2D Vibron Model Program Suite

Curro Perez-Bernal*[1], Jamil Khalouf Rivera[1], and Miguel Carvajal Zaera[1]

[1]Depto. CC. Integradas y Centro de Estudios Avanzados en Física, Matemáticas y Computación, Fac. CC. Experimentales, Universidad de Huelva

May 3, 2019

Contents

1 Abstract

This is the documentation of the `2DVM_U3` program suite that is composed of mainly `Fortran 90` but also `Perl` and `Python` scripts to perform calculations using the 2D limit of the Vibron model (2DVM).

2 Theory Outline

The original paper where the 2DVM was presented is Ref.~[**Iachello1996**]. We also make extensive use of the detailed description of the model provided in Ref.~[**PBernal2008**]. Other references deal mainly with the application of the 2DVM to molecular bending vibrations [**Ishikawa2002**, **Iachello2003**, **PBernal2005**], quantum phase transitions [**Caprio2008**, **PBernal2010**, **PFernandez2011**, **PRA86**, **Santos2015**, **santos2016**, **Castanos2016**, **PB_Santos2016**], or both [**Larese2011**, **algmonod1**]. New developments also consider the coupling of several $U(3)$ dynamical algebras for the modeling of systems with several degrees of freedom [**mp_u3xu3**, **pla_376**, **cocoyoc**, **Larese2014**, **Calixto2014**].

We proceed to review the main theory results used in this program suite.

The 2DVM has an $U(3)$ dynamical algebra whose nine generators are built as bilinear products of one scalar $\sigma^\dagger$ and two circular $\tau^\dagger_\pm$ bosonic creation operators with the corresponding annihilation operators, $\sigma, \tau_\pm$ [**PBernal2008**]. The nine generators are recasted to provide a deeper physical insight as [**Iachello1996**]

$$\begin{array}{lcl} \hat{n} = \tau^\dagger_+\tau_+ + \tau^\dagger_-\tau_- & , & \hat{n}_s = \sigma^\dagger\sigma \\ \hat{\ell} = \tau^\dagger_+\tau_+ - \tau^\dagger_-\tau_- & , & \hat{Q}_\pm = \sqrt{2}\,\tau^\dagger_\pm\tau_\mp \\ \hat{D}_\pm = \sqrt{2}(\pm\tau^\dagger_\pm\sigma \mp \sigma^\dagger\tau_\mp) & , & \hat{R}_\pm = \sqrt{2}(\tau^\dagger_\pm\sigma + \sigma^\dagger\tau_\mp)\ . \end{array} \tag{1}$$

*curropb@uhu.es

There are two possible subalgebra chains starting from the dynamical algebra $U(3)$ and ending in the symmetry algebra $SO(2)$. The symmetry algebra has the 2D angular momentum $\hat{\ell}$ as generator, which imply that this is a conserved quantity:

$$\begin{array}{ccccc} & & U(2) & & Chain(I) \\ & \nearrow & & \searrow & \\ U(3) & & & & SO(2) \\ & \searrow & & \nearrow & \\ & & SO(3) & & Chain(II) \end{array} \tag{2}$$

The corresponding subalgebras are composed by the following elements

$$\begin{array}{cc} U(2) & \{\hat{n},\hat{\ell},\hat{Q}_+,\hat{Q}_-\} \\ SO(3) & \{\hat{l},\hat{D}_+,\hat{D}_-\} \\ SO(2) & \{\hat{l}\} \end{array} \tag{3}$$

where the $SO(3)$ elements satisfy the usual angular momentum commutation relations. Because of an automorphism of the Lie algebra $U(3)$ constructed with $\tau_\pm$, σ there is an alternative $SO(3)$ subalgebra of $U(3)$, called $\overline{SO(3)}$, with elements

$$\overline{SO(3)} \quad \{\hat{l},\hat{R}_+,\hat{R}_-\} \ . \tag{4}$$

Another ingredient of the algebraic approach are the Casimir (or invariant) operators associated to each subalgebra chain [**frank**, **bookalg**]. The first and second order Casimir operators for the subalgebras in Eqs.\ (**??**) and (**??**) are

$$\begin{array}{l} \hat{C}_1[U(2)] = \hat{n} \quad , \quad \hat{C}_2[U(2)] = \hat{n}(\hat{n}+1) \\ \hat{C}_2[SO(3)] = \hat{W}^2 = (\hat{D}_+\hat{D}_- + \hat{D}_-\hat{D}_+)/2 + \hat{l}^2 \\ \hat{C}_1[SO(2)] = \hat{l} \quad , \quad \hat{C}_2[SO(2)] = \hat{l}^2 \end{array} \tag{5}$$

and

$$\hat{C}_2[\overline{SO(3)}] = \hat{\overline{W}}^2 = (\hat{R}_+\hat{R}_- + \hat{R}_-\hat{R}_+)/2 + \hat{\ell}^2 \ . \tag{6}$$

We proceed to state the basis quantum numbers and branching rules for each dynamical symmetry.

