A-VCI: a flexible method to efficiently compute vibrational spectra

Isabelle Baraille, Didier Bégué, Olivier Coulaud, Vincent Le Bris, Marc Odunlami

▶ To cite this version:

HAL Id: hal-01485877
https://hal.inria.fr/hal-01485877

Submitted on 9 Mar 2017

informatics *mathematics*

Inria

A-VCI: a flexible method to efficiently compute vibrational spectra

Isabelle Baraille , Didier Bégué, Olivier Coulaud, Vincent Le Bris
Marc Odunlami

A-VCI: a flexible method to efficiently compute vibrational spectra

Isabelle Baraille [*], Didier Bégué[*], Olivier Coulaud[†], Vincent Le Bris[*] Marc Odunlami [*]

Project-Teams HiePACS

Research Report n° 9043 — march 2017 — 35 pages

Abstract: The Adaptive Vibrational Configuration Interaction (A-VCI) algorithm has been introduced as a new method to efficiently reduce the dimension of the set of basis functions used in a Vibrational Configuration Interaction (VCI) process. It is based on the construction of nested basis for the discretization of the Hamiltonian operator according to a theoretical criterion that ensures the convergence of the method. The purpose of this paper is to study the properties and outline the performance details of the main steps of the algorithm. New parameters have been incorporated to increase flexibility, and their influence have been thoroughly investigated. The robustness and reliability of the method are demonstrated for the computation of the vibrational spectrum up to 3000 cm^{-1} of a widely studied 6-atom molecule (acetonitrile). Our results are compared to the most accurate up to date computation, and we also give a new reference calculation for future work on this system. The algorithm has also been applied to a more challenging 7-atom molecule (ethylene oxide). The computed spectrum up to 3200 cm^{-1} is the most accurate computation that exists today on such a system.

Key-words: Vibrational Hamiltonian, vibrational Configuration Interaction, Eigenvalue problem, adaptive method, high dimension, spectrum, acetonitrile, ethylene oxide

[*] IPREM, Université de Pau et des Pays de l'Adour, 2, avenue du Président Pierre Angot, F-64053 Pau Cedex 9
[†] Inria Bordeaux Sud-Ouest, 200, avenue de la Vieille Tour, F-33405 Talence Cedex

A-VCI: une méthode flexible pour calculer rapidement des spectres vibrationnels

Résumé :   L'algorithme adaptatif d'interaction de configuration vibrationnelle (A-VCI) a été introduit comme une nouvelle méthode pour réduire efficacement la dimension de l'ensemble des fonctions de base utilisées dans un processus d'interaction de configuration vibrationnelle (VCI). Il est basé sur la construction de bases emboîtées pour la discrétisation de l'opérateur Hamiltonien selon un critère théorique qui assure la convergence de la méthode. Cet article présente les propriétés de la méthode et décrit les détails des principales étapes de l'algorithme. De nouveaux paramètres sont introduits pour accroître les potentialités de la méthode et leurs influences sont étudiées. La robustesse et la fiabilité de la méthode sont démontrées pour le calcul du spectre vibrationnel jusqu'à 3000 cm $^{-1}$ d'une molécule à 6 atomes (acétonitrile). Nos résultats sont comparés au calcul le plus précis à jour, et nous donnons également un nouveau calcul de référence pour les travaux futurs sur ce système. L'algorithme a également été appliqué à un système plus difficile l'oxyde d'éthylène qui comporte 7 atomes. Le spectre calculé jusqu'à 3200 cm $^{-1}$ est le calcul le plus précis qui existe aujourd'hui sur un tel système.

Mots-clés :   Hamiltonien vibrationnel, interaction de configuration vibrationnelle, problème aux valeurs propres, méthode adaptative, grande dimension, spectre, acétonitrile, oxyde d'éthylène

Contents

1 Introduction 4

2 Notations and general overview 5

3 The A-VCI algorithm 7
  3.1 Algorithm 7
  3.2 Approximation space definition 8
  3.3 Matrix Operations 8
    3.3.1 Sparse structure definition 8
    3.3.2 Sparse structure construction 9
    3.3.3 Sparse structure update 10
    3.3.4 Matrix coefficient evaluation 10
  3.4 Basis expansion strategies 10
    3.4.1 Component-wise strategies 11
    3.4.2 Collective component-wise strategies 11

4 Results and discussion 12
  4.1 Acetonitrile molecule, CH_3CN 12
    4.1.1 Influence of different strategies to expand the A-VCI space 13
    4.1.2 Convergence study of the A-VCI algorithm 13
    4.1.3 New results 15
  4.2 Ethylene oxide molecule, C_2H_4O 17
    4.2.1 The first 50 eigenvalues 17
    4.2.2 The first 200 eigenvalues 18

5 Conclusion 20

A Hermite functions 21

B Supplementary Material 23
  B.1 Acetonitrile molecule 23
  B.2 Ethylene oxide molecule 28

Contents

1 Introduction

The computation of vibrational properties of a polyatomic molecule usually requires solving the time-independant Schrödinger equation, under the Born-Oppenheimer approximation. Within this framework, the vibrational energy levels are solutions of an eigenvalue problem, where the Hamiltonian of the system is discretized in a finite dimensional sub-space of the Hilbert space associated with the system. The variational [10, 15, 8, 9, 32, 47, 42, 18] principle states that the larger the subspace is, the more accurate the eigenpairs is.

