.
- Morse PM. *Phys Rev* 1929;34:57.
- Hua W. *Phys Rev A* 1990;42:2524.
- Le Roy RJ, Henderson RDE. *Mol Phys* 2007;105:663.
- Margenau H. *Rev Mod Phys* 1939;11:1.
- Hirschfelder JO, Curtiss CF, Bird RB. *Molecular theory of gases and liquids*. New York: Wiley; 1964.
- Hirschfelder JO, Meath WJ. Intermolecular forces. In: Hirschfelder JO, editor. *Advances in chemical physics*, vol. 12. New York: Interscience; 1967. p. 3–106 [Chapter 1].
- Le Roy RJ, Bernstein RB. *J Chem Phys* 1970;52:3869.
- Cho Y-S, Le Roy RJ. *J Chem Phys* 2016;144:024311.
- Tao J, Le Roy RJ, Pashov A. Merging of the spline-pointwise and MLR potential forms. In: 65th Ohio State University international symposium on molecular spectroscopy, Columbus, Ohio; 2010. p. MG07.
- Le Roy RJ, Pashov A. betaFIT: a computer program to fit pointwise potentials to selected analytic functions. *J Quant Spectrosc Radiat Transf*; 2016. <http://dx.doi.org/10.1016/j.jqsrt.2016.03.036> (this issue).
- Pashov A, Jastrzębski W, Kowalczyk P. *Comput Phys Commun* 2000;1:622.
- Ivanova M, Stein A, Pashov A, Knöckel H, Tiemann E. *J Chem Phys* 2011;134:024321.
- Douketis C, Scoles G, Marchetti S, Zen M, Thakkar AJ. *J Chem Phys* 1982;76:3057.
- Tang KT, Toennies JP. *J Chem Phys* 1984;80:3726.
- Gunton W, Semczuk M, Dattani NS, Madison KW. *Phys Rev A* 2013;88:062510.
- Dattani N, Le Roy RJ. *J Mol Spectrosc* 2011;268:199.
- Semczuk M, Li X, Gunton W, Haw M, Dattani NS, Witz J, et al. *Phys Rev A* 2013;87:052505.
- Hepburn J, Scoles G, Penco R. *Chem Phys Lett* 1975;36:451.
- Aziz R, Nain V, Carley J, Taylor W, McConville G. *J Chem Phys* 1979;70:4330.
- Tang KT, Toennies JP. *J Chem Phys* 2003;118:4976.
- Hellmann R, Bich E, Vogel E. *Mol Phys* 2008;10:133.
- Patkowski K, Szalewicz K. *J Chem Phys* 2010;133:094304.
- Waldrop JM, Song B, Patkowski K, Wang X. *J Chem Phys* 2015;142:204307.
- Samuelis C, Tiesinga E, Laue T, Elbs M, Knöckel H, Tiemann E. *Phys Rev A* 2000;63:012710.
- While not used in **LEVEL**, LNPT is retained in the calling sequences to facilitate the use of this subroutine package in other programs.

- [60] Le Roy RJ. Molecular spectroscopy. In: Barrow RN, Long DA, Millen DJ, editor. Specialist periodical report 3, vol. 1. London: Chemical Society of London; 1973. p. 113–76.
- [61] Le Roy RJ. *J Chem Phys* 1980;73:6003.
- [62] Le Roy RJ. *J Chem Phys* 1994;101:10217.
- [63] Le Roy RJ, Huang Y. *J Mol Struct (Theochem)* 2002;591:175.
- [64] Mohr P, Taylor B, Newell D. *Rev Mod Phys* 2012;84:1527.
- [65] Wang M, Audi G, Wapstra A, Kondev F, MacCormick M, Xu X, et al. *Chin Phys C* 2012;36:1603.
- [66] Meshkov V, Stolyarov A, Le Roy RJ. *Phys Rev A* 2008;78:052510.
- [67] Meshkov VV, Stolyarov AV, Le Roy RJ. *J Chem Phys* 2011;135:154108.
- [68] Coxon JA, Hajigeorgiou PG. *J Mol Spectrosc* 1999;193:306.
- [69] Aubert-Frécon M, Hadinger G, Magnier S, Rousseau S. *J Mol Spectrosc* 1998;188:182.
- [70] Le Roy RJ. *J Mol Spectrosc* 1999;194:189.
- [71] Peyerimhoff SD, Buenker RJ. *Chem Phys* 1981;57:279.
- [72] Le Roy RJ, Appadoo DRT, Anderson K, Shayesteh A, Gordon IE, Bernath PF. *J Chem Phys* 2005;123:204304.
- [73] Henderson RDE, Shayesteh A, Tao J, Haugen CC, Bernath PF, Le Roy RJ. *J Phys Chem A* 2013;117:13373.
- [74] Allard O, Pashov A, Knöckel H, Tiemann E. *Phys Rev A* 2002;66:042503.
- [75] Barrow D, Slaman M, Aziz R. *J Chem Phys* 1991;91:6348.
- [76] Jäger B, Hellmann R, Bich E, Vogel E. *Mol Phys* 2009;107:2181 [erratum: Jäger B, Hellmann R, Bich E, Vogel E. *Mol Phys* 2010;108:105].