2.1 $U(2)$ or Cylindrical Oscillator Dynamical Symmetry

In this chain states are labeled by the quantum numbers n and ℓ

$$\left| \begin{array}{ccccc} U(3) & \supset & U(2) & \supset & SO(2) \\ [N] & & n & & \ell \end{array} \right\rangle \ , \tag{7}$$

and the basis is $|[N];n,\ell\rangle$. An alternative notation, often used in molecular physics, is $|[N];n^\ell\rangle$. The quantum number N labels the totally symmetric representation of $U(3)$, $[N]$, and it is related to the total number of bound states of the system, n is the vibrational quantum number, and ℓ is the vibrational angular momentum. The branching rules are

$$\begin{array}{l} n = N, N-1, N-2, \ldots, 0 \\ l = \pm n, \pm(n-2), \ldots, \pm 1 \text{ or } 0 \ , \quad (n = \text{odd or even}) \ . \end{array} \tag{8}$$

This is the natural basis to fit experimental bending vibration data in the case of linear and quasilinear molecules.

2.2 $SO(3)$ or Displaced Oscillator Dynamical Symmetry

States in this chain are characterized by the quantum numbers

$$\left| \begin{array}{ccccc} U(3) & \supset & SO(3) & \supset & SO(2) \\ [N] & & \omega & & \ell \end{array} \right\rangle \ . \tag{9}$$

and will be labelled as $|[N];\omega,\ell\rangle$. The branching rules are

$$\begin{aligned} &\omega = N, N-2, N-4, \ldots, 1 \text{ or } 0 \ , \ \ (N = \text{odd or even}), \\ &l = \pm\omega, \pm(\omega-1), \ldots, 0 \ . \end{aligned} \tag{10}$$

In this case, it is convenient to introduce a vibrational quantum number v, which can be identified with the number of quanta of excitation in the displaced oscillator:

$$v = \frac{N-\omega}{2} \ . \tag{11}$$

The branching rules in this case are

$$\begin{aligned} v &= 0, 1, \ldots, \frac{N-1}{2} \text{ or } \frac{N}{2} \ , \ \ (N = \text{odd or even}), \\ l &= 0, \pm1, \pm2, \ldots, \pm(N-2v) \ . \end{aligned} \tag{12}$$

This is the natural basis to fit experimental bending vibration data in the case of quasi-rigid and rigidly-bent molecules.

2.3 2DVM Hamiltonians included in the program suite

We present the three different Hamiltonians that are considered in the present programs. The simplest Hamiltonian, used mainly in the study of QPTs and ESQPTs, is the so called model Hamiltonian, that includes the $\hat{n} = \tau_+^\dagger \tau_+ + \tau_-^\dagger \tau_-$ operator from the dynamical symmetry (I) (DS-I) and the) and the Pairing operator $\hat{P}$ which is built adding a diagonal contribution to the $SO(3)$ second order Casimir operator $\hat{W}^2 = (\hat{D}_+\hat{D}_- + \hat{D}_-\hat{D}_+)/2 + \hat{\ell}^2$. The Pairing operator is $\hat{P} = N(N+1) - \hat{W}^2$. the $SO(3)$ algebra).

The model Hamiltonian $\hat{\mathscr{H}}$ is then

$$\hat{\mathscr{H}} = \varepsilon \left[(1-\xi)\hat{n} + \frac{\xi}{N-1}\hat{P} \right] \ . \tag{13}$$

The Hamiltonian has two parameters: a global energy scale ε and a control parameter ξ such that for $\xi = 0.0$ the system is in the DS-I and for $\xi = 1.0$ the system is in the DS-I. This is specially adequate to characterize the ground state and excited state QPTs as there is a second order ground state QPT for the critical value $\xi_c = 0.2$ [**PBernal2008**].

A second Hamiltonian is $\hat{H}_{2b}$, the most general one- and two-body Hamiltonian of the 2DVM defined in [**Iachello1996**].

$$\hat{H}_{2b} = E_0 + \varepsilon\hat{n} + \alpha\hat{n}(\hat{n}+1) + \beta\hat{\ell}^2 + A\hat{P} \ , \tag{14}$$

where the operators $\hat{n}$ and $\hat{n}(\hat{n}+1)$ belong to the DS-I and the pairing operator $\hat{P}$ to the DS-II while the vibrational angular momentum, $\hat{\ell}$, is common to both dynamical chains. In fact in all the cases considered the angular momentum is a constant of the motion, ℓ is a good quantum number, and the Hamiltonian are block diagonal in ℓ, a fact that simplifies the numerical calculations reducing the dimension of the considered matrices. In addition to this, as in the absence of external fields, the first

order angular momentum operator $\hat{\ell}$ is not included there is a degeneracy between positive and negative ℓ values for $\ell \neq 0$ and only $\ell \geq 0$ are considered. The formula for the dimension of a subspace of the totally symmetric representation N with angular momentum ℓ is $(N-|\ll|)/2+1$, if N and ℓ have the same parity, and $(N-|\ell|+1)/2$ if they have different parity. Both cases are considered simultaneously in the programs as

$$dim(N,\ell) = \frac{N-|\ell|+mod(N-|\ell|,2)+1}{2} \quad . \tag{15}$$

The third and last Hamiltonian considered is $\hat{H}_{4b}$, the most general 1-, 2-, 3-, and 4-body Hamiltonian, that can be expressed in terms of the Casimir operators $\hat{n}$, $\hat{n}^2$, $\hat{\ell}^2$, $\hat{W}^2$, and $\hat{\overline{W}}^2$ can be expressed as follows