Several methods can be used for this discretization. The Hamiltonian and the wave functions can be expanded using finite basis functions (FBR) [38, 5, 8, 41, 36, 20, 20, 1, 2, 3, 13], or sampled on some grid of points in the configuration space (DVR) [11]. An alternative approach is the phase space representation. It uses Gaussian functions localized in phase space (Von Newmann basis set) [44, 25, 26].

The simplest representation uses the harmonic oscillator direct product basis set, since they are exact wavefunctions of the harmonic Hamiltonian. Each function is a product of one-dimensional Hermite polynomials. However, the direct product structure of the basis set makes it unusable for molecular systems of dimension D, with $D = 3N - 6$ ("curse of dimensionality"), due to its memory requirements [35, 11, 12].

Some recent works try to overcome this limitation by finding the smallest basis size to be able to deal with larger systems. There are two common strategies for selecting a subset of a direct product basis [24, 17, 48, 50, 37, 21, 22]. In the first strategy a pruning condition is used to remove basis functions [20, 25, 3, 13] while in the second strategy we begin with a small basis and we enlarge it by adding functions that satisfy some conditions [5, 43, 19, 27, 45, 44, 34, 14, 23, 13, 23].

The first way is then based on a correlated truncation scheme, imposing the pruning condition:

$$\sum_{i=1}^{D} \alpha_i n_i \le b, \tag{1}$$

where the n_i is the maximal degree of the Hermite polynomial related to the coordinate q_i, and b is a convergence parameter. Several choices are possible for α_i depending if we prune only by the ordering of the element in the energy or by an energy criterion. The simplest approach considers $\alpha_i = 1$ (called polyad [25, 40, 29, 16] or binomial [23] truncation) and then retains all polynomials of total degree lower than b. In such approcah, we have the same discretisation in every the directions even if the harmonic frequencies are very different. By using an energy criterion we take into account this difference in the pruning space. Two criteria are classically used $\alpha_i = \omega_i$ (Ref. [25]), where ω_i is the harmonic frequency for the coordinate q_i, $\alpha_i = \left\lfloor \dfrac{\omega_i}{\omega_{min}} + 0.5 \right\rfloor$ (Ref. [13]) or $\alpha_i = \left\lfloor \sqrt{\dfrac{\omega_i}{\omega_{min}}} + 0.5 \right\rfloor$ (Ref. [25, 13]) where ω_{min} is the lowest harmonic frequency. It is also possible to impose a more general condition [1, 2, 4, 3, 13]:

$$\sum_{i=1}^{D} g_i(n_i) \le b. \tag{2}$$

While this condition leads to smaller basis sets than the previous one, the choice of suitable $g_i(n_i)$ functions is difficult, and strongly depends on the studied molecular system. These choices lead to different convergence properties when the parameter b is increased [25], depending on the energy range considered. However it is difficult to choose b to reach a given accuracy on the eigenvalues.

In the second approach, we start with a small basis and we iteratively add functions to expand the basis until we get accurate enough eigenpairs. This requires to define for each iteration: an admissible set of functions, a method to choose which functions to add and a convergence criterion. The functions are selected from the full direct product space[5, 13, 23]. This selection is based on the definition of neighboring functions. These neighbours are defined differently depending on the authors. For example, Brown and Carrington [13] define the neighbours of $\mathbf{n} = \{n_1, \ldots, n_D\}$ by augmenting one of the indices n_i by 1 for each i, and select elements

such that the basis is increased by 5% at each iteration. An alternative idea is to evaluate the contribution of each function to the energy at the second order perturbational level [5], and only select the most significant contributions.

Concerning the convergence criterion, it is generally based on differences between the eigenvalues for different values of b in the first approach, or between consecutive iterations to stop the basis growth in an iterative process. These two choices seems misleading since the convergence curve of the eigenvalues exhibit a "stair-step" shaped pattern (see for example Ref. [13]).

The Adaptative Vibrational Configuration Interaction (A-VCI) algorithm [23] has been recently introduced as a more efficient way to select relevant Hermite functions without any limit on their degree, given an energy range of interest. The potential energy operator is used to build the image space of a given set of functions at each iteration. This iterative approach constructs nested subspaces using a result of perturbation theory for linear operators [6, 28] to define an *a-posteriori* error estimator. This estimator acts both as a condition to build the active space of the configuration interaction process by selecting the most relevant basis elements, as well as a trustworthy convergence criterion on both the eigenvalues and the eigenvectors. A-VCI algorithm is then a robust and general procedure to compute spectrum in the case of sum of product basis without any *a-priori* chemical and physical information and without defining any arbitrary criterion.

The aim of this paper is to detail the different numerical ingredients of the A-VCI algorithm. We establish new growth strategies resulting in smaller final basis sets to calculate spectra of molecules with 6 and 7 atoms. In Section 2 we briefly recall some notations and we set up the key ideas used by the algorithm. Section 3 presents the algorithm and explain how the matrices are built and how different strategies are used to increase the subspace leading to significant cost reduction of the algorithm. In Section 4, we exhibit results for two medium sized molecules, CH_3CN and C_2H_4O, compared to the pruned basis method, and illustrate the good convergence properties of the A-VCI algorithm. Results reported in the Supplementary Material are also compared to the best references that can be obtained using the A-VCI algorithm when pushed to its limits.

2 Notations and general overview

Let introduce the vibrational Hamiltonian $\mathcal{H}$ for a N-atom molecular system:

$$\mathcal{H}(\mathbf{q}) = \mathcal{H}_0(\mathbf{q}) + \mathcal{V}(\mathbf{q}) = \sum_{i=1}^{D} \frac{\omega_i}{2}\left(-\frac{\partial^2}{\partial q_i^2} + q_i^2\right) + \sum_{\|\mathbf{s}\|_1=3}^{S} K_{\mathbf{s}} \prod_{i=1}^{D} q_i^{s_i} \qquad (3)$$

where $\mathcal{H}_0$ is the harmonic Hamiltonian operator, $\mathcal{V}$ the anharmonic part of the Potential Energy Surface (PES) as a Taylor expansion, S the maximal degree of the PES, $D = 3N - 6$ the number of vibrational degrees of freedom of the considered system, $\mathbf{q} = (q_1, q_2, \ldots, q_D)$ the normal dimensionless coordinates and $\|\mathbf{s}\|_1$ the sum of all components of the multi-index $\mathbf{s}$. Each harmonic frequency ω_i and polynomial degree s_i are associated with the q_i coordinate.

The solution of the vibrational Schrödinger eigenvalue problem requires the discretization of the Hamiltonian operator $\mathcal{H}$ in a suitable orthonormal basis set. Since $\mathcal{H}$ can be seen as a perturbation of the operator $\mathcal{H}_0$ that is Hermitian. Consequently a natural choice is to search the eigenpairs of $\mathcal{H}$ in the space $\Pi_{\mathbf{d}}$ spanned by the eigenvectors of $\mathcal{H}_0$ analytically defined by

$$\phi_{\mathbf{n}}^0(\mathbf{q}) = \prod_{i=1}^{D} \psi_{n_i}(q_i), \text{ for all } \mathbf{n} = (n_1, \ldots, n_D), \qquad (4)$$

where the quantum number n_i corresponds to ω_i and ψ_{n_i} is the 1-D normalized Hermite function of degree n_i. The discretization space $\Pi_{\mathbf{d}}$ is a direct-product space of size M and is defined by

$$\Pi_{\mathbf{d}} = \left\{ \phi_{\mathbf{n}}^0 \, / \, \mathbf{n} \in \prod_{i=1}^{D}[0, d_i] \right\}, \qquad (5)$$

where $\mathbf{d} = (d_1, \ldots, d_D)$ and d_i is the maximal degree of the Hermite function with respect to the q_i coordinate.

The discretized Hamiltonian H in $\Pi_{\mathbf{d}}$ is represented by a $M \times M$ matrix, in which each element (i, j) has the form:

$$H_{i,j} = \langle \phi_{\mathbf{i}}^0 | \mathcal{H} \phi_{\mathbf{j}}^0 \rangle, \tag{6}$$

where the row i (resp. the column j) of the matrix corresponds to the index of basis function $\phi_{\mathbf{i}}^0$ (resp. $\phi_{\mathbf{j}}^0$) in $\Pi_{\mathbf{d}}$.

Let B be a subset of $\Pi_{\mathbf{d}}$, we define $\mathcal{H}(B)$ its image space by the operator $\mathcal{H}$, which is spanned by all harmonic functions $\phi_{\mathbf{n}}^0$ such that $\langle \phi_{\mathbf{n}}^0 | \mathcal{H} \phi \rangle \neq 0$ for all ϕ in B. As $\mathcal{H}_0$ is diagonal in the basis of the 1-D Hermite function products, the image space can be decomposed as the direct sum of two orthogonal spaces

$$\mathcal{H}(B) = B \oplus B_R. \tag{7}$$

Let m (resp. m_R) be the number of elements in B (resp. B_R), we denote by $\widetilde{H}$ the Hamiltonian matrix of operator $\mathcal{H}$ in space $\mathcal{H}(B)$. Thanks to the above space decomposition this matrix is decomposed as follows

$$\widetilde{H} = \left(\begin{array}{c|c} H & H_R^T \\ \hline H_R & H_C \end{array} \right), \tag{8}$$

where $H = B^T \mathcal{H} B$ is a $(m \times m)$ Rayleigh matrix that approximates the Hamiltonian operator $\mathcal{H}$ in the orthonormal basis B, $H_R = B_R^T \mathcal{H} B$ a $(m_R \times m)$ matrix and $H_C = B_R^T \mathcal{H} B_R$ a $(m_R \times m_R)$ matrix.