$$\begin{aligned}
\hat{H}_{4b} =& P_{11}\hat{n} \\
&+ P_{21}\hat{n}^2 + P_{22}\hat{\ell}^2 + P_{23}\hat{W}^2 \\
&+ P_{31}\hat{n}^3 + P_{32}\hat{n}\hat{\ell}^2 + P_{33}(\hat{n}\hat{W}^2 + \hat{W}^2\hat{n}) \\
&+ P_{41}\hat{n}^4 + P_{42}\hat{n}^2\hat{\ell}^2 + P_{43}\hat{\ell}^4 + P_{44}\hat{\ell}^2\hat{W}^2 \\
&+ P_{45}(\hat{n}^2\hat{W}^2 + \hat{W}^2\hat{n}^2) + P_{46}\hat{W}^4 + P_{47}(\hat{W}^2\hat{\overline{W}}^2 + \hat{\overline{W}}^2\hat{W}^2)/2 \; .
\end{aligned} \tag{16}$$

The Hamiltonian has fourteen spectroscopic constants P_{ij}, where the subindeces indicate that this parameter corresponds to the j-th operator of the i-body operators.

From the matrix elements of the creation and annihilation sigma and tau bosons in the two dynamical symmetries published in [**PBernal2008**], the ensuing operator matrix elements can be derived.

2.3.1 Operator Matrix Elements in the Dynamical Symmetry (I)

The diagonal operators in this dynamical symmetry are

Operator $\hat{n}^p$ $\langle [N];n^\ell|\hat{n}^p|[N];n^\ell\rangle = n^p$ for $p = 1,2,3,4$.

Operator $\hat{\ell}^{2q}$ $\langle [N];n^\ell|\hat{\ell}^{2q}|[N];n^\ell\rangle = \ell^{2q}$ for $q = 1,2$.

Operator $\hat{n}^p\hat{\ell}^{2q}$ $\langle [N];n^\ell|\hat{n}^p\ell^{2q}|[N];n^\ell\rangle = n^p\ell^{2q}$ for $p = 1,2$ and $q = 1$.

The non-diagonal matrix elements in this basis are

$SO(3)$ **Casimir Operator** $\hat{W}^2$

$$\begin{aligned}
\langle [N];n_2^l|\hat{W}^2|[N];n_1^l\rangle =& \left[(N-n_1)(n_1+2)+(N-n_1+1)n_1+l^2\right]\delta_{n_2,n_1} \\
&-\sqrt{(N-n_1+2)(N-n_1+1)(n_1+l)(n_1-l)}\,\delta_{n_2,n_1-2} \\
&-\sqrt{(N-n_1)(N-n_1-1)(n_1+l+2)(n_1-l+2)}\,\delta_{n_2,n_1+2} \quad .
\end{aligned} \tag{17}$$

Note that this is the main nondiagonal operator in this case and it is a band matrix as the non-zero matrix elements are located in the main and first diagonals only.

Operator $\hat{n}\hat{W}^2+\hat{W}^2\hat{n}$ As the operator $\hat{n}$ is diagonal the matrix is also band diagonal with matrix elements

$$\begin{aligned}\langle[N];n_2^l|\hat{n}\hat{W}^2+\hat{W}^2\hat{n}|[N];n_1^l\rangle =&2n_1\left[(N-n_1)(n_1+2)+(N-n_1+1)n_1+l^2\right]\delta_{n_2,n_1}\\ &-(2n_1-2)\sqrt{(N-n_1+2)(N-n_1+1)(n_1+l)(n_1-l)}\,\delta_{n_2,n_1-2}\\ &-(2n_1+2)\sqrt{(N-n_1)(N-n_1-1)(n_1+l+2)(n_1-l+2)}\,\delta_{n_2,n_1+2}\ .\end{aligned}\tag{18}$$

Operator $\hat{\ell}^2\hat{W}^2$ This operator is computed for $\ell\neq 0$ multiplying the $\hat{W}^2$ operator matrix times ℓ^2.

Operator $\hat{n}^2\hat{W}^2+\hat{W}^2\hat{n}^2$ This is computed as the $\hat{n}\hat{W}^2+\hat{W}^2\hat{n}$ operator.

$$\begin{aligned}\langle[N];n_2^l|\hat{n}^2\hat{W}^2+\hat{W}^2\hat{n}^2|[N];n_1^l\rangle =&2n_1^2\left[(N-n_1)(n_1+2)+(N-n_1+1)n_1+l^2\right]\delta_{n_2,n_1}\\ &-[n_1^2+(n_1-2)^2]\sqrt{(N-n_1+2)(N-n_1+1)(n_1+l)(n_1-l)}\,\delta_{n_2,n_1-2}\\ &-[n_1^2+(n_1+2)^2]\sqrt{(N-n_1)(N-n_1-1)(n_1+l+2)(n_1-l+2)}\,\delta_{n_2,n_1+2}\ .\end{aligned}\tag{19}$$

Operator $\hat{W}^4$ This operator is computed as the matrix product of the $\hat{W}^2$ operator matrix times itself.

Operator $\hat{W}^2\hat{\overline{W}}^2+\hat{\overline{W}}^2\hat{W}^2$ In this basis the only difference between the matrix elements of the $\hat{W}^2$ and $\hat{\overline{W}}^2$ operators is the sign of the non-diagonal contribution, which is positive in this case. The full operator is computed via matrix multiplication.

2.3.2 Operator Matrix Elements in the Dynamical Symmetry (II)

The diagonal operators in this dynamical symmetry are

$SO(3)$ **Casimir Operator** $\hat{W}^2$ $\langle[N];\omega\ell|\hat{W}^2|[N];\omega\ell\rangle=\omega(\omega+1)$.

Operator $\hat{\ell}^{2q}$ $\langle[N];\omega\ell|\hat{\ell}^{2q}|[N];\omega\ell\rangle=\ell^{2q}$ for $q=1,2$.

Operator $\hat{\ell}^2\hat{W}^2$ $\langle[N];\omega\ell|\hat{\ell}^2\hat{W}^2|[N];\omega\ell\rangle=\ell^2\omega(\omega+1$.

Operator $\hat{W}^4$ $\langle[N];\omega\ell|\hat{W}^2|[N];\omega\ell\rangle=\omega^2(\omega+1)^2$.

The non-diagonal matrix elements in this basis are

Operator $\hat{n}$

$$\langle [N];w_2^l|\hat{n}|[N];w_1^l\rangle = \left\{ \frac{(N-w_1)\left[(w_1-l+2)(w_1-l+1)+(w_1+l+2)(w_1+l+1)\right]}{2(2w_1+1)(2w_1+3)} \right.$$
$$\left. + \frac{(N+w_1+1)\left[(w_1+l)(w_1+l-1)+(w_1-l)(w_1-l-1)\right]}{2(2w_1+1)(2w_1-1)} \right\} \delta_{w_2,w_1} \tag{20}$$
$$+\sqrt{\frac{(N-w_1)(N+w_1+3)(w_1-l+2)(w_1-l+1)(w_1+l+2)(w_1+l+1)}{(2w_1+1)(2w_1+3)^2(2w_1+5)}}\,\delta_{w_2,w_1+2}$$
$$+\sqrt{\frac{(N-w_1+2)(N+w_1+1)(w_1-l)(w_1-l-1)(w_1+l)(w_1+l-1)}{(2w_1-3)(2w_1-1)^2(2w_1+1)}}\,\delta_{w_2,w_1-2} \tag{21}$$

Note that this is the main nondiagonal operator in this case and it is again a band matrix with non-zero matrix elements are located in the main and first diagonals only. The $\hat{n}$ matrix element in this basis are taken from [**PBernal2008**] with a typo that has been corrected.

Operators $\hat{n}^2$, $\hat{n}^3$, and $\hat{n}^4$ These three operators are computed by matrix multiplication of the basic operator (**??**)

Operator $\hat{n}\hat{\ell}^2$ This operator is computed for $\ell \neq 0$ multiplying the $\hat{n}$ operator matrix times ℓ^2.

Operator $\hat{n}\hat{W}^2+\hat{W}^2\hat{n}$ As the operator $\hat{n}$ is band diagonal $\hat{W}^2$ is diagonal this operator matrix is also band diagonal with matrix elements

$$\langle [N];w_2\ell|\hat{n}\hat{W}^2+\hat{W}^2\hat{n}|[N];w_1\ell\rangle = 2\omega_1(\omega_1+1)\left\{ \frac{(N-w_1)\left[(w_1-l+2)(w_1-l+1)+(w_1+l+2)(w_1+l+1)\right]}{2(2w_1+1)(2w_1+3)} \right.$$
$$\left. + \frac{(N+w_1+1)\left[(w_1+l)(w_1+l-1)+(w_1-l)(w_1-l-1)\right]}{2(2w_1+1)(2w_1-1)} \right\} \delta_{w_2,w_1} \tag{22}$$
$$+\left[\omega_1(\omega_1+1)+(\omega_1+2)(\omega_1+3)\right]$$
$$\times\sqrt{\frac{(N-w_1)(N+w_1+3)(w_1-l+2)(w_1-l+1)(w_1+l+2)(w_1+l+1)}{(2w_1+1)(2w_1+3)^2(2w_1+5)}}\,\delta_{w_2,w_1+2}$$
$$+\left[(\omega_1-2)(\omega_1-1)+\omega_1(\omega_1+1)\right]$$
$$\times\sqrt{\frac{(N-w_1+2)(N+w_1+1)(w_1-l)(w_1-l-1)(w_1+l)(w_1+l-1)}{(2w_1-3)(2w_1-1)^2(2w_1+1)}}\,\delta_{w_2,w_1-2}$$

Operator $\hat{n}^2\hat{W}^2+\hat{W}^2\hat{n}^2$ This is computed in the same way that the $\hat{n}\hat{W}^2+\hat{W}^2\hat{n}$ operator but taking into account that the $\hat{n}^2$ operator is double banded. Therefore the operator matrix elements can be expressed as follow

$$\langle [N];w_2\ell|\hat{n}^2\hat{W}^2+\hat{W}^2\hat{n}^2|[N];w_1\ell\rangle = 2\omega_1(\omega_1+1)[\hat{n}^2]_{w_1,w_1}\,\delta_{w_2,w_1}$$
$$+\left[\omega_1(\omega_1+1)+(\omega_1+2)(\omega_1+3)\right][\hat{n}^2]_{w_1,w_1+2}\delta_{w_2,w_1+2}$$
$$+\left[(\omega_1-2)(\omega_1-1)+\omega_1(\omega_1+1)\right][\hat{n}^2]_{w_1,w_1-2}\delta_{w_2,w_1-2} \tag{23}$$
$$+\left[\omega_1(\omega_1+1)+(\omega_1+4)(\omega_1+5)\right][\hat{n}^2]_{w_1,w_1+4}\delta_{w_2,w_1+4}$$
$$+\left[(\omega_1-4)(\omega_1-3)+\omega_1(\omega_1+1)\right][\hat{n}^2]_{w_1,w_1-4}\delta_{w_2,w_1-4}\ ,$$

where $[\hat{n}^2]_{w_i,w_j}$ are the $\hat{n}^2$ operator matrix elements.