As B is strictly included into $\mathcal{H}(B)$ when we search eigenvalues of $\mathcal{H}$ we have to evaluate the error due to the projection in B. In other words, we want to measure the distance between the eigenvalue of H and the eigenvalue of $\widetilde{H}$. To this end, we introduce an *a-posteriori* error estimator based on the Bauer-Fike theorem [6, 28]. This theorem is a classical tool in spectral perturbation theory to localize eigenvalues and indicates that if ΔH is a symmetric perturbation of a symmetric matrix H then for any eigenpairs $(\widetilde{E}, \widetilde{\mathbf{X}})$ of $H + \Delta H$ with $\|\widetilde{\mathbf{X}}\| = 1$, there is an eigenvalue E of H such that

$$|\widetilde{E} - E| \leq \|\Delta H \widetilde{\mathbf{X}}\| \leq \|\Delta H\|_F \tag{9}$$

where $\|.\|$ is the usual Euclidian norm, and $\|.\|_F$ is the Frobenius norm.

Thanks to the matrix decomposition (8), the matrix $\widetilde{H}$ can be written as

$$\widetilde{H} = \overline{H} + \Delta H = \left(\begin{array}{c|c} H & 0 \\ \hline 0 & 0 \end{array} \right) + \left(\begin{array}{c|c} 0 & H_R^T \\ \hline H_R & H_C \end{array} \right) \tag{10}$$

and it can be viewed as a perturbation of the matrix $\overline{H}$ which is an extension of the matrix H in the image space of B. Let $(E, \mathbf{X})$ be an eigenpair of H, and $\overline{\mathbf{X}} = (\mathbf{X}, \mathbf{0}_{m_R})^T$ its extended eigenvector in $\mathcal{H}(B)$. Then, the residual vector $\mathbf{R} = \widetilde{H}\overline{\mathbf{X}} - E\overline{\mathbf{X}}$ satisfies

$$\mathbf{R} = \widetilde{H}\overline{\mathbf{X}} - E\overline{\mathbf{X}} = \overline{H}\,\overline{\mathbf{X}} - E\overline{\mathbf{X}} + \Delta H \overline{\mathbf{X}} = \Delta H \overline{\mathbf{X}} = (\mathbf{0}_m, H_R \mathbf{X})^T. \tag{11}$$

By using (9) and (11) we obtain the *a-posteriori* error estimator

$$|\widetilde{E} - E| \leq \|\mathbf{R}\| = \|H_R \mathbf{X}\|. \tag{12}$$

The distance between E and the target eigenvalue $\widetilde{E}$ of $\widetilde{H}$ is bounded by $\|R\|$. More precisely, the above inequality shows that if E is an eigenvalue of the matrix H and $\|\mathbf{R}\| \leq \varepsilon$, then E is also a good approximation of an eigenvalue $\widetilde{E}$ of the matrix $\widetilde{H}$ in the space $\mathcal{H}(B)$, which is a larger space.

The next section presents in a detailed fashion how this criterion can be used to control the behavior of the A-VCI algorithm, and provides more information on how it is implemented.

3 The A-VCI algorithm

The A-VCI algorithm introduced in Garnier et. al[23] is an iterative procedure that computes both a minimal basis set in the discretized $\Pi_{\mathbf{d}}$ space and the eigenpairs of the vibrational Hamiltonian discretized in this basis.

The main components of such an iterative procedure are 1) a discretization space and a starting space 2) a method to enlarge the current basis of the active space and 3) a robust criterion to assess the convergence.

In this section we briefly recall the main steps of the A-VCI algorithm, then we describe these different elements and how we efficiently use them in the algorithm.

3.1 Algorithm

Algorithm 1 describes the full A-VCI process to compute both the first F eigenpairs of the Hamiltonian in the discretized space $\Pi_{\mathbf{d}}$ and the final basis set adjusted to the required accuracy.

Algorithm 1: Adaptive vibrational configuration interaction (A-VCI) algorithm

---

Result: The first F eigenpairs of the discretized Hamiltonian and the corresponding minimal basis

begin

1   Define the initial orthonormal basis $B^{(0)}$

2   Build the sparse structure of the matrices $H^{(0)} = B^{(0)^T} \mathcal{H} B^{(0)}$ and $H_R^{(0)} = B_R^{(0)^T} \mathcal{H} B^{(0)}$