Operator $\hat{W}^2\hat{\overline{W}}^2+\hat{\overline{W}}^2\hat{W}^2$ In this basis we need first to compute the matrix elements of the $\hat{\overline{W}}^2$ making use of Eqs.\ (37) and (38) of Ref.\ [**PBernal2008**].

$$\langle[N];w_2\ell_2|\hat{R}_-|[N];w_1\ell_1\rangle = A_{w_1,\ell_1}\delta_{w_2,w_1}\delta_{\ell_2,\ell_1-1}+B_{w_1,\ell_1}\delta_{w_2,w_1-2}\delta_{\ell_2,\ell_1-1}+C_{w_1,\ell_1}\delta_{w_2,w_1+2}\delta_{\ell_2,\ell_1-1}\ , \quad (24)$$

where

$$\begin{aligned}
A_{w,\ell} =& \frac{(2N+3)(2\ell+1)}{(2w-1)(2w+3)}\sqrt{(w+\ell)(w-\ell+1)/2}\\
B_{w,\ell} =& -\sqrt{\frac{2(N+w+1)(N-w+2)(w+\ell)(w-\ell)(w+\ell-1)(w+\ell-2)}{(2w+1)(2w-1)^2(2w-3)}}\\
C_{w,\ell} =& \sqrt{\frac{2(N+w+3)(N-w)(w+\ell+1)(w-\ell+1)(w-\ell+2)(w-\ell+3)}{(2w+1)(2w+3)^2(2w+5)}}\ .
\end{aligned}$$

The previous result can be used for the obtention of an expression for the $\hat{R}_+$ operator matrix elements

$$\begin{aligned}
\langle[N];w_2\ell_2|\hat{R}_-|[N];w_1\ell_1\rangle^\dagger =& \langle[N];w_1\ell_1|\hat{R}_+|[N];w_2\ell_2\rangle \quad (25)\\
=& A_{w_2,\ell_2+1}\delta_{w_1,w_2}\delta_{\ell_1,\ell_2+1}+B_{w_2+2,\ell_2+1}\delta_{w_1,w_2-2}\delta_{\ell_1,\ell_2+1}\\
&+C_{w_2-2,\ell_2+1}\delta_{w_1,w_2+2}\delta_{\ell_1,\ell_2+1}\ .
\end{aligned}$$

The upper diagonal matrix elements of the Casimir operator $\hat{\overline{W}}^2=\hat{R}_+\hat{R}_-+\hat{\ell}^2$ can then be expressed as

$$\begin{aligned}
\langle[N];w_2\ell|\hat{\overline{W}}^2|[N];w_1\ell\rangle =& A^2_{w_1,\ell_1}+B^2_{w_1,\ell_1}+C^2_{w_1,\ell_1}\,\delta_{w_2,w_1}\\
&+(A_{w_1,\ell}B_{w_1+2,\ell}+C_{w_1,\ell}A_{w_1+2,\ell})\delta_{w_2,w_1+2} \quad (26)\\
&+C_{w_1,\ell}B_{w_1+4,\ell})\delta_{w_2,w_1+4}\ ,
\end{aligned}$$

and the lower diagonal matrix elements can be computed from this equation.

The $\hat{W}^2\hat{\overline{W}}^2+\hat{\overline{W}}^2\hat{W}^2$ operator is then computed as in Eq. (**??**)

$$\begin{aligned}
\langle[N];w_2\ell|\hat{W}^2\hat{\overline{W}}^2+\hat{\overline{W}}^2\hat{W}^2|[N];w_1\ell\rangle =& 2\omega_1(\omega_1+1)[\hat{\overline{W}}^2]_{w_1,w_1}\,\delta_{w_2,w_1}\\
&+[\omega_1(\omega_1+1)+(\omega_1+2)(\omega_1+3)]\,[\hat{\overline{W}}^2]_{w_1,w_1+2}\delta_{w_2,w_1+2}\\
&+[(\omega_1-2)(\omega_1-1)+\omega_1(\omega_1+1)]\,[\hat{\overline{W}}^2]_{w_1,w_1-2}\delta_{w_2,w_1-2} \quad (27)\\
&+[\omega_1(\omega_1+1)+(\omega_1+4)(\omega_1+5)]\,[\hat{\overline{W}}^2]_{w_1,w_1+4}\delta_{w_2,w_1+4}\\
&+[(\omega_1-4)(\omega_1-3)+\omega_1(\omega_1+1)]\,[\hat{\overline{W}}^2]_{w_1,w_1-4}\delta_{w_2,w_1-4}\ ,
\end{aligned}$$

where $[\hat{n}^2]_{w_i,w_j}$ are the $\hat{n}^2$ operator matrix elements.

3 Programs included in the package

We proceed to describe the different programs provided. Compiled `Fortran` programs have names with the suffix `_gfortran` or `_ifort` depending on the compiler used.