    // Start the iterations

    for $j \geq 0$ **do**

3   Build the coefficients of the sparse Rayleigh matrix $H^{(j)} = B^{(j)^T} \mathcal{H} B^{(j)}$

4   Compute the first F eigenpairs of $H^{(j)}$ denoted $(E_\ell^{(j)}, \mathbf{X}_\ell^{(j)})_{\ell=0}^{F-1}$

5   Build the coefficients of the matrix $H_R^{(j)} = B_R^{(j)^T} \mathcal{H} B^{(j)}$

6   Check the convergence

     Exit if the algorithm has converged

7   Expand $B^{(j)}$ with a selected orthonormal subset $A^{(j)}$ of $B_R^{(j)}$: $B^{(j+1)} = B^{(j)} \oplus A^{(j)}$

8   Build the sparse structure of the matrices $H^{(j+1)}$ and $H_R^{(j+1)}$

After selecting an approximation space $\Pi_{\mathbf{d}}$, an initial basis $B^{(0)}$ belonging to this space must be explicitly defined (line 1). We then construct the sparse structure of the matrices $H^{(0)}$ and $H_R^{(0)}$, that is to say only the indices of the rows and the columns of non-zero elements (line 2). During this step we also build the basis of admissible nodes $B_R^{(0)}$, which is needed for the sparse structure of the matrix $H_R^{(0)}$. The iterative procedure begins by calculating the coefficients of the Hamiltonian matrix (line 3), then the first F eigenpairs (line 4) are computed by an iterative eigensolver. To check the convergence of the algorithm (line 6) we evaluate for all eigenpairs $(E_\ell^{(j)}, \mathbf{X}_\ell^{(j)})$ the scaled residual norms $\|\mathbf{r}_\ell^{(j)}\| = \|H_R^{(j)} \mathbf{X}_\ell^{(j)}\| / E_\ell^{(j)}$. If the maximal value of these norms is lower than the target threshold ε the method has converged. This evaluation requires to compute all coefficients of the rectangle matrix $H_R^{(j)}$ (line 5). If the convergence is not reached we build the new active space by adding directions selected in $B_R^{(j)}$ to the $B^{(j)}$ basis (line 7). Finally, we update the sparse structure of the two matrices thanks to the new basis elements we add (line 8). According to (12) at the convergence of the algorithm the eigenpairs computed are also the eigenpairs of the Hamiltonian discretized in $B^{(j)} \cup B_R^{(j)}$ and not only in $B^{(j)}$.

The main parts of the algorithm are presented in the following subsections. First, we describe how to define the approximation space we work in. Then we explain how we build and update the sparse structure of $H^{(j)}$ and $H_R^{(j)}$. Finally, the last part of this section focuses on how the new directions we add to $B^{(j)}$ are selected.

3.2 Approximation space definition

In order to compute the eigenvalues of the Hamiltonian operator (3), we first define the approximation space in which the operator will be discretized.

We consider the product space $\Pi_{\mathbf{d}}$ defined in (5). The maximal degree of Hermite d_i in each direction i can be determined in different ways. The first idea is to set a constant value N_{max} in each dimension. In this case each direction is discretized by the same number of functions without taking into account harmonic frequencies. Thus, for small harmonic frequencies the highest state has a lower energy than for large ones. To overcome this problem we propose a second approach based on an energetic criterion to reach the same 1-D harmonic energy in each direction. Let E_{max} be a given maximal energy above the ground state for the 1-D Hermite functions. Then the maximal degree in the direction i is

$$d_i = 1 + \left\lfloor \frac{E_{max}}{\omega_i} \right\rfloor. \tag{13}$$

This last constraint decreases drastically the number of elements in the approximation space. We can reduce its size even more by introducing a pruning condition. There exist many pruning conditions (see Ref. [20, 25, 3, 13]). Let P_b be the pruning space in $\Pi_{\mathbf{d}}$ defined by

$$P_b = \{\phi_{\mathbf{n}}^0 \in \Pi_{\mathbf{d}} \text{ such that } g(\mathbf{n}) \leq b\}. \tag{14}$$

Usual pruning functions write $g(\mathbf{n}) = \sum_{i=1}^{D} \alpha_i n_i$ but they can be more complex such as vectorial or non-linear conditions. Typically $\alpha_i = 1$ corresponds to the binomial pruning, denoted BP(b), and $\alpha_i = \omega_i$ is the energetic pruning.

3.3 Matrix Operations

The algorithm is based on the following operations on matrices: the construction and update of their structure, and the calculation of their coefficients. In this subsection we describe these three steps.

3.3.1 Sparse structure definition

Before describing the different steps of the matrix building algorithm we recall some definitions on sparse matrices. Due to the choice of the basis elements, the discretized Hamiltonian operator is a sparse matrix. Therefore we only store non-zero elements of H and H_R to optimize the storage. We define by sparse structure the rows and the columns of the non-zero elements.

The structure of the sparse symmetric matrix H is represented by its graph $G(V, \mathcal{E})$, characterized by a set of nodes V and their connecting edges $\mathcal{E}$. A node of V stands for a row or a column of H (i.e., an element of the basis B). There is an edge between two nodes $(i, j) \in V \times V$ if and only if the coefficient $H_{i,j}$ is not zero.

We denote by $G_{H_R}(W, V, \mathcal{E}_{\mathcal{R}})$ the bipartite graph of the rectangular matrix H_R. The node sets W and V represent the rows and columns of H_R, respectively, and are connected by the set of edges $\mathcal{E}_{\mathcal{R}}$. The condition $(H_R)_{i,j} \neq 0$ defines an edge between $i \in W$ and $j \in V$, and $G_{H_R}(W, V, \mathcal{E}_{\mathcal{R}})$ gives the structure of H_R.

Thanks to their graphs we store the matrices H and H_R in the Compressed Sparse Row (CSR) format.

3.3.2   Sparse structure construction

Starting from the basis $B^{(0)}$ the second step of our algorithm (line 2) is to construct the sparse structures of the matrices H and H_R. As the matrix H is symmetric we only consider its lower triangular part. Moreover, this step also constructs the set of the admissible nodes $B_R^{(0)}$ associated to the initial basis $B^{(0)}$. This means that for each element $\mathbf{n} = (n_1, \ldots, n_D)$ of $B^{(0)}$ we should determine all nodes $\mathbf{m}$ in $\Pi_{\mathbf{d}}$ connected to it through the anharmonic part of the PES operator. As the PES is a sum of products of monomials we proceed as follows.