The subroutine used for matrix diagonalization is `LA_SYEVR` from the `Lapack95` [**LAPACK95**], the `Fortran 95` interface to `Lapack` [**laug**].

3.1 `Fortran` programs to compute eigenvectors, eigenvalues and Inverse Participation Ratio (IPR)

The names of the programs `vibron2D_[Ham]_[Chain]_[CompOpt]_[Compiler]` mean:

- [Ham]: Hamiltonian to be used. It must be replace by `Hmod`, `H2b` or `H4b` to choose one of the Hamiltonians of eqs. (**??**), (**??**) or (**??**) respectively.
- [Chain]: It must be `U3_U2` (`U2_SO3`) to use the linear (displaced oscillator) basis.
- [CompOpt]: This option is only available if the program has been compiled using the GNU compiler and allowing parallelization with openmp. In this case `[CompOpt]=openmp`.
- [Compiler]: Name of the compiler used. The source code has been written to compile with GNU compiler (gfortran) or the intel's one (ifort).

For example, if the 2body Hamiltinian of 2DVM is requeried, it's just necessary to type `vibron2D_H2b_U3_U2_gfo` in order to use the Hamiltonian of eq. (**??**) diagonalized in the basis of Chain I and compiled sequentially.

Each program needs its own input file, in spite of sharing global variables. They are stored in `test` folder and have been based on `fortran` namelist. The lines `&par_aux` and `&par_0` are the same in the three files. The rest are related with the parameters of the chosen Hamiltonian.

```
#
#       INPUT FILE FOR 2DVM U(3) Calculations
#       Model Hamiltonian
#
&par_aux Iprint = 0, Eigenvec_Log = .T., Excitation_Log = .T., Save_avec_Log = .T.  /
&par_0   N_val = 10, L_min = 0, L_max = 5 /
&par_1   epsilon = 1.0,   xi = 0.4 /
```

The line called `&par_aux` contains controls variables in order to choose the desired output:

- `Iprint` *Integer variable*. Control of verbosity.
- `Eigenvec_Log` *Logical variable*. If `.True.`, The Inverse Participation Ratio (IPR) will be computed.
- `Excitation_Log` *Logical variable*. If `.True.`, the excitation energies will be referred to the ground state.
- `Save_avec_Log` *Logical variable*. If `.True.`, eigenstates will be saved in the current folder.

In the next line, known as `&par_0`, is where user can define the number of bosons N and the interval of angular momentum requiered:

- `N_val` *Integer variable*. Totally symmetric irrep of $U(3)$. Determines the Hilbert space size.
- `L_min` and `L_max` *Integer variables*. Vibrational angular momentum range $\ell \in [\ell_{min}, \ell_{max}]$.

As an example, we present the results of `vibron2D_Hmod_U3_U2_gfortran` setting `Iprint` to 3.

```
$ [...]/bin/vibron2D_Hmod_U3_U2_gfortran  /*Press enter*/
$ u3_Hmod.inp                             /*Type the name of the input file*/
```

The screen will show:

```
Iprint =  3; Eigenvec_LOG =  T; Excitation_Log =  T; Save_Avec_Log =  T
N_val =     10; L_min =      0; L_max =      5
epsilon =  1.0000000E+00; xi =  4.0000000E-01
Operator number            1
Operator number            2
Operator number            1
Operator number            2
Operator number            1
Operator number            2
Operator number            1
Operator number            2
Operator number            1
Operator number            2
Operator number            1
Operator number            2
Operator number            1
Operator number            2
```

The above information is related with the computing process and the input variables. The physical data are stored in the files called `autoval_[...].dat` and `eigvec_[...]`, where the name's tail is generated using the input information. For example: `autoval_mh_u2_N10_L0.dat` or `eigvec_u2_N10_L0.dat`. With `Iprint`>1, both type of files displays the eigenvectors' coefficients. The output of `autoval_mh_u2_N10_L0.` with `Iprint`$=3$ is:

```
 Hamiltonian Matrix
  4.0000000000000000       0.84327404271156792       0.0000000000000000       ...
 0.84327404271156792        3.8666666666666671       1.3303670708529569       ...
  0.0000000000000000        1.3303670708529569       4.4444444444444446       ...
  0.0000000000000000        0.0000000000000000       1.4605934866804429       ...
  0.0000000000000000        0.0000000000000000       0.0000000000000000       ...
  0.0000000000000000        0.0000000000000000       0.0000000000000000       ...
GS_energy =    2.2741144458696825
L_val =            0
          0   4.4408920985006262E-016   2.8418774855178928
-0.32933743241225288      |           0 >
 0.67403796185517229      |           2 >
-0.59811984409478036      |           4 >
 0.27481947241670696      |           6 >
 -6.2557421702539709E-002 |           8 >
  4.8125108622299488E-003 |          10 >
          2   1.4773749338491529        2.6037467907764698
 0.74575014268070683      |           0 >
-0.21977058600834720      |           2 >
-0.45367866587132649      |           4 >
 0.41541639448225348      |           6 >
```

```
-0.13043184131934032      |           8 >
  1.2248931213675825E-002 |          10 >
          4   2.6183838541904825        3.3349012803683000

  ...
```

The first lines (truncated) show the terms of the Hamiltonian matrix. After that, the program displays information about the Zero Point Energy (ZPE) and the angular momentum. The last part alternates a line that contains the vibrational (in this case) quantum number, the energy and the IPR of the levels with the eigenstate in the choosen basis. A simpler version of this output could be obtained setting `Iprint` to 0 (**recommended**):

```
 0   4.4408920985006262E-016   2.8418774855178928
 2   1.4773749338491529        2.6037467907764698
 4   2.6183838541904825        3.3349012803683000
 6   4.1112373940234130        3.7060015095032330
 8   6.0440884180968331        2.0756877528461386
10   8.3264509468442469        1.1333042664402908
```

The `eigvec_u2_N10_L0.dat` file with `Iprint`= 3 contains similar information but different structure. This file will not be generate if `Save_avec_Log` is false.

```
# N =           10 ; L =            0  ; u2 basis
# Eigenvalues
  4.4408920985006262E-016   1.4773749338491529        2.6183838541904825       ...
# Hamiltonian Diagonal
  1.7258855541303175        1.5925522207969847        2.1703299985747622       ...
-0.32933743241225288       0.74575014268070683       0.56404597555589175       ...
 0.67403796185517229      -0.21977058600834720       0.59697091199520735       ...
-0.59811984409478036      -0.45367866587132649       0.10278840979622948       ...
 0.27481947241670696       0.41541639448225348      -0.51221349887982404       ...
 -6.2557421702539709E-002 -0.13043184131934032       0.22778225002450472       ...
  4.8125108622299488E-003   1.2248931213675825E-002  -2.5787373682437546E-002  ...
```

The first line is the information about the system. Next two consist of the eigenvalues and Hamiltonian diagonal. The rest lines are the n-th eigenstate coefficients refer to the chosen basis.

3.2 Programs to fit experimental vibrational bending spectra

This part of the software has two different purposes depending on the input definitions, so, before explaining them, we present experimental energies and input files.

3.2.1 Program `chi2_U3`

`2D Vibron Model Program Suite` includes a `chi2_U3` program to compute the χ^2 and use it in different minimization methods.

This part of the software has two different purposes depending on the input definitions, so, before explaining them, we present experimental energies and input files.

First of all we need to collect the experimental data into a file. In this example we use the `hydrogen isocyanide` (HNC) molecule [**MellauHNC**, **BunkerHNC**].

```
19
462.72    1.0   1    1
926.50    1.0   2    0
936.05    1.0   2    2
1398.56   1.0   3    1
1419.97   1.0   3    3
1867.05   1.0   4    0
1878.72   1.0   4    2
1913.87   1.0   4    4
2341.84   1.0   5    1
2366.83   1.0   5    3
2417.57   1.0   5    5
2809.29   1.0   6    0
2822.75   1.0   6    2
2863.11   1.0   6    4
2930.90   1.0   6    6
3281.50   1.0   7    1
3309.78   1.0   7    3
3367.37   1.0   7    5
3453.78   1.0   7    7
```

The first line must be the total number of experimental data. After that, the file is structured in 4 columns. The first column corresponds with the energy, the next one to the experimental error of data and, the last two are the quantum numbers. If you want to include any data in the prediction but not at the fitting time, you can set the experiemntal error to 0.0 . The energy's label must be in accordance with the nature of the molecule: linear, $U(2)$ basis $|n\ell\rangle$, or bent, $SO(3)$ $\left|v_b{}^{\ell}\right\rangle$.

Once finished, the next step is to construct the input file:

```
 &INP0 BENT=.F., exp_data_file='exp_HNC_Danielle.dat' /
 &INP1 N_val=40, LMAX=7, VMAX=7, EMINL=.F. /
#
# PARAMETERS THAT TUNE THE OUTPUT
#
 &INP2 IPRINT=0, DIS_RES = *** /
 &INP1b P11=1414.0D0 /
 &INP2b P21=-29.837D0, P22=15.81D0, P23=-8.054D0 /
 &INP3b P31=0.0D0, P32=4.9D-2, P33=0.0D0 /
 &INP4b P41=0.0D0, P42=0.0D0, P43=0.0D0, P44=0.0D0, P45=0.0D0, P46=0.0D0, P47=0.0D0 /
```

- `BENT`: `.T.` (`.F.`) if the molecule is bent (linear).
- `exp_data_file`: The name of the experimental energies' file.
- `N_val`: Vibron number. Determines the Hilbert space size.
- `LMAX`: The highest angular momentum number ℓ in `exp_data_file`.
- `VMAX`: The highest harmonic number n for the linear case or the bending number v_b when `BENT` is true.

- EMINL: Fix the lowest level at the zero energy.
- Iprint: Integer variable. Controls the program's verbosity.
- DIS_RES: Match as true (.T.) to display residuals energies or false (.F.) to get only the χ^2.
- P_{ij}: Initial value of the parameter P_{ij}
  Running the program with DIS_RES=.T.:

```
$ echo fit.inp | ../../bin/chi2_U3_gfortran


   7.0788432691415437E-003
  -2.7776246656685544E-004
   9.1877661752732820E-003
  -3.3695279357061736E-002
  -3.0352671044056478E-002
  -8.4165727005256485E-002
  -4.9249332600084017E-002
   5.6619175268906474E-002
  -3.3439091556601852E-002
  -4.1176236554747447E-002
  -5.0164535727162729E-002
   7.7310508308073622E-002
  0.16723874181343490
  -2.9727267470207153E-002
   1.0623963672060199E-002
   5.5108955660216452E-002
 -0.11479814853555581
  0.16486361585430132
  0.19601871556506012
```