Firstly, let $C(\mathbf{n}, \mathbf{s}) = \{\mathbf{m} \text{ such that } \langle \Phi_{\mathbf{n}}(\mathbf{q}) | \mathbf{q}^{\mathbf{s}} \Phi_{\mathbf{m}}(\mathbf{q}) \rangle \neq 0\}$ be introduced as the set of all nodes connected to $\mathbf{n}$ through the monomial $\mathbf{q}^{\mathbf{s}}$, with $\mathbf{s} = (s_1, \ldots, s_D)$. Since the basis functions are products of 1-D functions we only have to check if the 1-D integral $\langle \phi_{n_i}(q) | q^{s_i} \phi_{m_i}(q) \rangle$ does not vanish for all $i = 1, \ldots, D$. For 1-D Hermite functions these integrals have analytic expressions (see Prop. 2 in Annex A) and due to the approximation space we only keep indices lower than d_i. Then, to construct all multi-indices connected to $\mathbf{n} = (n_1 \ldots, n_D)$ with a monomial of the PES, we calculate for each dimension all the integers connected to n_i by using Prop. 1 of Annex A. The selected multi-indices are reconstructed and sorted by their key. The key of a node $\mathbf{n} = (n_1, \ldots, n_D)$ is defined by

$$key(\mathbf{n}) = n_1 + n_2(d_1 + 1) + \ldots + n_D(d_1 + 1) \ldots (d_{D-1} + 1),$$

where d_i is the maximal Hermite degree in each direction. If needed, a pruning condition can be applied before building the key to decrease the number of connected nodes. It is easy to see that the number of elements in $C(\mathbf{n}, \mathbf{s})$ is bounded by $D\left(\|\mathbf{s}\|_\infty + 1\right)$, where $\|\mathbf{s}\|_\infty = \max_{i=1,\ldots,D} s_i$ and the cost to obtain them is $O(D\|\mathbf{s}\|_\infty \ln(D\|\mathbf{s}\|_\infty))$.

Finally, the set of all nodes connected to $\mathbf{n}$ is obtained by iterating on all monomial elements $\mathbf{q}^{\mathbf{s}}$ of the PES

$$C(\mathbf{n}) = \bigcup_{\mathbf{s} \in Deg(\text{PES})} C(\mathbf{n}, \mathbf{s}), \tag{15}$$

where $Deg(\text{PES})$ is the set of the monomial degrees $\mathbf{s}$ in the PES such that $\|\mathbf{s}\|_1 \geq 3$. The total number of nodes connected to $\mathbf{n}$ is bounded by $D\left(S + 1\right) N_{\text{PES}}$, where S is the maximal degree of the PES and N_{PES} is the number of anharmonic terms in the PES. This number is also the bound of all non-zero elements on the row associated with the node $\mathbf{n}$ of the Hamiltonian matrix in the full space $\Pi_{\mathbf{d}}$. The complexity of this step is bounded by $O(DS \ln(DS) N_{\text{PES}})$.

Thanks to the space decomposition (7) the set of all nodes connected to $\mathbf{n}$ at the beginning of the iterative process splits in two parts as follows:

$$C(\mathbf{n}) = \mathcal{E}(\mathbf{n}) \oplus \mathcal{E}_{\mathcal{R}}(\mathbf{n}),$$

where $\mathcal{E}(\mathbf{n})$ contains the nodes inside $B^{(0)}$ and $\mathcal{E}_{\mathcal{R}}(\mathbf{n})$ the nodes outside $B^{(0)}$ sorted by their key. Since the elements of $\mathcal{E}_{\mathcal{R}}(\mathbf{n})$ belongs to the basis set of admissible nodes $B_R^{(0)}$ we have

$$B_R^{(0)} = \bigcup_{\mathbf{n} \in B^{(0)}} \mathcal{E}_{\mathcal{R}}(\mathbf{n}). \tag{16}$$

As the nodes of $\mathcal{E}_{\mathcal{R}}(\mathbf{n})$ are already sorted by their key the construction of $B_R^{(0)}$ is just a merge of several arrays. The cost is linear with respect to the maximal length of these arrays.

In this approach, the sparse structure built on the partition space $B^{(0)} \times B_R^{(0)}$ corresponds to the sparse structure of the transposed of H_R. Therefore, the last step consists in transposing this structure to get the final pattern of $H_R^{(0)} = {B^{(0)}}^T \mathcal{H} B_R^{(0)}$.