On the other hand, if you set DIS_RES to false:

```
$ echo fit.inp | ../../bin/chi2_U3_gfortran

  0.13699774310846663
```

3.2.2 Minimization using the Python-script

3.2.3 Minimization using Minuit-CERN package

In this section we are going to explain how to use this minimization method, where Minuit-CERN package [**minuit**] has been used. Most of the inputs items have been detailed at subsection **??**. The energies file have the same structure and the input file is similar changing the meaning of DIS_RES:

```
 &INP0 BENT=.F., exp_data_file='exp_HNC.dat', output_file_0="output_HNC" /
 &INP1 N_val=40, LMAX=7, VMAX=7, EMINL=.F. /
#
# PARAMETERS THAT TUNE THE OUTPUT
```

```
#
 &INP2 IPRINT=0, DIS_RES = .F. /
 &INP1b P11=740.0D0 /
 &INP2b P21=-3.597D0, P22=1.5D0, P23=-7.2D0 /
 &INP3b P31=0.0D0, P32=0.002D0, P33=0.0D0 /
 &INP4b P41=0.0D0, P42=0.0D0, P43=0.0D0, P44=0.0D0, P45=0.0D0, P46=0.0D0, P47=0.0D0 /
 &fix_par fixed_par = 'fix  5 7 8 9 10 11 12 13 14' /

 # P11 ---> 1
 # P21 ---> 2
 # P22 ---> 3
 # P23 ---> 4
 # P31 ---> 5
 # P32 ---> 6
 # P33 ---> 7
 # P41 ---> 8
 # P42 ---> 9
 # P43 ---> 10
 # P44 ---> 11
 # P45 ---> 12
 # P46 ---> 13
 # P47 ---> 14
```

Most of definitions have been explained in previous sections, the others mean:

- `BENT`: `.T.` (`.F.`) if the molecule is bent (linear).
- `exp_data_file`: The name of the experimental energies' file.
- `output_file_0`: To choose the name of the file where the fit report will be saved.
- `LMAX`: The highest angular momentum number ℓ in `exp_data_file`.
- `VMAX`: The highest harmonic number n for the linear case or the bending number v_b when `BENT` is true.
- `DIS_RES`: Match as true (.T.) to display residuals energies.
- P_{ij}: Initial value of the parameter P_{ij}
- `fixed_par`: `Minuit` order. This string of characters consists of different integers that are associated with the P_{ij} parameters. If one of them appears, `Minuit` will keep it fixed.

If the program `Min_minuit_gfortran` is executed as mentioned:

```
$ echo fit.inp | ../bin/Min_minuit_gfortran

Note: The following floating-point exceptions are signalling: IEEE_INVALID_FLAG IEEE_DIVI

$ tail -n 43 output_HNC

  EXT PARAMETER                APPROXIMATE        STEP         FIRST
```

```
  NO.   NAME        VALUE          ERROR          SIZE      DERIVATIVE
   1     P11        1413.8       0.37063       0.67417E-03  -0.14902
   2     P21       -29.832       0.15233E-01   0.14225E-04   -1.8667
   3     P22        15.807       0.10429       0.75373E-05  -0.14985
   4     P23       -8.0528       0.28648E-02   0.38399E-05   -13.647
   5     P31        0.0000         fixed
   6     P32       0.48757E-01   0.95612E-02   0.65872E-06   -1.3627
   7     P33        0.0000         fixed
   8     P41        0.0000         fixed
   9     P42        0.0000         fixed
  10     P43        0.0000         fixed
  11     P44        0.0000         fixed
  12     P45        0.0000         fixed
  13     P46        0.0000         fixed
  14     P47        0.0000         fixed

 EXTERNAL ERROR MATRIX.    NDIM=  50    NPAR=  5    ERR DEF=  1.00
  0.137E+00-0.432E-02-0.926E-03-0.915E-03 0.200E-03
 -0.432E-02 0.232E-03 0.563E-03 0.154E-04-0.537E-04
 -0.926E-03 0.563E-03 0.109E-01-0.862E-04-0.978E-03
 -0.915E-03 0.154E-04-0.862E-04 0.821E-05 0.628E-05
  0.200E-03-0.537E-04-0.978E-03 0.628E-05 0.914E-04
 ERR MATRIX NOT POS-DEF

 PARAMETER  CORRELATION COEFFICIENTS
       NO.  GLOBAL     1     2     3     4     6
        1  0.99666  1.000-0.766-0.024-0.861 0.056
        2  0.98707 -0.766 1.000 0.355 0.353-0.369
        3  0.99201 -0.024 0.355 1.000-0.288-0.980
        4  0.99374 -0.861 0.353-0.288 1.000 0.229
        6  0.99086  0.056-0.369-0.980 0.229 1.000
 ERR MATRIX NOT POS-DEF
minuit>  **********
 **    4 **CALL    3.000
 **********
  ------------------------------------
 | rms =    7.9576384169698269E-002  |
 |____________________________________|
minuit>  **********
 **    5 **EXIT
 **********
 ..........MINUIT TERMINATED BY MINUIT COMMAND: EXIT

$
```

3.3 `Perl` scripts