3.3.3 Sparse structure update

Starting at iteration j from a set of nodes $A^{(j)}$ selected in $B_R^{(j)}$, the purpose is to build the new sparse structure $G(B^{(j+1)}, \mathcal{E}^{(j+1)})$ (respectively, $G_{H_R}(B_R^{(j+1)}, B^{(j+1)}, \mathcal{E}_{\mathcal{R}}^{(j+1)})$) of the matrix $H^{(j+1)}$ (resp. $H_R^{(j+1)}$) without reconstructing them from scratch. To do that, we only compute the structure for the new nodes and merge it with the one built at the previous iteration. We proceed in three steps:

Step 1 Add nodes of $A^{(j)}$ in $B^{(j)}$ to obtain the new basis set $B^{(j+1)} = B^{(j)} \cup A^{(j)}$. Fill $\mathcal{E}^{(j+1)}$ with the old sparse structure $\mathcal{E}^{(j)}$ and $\mathcal{E}_{\mathcal{R}}^{(j+1)}$ with the old structure $\mathcal{E}_{\mathcal{R}}^{(j)}$ without the nodes in $A^{(j)}$.

Step 2 For all elements $\mathbf{a}$ in $A^{(j)}$ we compute the set of the nodes connected with $\mathbf{a}$ defined by: $C(\mathbf{a}) = \mathcal{E}(\mathbf{a}) \oplus \mathcal{E}_{\mathcal{R}}(\mathbf{a})$, where $\mathcal{E}(\mathbf{a})$ are the nodes in $B^{(j+1)}$ and $\mathcal{E}_{\mathcal{R}}(\mathbf{a})$ the nodes outside $B^{(j+1)}$. We complete the structure $\mathcal{E}^{(j+1)}$ with $\bigcup\limits_{\mathbf{a} \in A^{(j)}} \mathcal{E}(\mathbf{a})$ and the new set of admissible nodes is $B_R^{(j+1)} = (B_R^{(j)} \backslash A^{(j)}) \cup \mathcal{E}_{\mathcal{R}}(A^{(j)})$, with $\mathcal{E}_{\mathcal{R}}(A^{(j)}) = \bigcup\limits_{\mathbf{a} \in A^{(j)}} \mathcal{E}_{\mathcal{R}}(\mathbf{a})$.

Step 3 Finally, we transpose $\mathcal{E}_{\mathcal{R}}(A^{(j)})$ and add it to $\mathcal{E}_{\mathcal{R}}^{(j+1)}$ to obtain the final sparse structure of $H_R^{(j+1)}$.

The most costly part of this update is step 2 but it can easily be parallelized.

3.3.4 Matrix coefficient evaluation

Let consider two basis functions $\Phi_{\mathbf{n}}$ and $\Phi_{\mathbf{m}}$, the matrix element $\langle \Phi_{\mathbf{n}}(q) | \mathcal{H} \Phi_{\mathbf{n}}(q) \rangle$ is

$$\langle \Phi_{\mathbf{n}}(q) | \mathcal{H} \Phi_{\mathbf{m}}(q) \rangle = \sum_{i=1}^{D} \omega_i \left(\frac{1}{2} + n_i \right) \delta_{\mathbf{n},\mathbf{m}} + \sum_{\|\mathbf{s}\|_1=3}^{S} K_{\mathbf{s}} \prod_{i=1}^{D} \langle \phi_{n_i}(q_i) | q_i^{s_i} \phi_{m_i}(q_i) \rangle.$$

All the 1-D integrals $\mathcal{M}(s, n, m) = \langle \phi_n(x) | x^s \phi_m(x) \rangle$ have an analytic expression (see Annex A). They are pre-computed once and stored in the tensor moment $\mathcal{M}$. Then, the matrix coefficients are given by

$$\langle \Phi_{\mathbf{n}}(q) | \mathcal{H} \Phi_{\mathbf{m}}(q) \rangle = \sum_{i=1}^{D} \omega_i \left(\frac{1}{2} + n_i \right) \delta_{\mathbf{n},\mathbf{m}} + \sum_{\|\mathbf{s}\|_1=3}^{S} K_{\mathbf{s}} \prod_{i=1}^{D} \mathcal{M}(s_i, n_i, m_i).$$

It is clear that the complexity to build one matrix element is $O(D \, N_{\text{PES}})$ then the total complexity to build the Hamiltonian matrix is $O(D \, N_{\text{PES}} \, Z_H)$ and $O(D \, N_{\text{PES}} \, Z_{H_R})$ for H_R, where Z_H and Z_{H_R} are the number of non-zero elements of the matrices H and H_R, respectively. However, this construction is highly parallel and its cost is negligible compared to the construction of the sparse structure of the matrix.

3.4 Basis expansion strategies

The cost of A-VCI is mainly related to the total matrix operation complexity, which depends on the size of the basis involved in this adaptive algorithm. The number of elements in $B^{(j)}$ and $B_R^{(j)}$ at iteration j and the number of nodes connected to them increase with the dimension D, the number of terms of the PES and the required precision. Therefore, the way to select the most relevant nodes to expand the active space is crucial in terms of accuracy and performance. The less functions at each iteration we add, the smaller the final basis and the related memory footprint will be, but the number of iterations can increase and this can penalize the computational time to reach the convergence. In fact, there is a trade-off between the memory requirement and the convergence speed of the algorithm. Moreover, developing an automatic process to select such nodes

is important for the efficiency of the algorithm. In this section we discuss different approaches to enlarge the A-VCI basis (line 7 in Algorithm 1) with respect to the previous constraints.

We remind that the convergence of A-VCI is checked by comparing the norms of the scaled residual vectors with a given threshold denoted ε. Let $K^{(j)} = \{\ell \in \{0, \dots, F-1\}$ such that $\|\mathbf{r}_\ell^{(j)}\| > \varepsilon\}$ the list of the $k^{(j)}$ non-converged relative residues at iteration j. In all cases the expansion strategies will be based on the elements of $K^{(j)}$. Consider a vector $\mathbf{v}$ of size m_R, we denote by $M_\eta(\mathbf{v}) = \{i$ such that $|v_i| > \eta\}$ the space of admissible components of size m_η and we define the generalized average of $\mathbf{v}$ with respect to this space by

$$mean(\mathbf{v}, p, \eta) := \sqrt[p]{\frac{1}{m_\eta} \sum_{i=1}^{m_R} |v_i|^p \, \mathbf{1}_{M_\eta(\mathbf{v})}(i)}, \tag{17}$$

where $\mathbf{1}_{M_\eta(\mathbf{v})}$ is the indicator function of space $M_\eta(\mathbf{v})$ (i.e., 1 when i is in $M_\eta(\mathbf{v})$ and 0 otherwise). We also introduce $N_J(\mathbf{v})$ the space of the indices of the J largest components of $\mathbf{v}$.

3.4.1 Component-wise strategies

Garnier *et al.* [23] introduced a component-wise procedure on each non-converged relative residue to expand the $B^{(j)}$ basis based on the usual mean (i.e., $p = 1$ in (17)). We denote this strategy CW($p=1$). In this approach, the set of nodes added at iteration j is

$$A^{(j)} = \bigcup_{\ell \in K^{(j)}} \left\{ i \text{ such that } |(\mathbf{r}_\ell^{(j)})_i| > mean(\mathbf{r}_\ell^{(j)}, 1, \varepsilon/\sqrt{m_R}) \right\}. \tag{18}$$

Such procedure leads to introduce a lot of components at each step. Then, both $B^{(j)}$ and $B_R^{(j)}$ grow quickly during the iterations. Another way, denoted CW2, consists in selecting at iteration j the functions of $B_R^{(j)}$ corresponding to the $2(j+1)$ largest components of each non-converged residual vector. Using CW2 the set of nodes enlarging $B^{(j)}$ is

$$A^{(j)} = \bigcup_{\ell \in K^{(j)}} N_{2(j+1)}(\mathbf{r}_\ell^{(j)}), \tag{19}$$

and the maximal number of components added at iteration j is $2(j+1)k^{(j)}$. Therefore the expansion of the $B^{(j)}$ basis is moderate compared to the CW(1) method. Another advantage of this strategy is that the convergence is adaptive since the number of added nodes increases with respect to the iteration number. At the beginning of the iterative process the error is large in any case, so it is not necessary to expand $B^{(j)}$ in every direction. It seems more sensible to add as many elements as possible when we get close to the convergence in order to accelerate it and refine it. At the end, the basis contains the most relevant nodes to compute the eigenvalues at the given accuracy. On the other hand, to select the largest components we sort each residual vector leading to the global cost $O(k^{(j)} m_R \ln(m_R))$ compared to the linear cost $O(k^{(j)} m_R)$ for the former CW(1) strategy.

3.4.2 Collective component-wise strategies

The main drawback of the component-wise strategies is that the basis expansion procedures take into account each residual vector independently. For CW(1) the value of $mean(\mathbf{r}_\ell^{(j)}, 1, \varepsilon/\sqrt{m_R})$ depends on the considered vector $\mathbf{r}_\ell^{(j)}$, and in CW2 the $2(j+1)$ largest components do not necessarily correspond to the same level of approximation for two different residues. In fact, it is difficult to say if a relevant component for a given residual vector is useful to decrease the residual components of other eigeinvalues. Moreover the component-wise selection can be redundant since the same nodes are possibly chosen from different residual vectors. To overcome these issues we introduce at iteration j a global residual vector $\mathbf{R_A}^{(j)}$ such that each component is

$$\mathbf{R_A}_i^{(j)} = \frac{1}{k^{(j)}} \sum_{\ell \in K^{(j)}} |(\mathbf{r}_l^{(j)})_i|, \; i = 1, \dots, m_R.$$

The $\mathbf{R_A}^{(j)}$ vector allows to measure the global contribution to the residue of all individual vectors belonging to $K^{(j)}$. The same strategies used for the component-wise selection can be applied to the $\mathbf{R_A}^{(j)}$ vector:

CCW(p) we choose the functions thanks to a general mean criterion on $\mathbf{R_A}^{(j)}$ components. The nodes to add to the A-VCI space are defined by

$$A^{(j)} = \{i \text{ such that } \mathbf{R_A}_i^{(j)} > mean(\mathbf{R_A}^{(j)}, p, \varepsilon/\sqrt{m_R})\}. \tag{20}$$

CCW2 we select the nodes with the $2(j+1)$ maximal components of $\mathbf{R_A}^{(j)}$ such that

$$A^{(j)} = N_{2(j+1)}(\mathbf{R_A}^{(j)}). \tag{21}$$

The number of directions added by CW(p) or CCW(p) increases rapidly with respect to the required accuracy. This leads to large CPU times in the matrix operations. In such cases, it is crucial to make the computational cost as affordable as possible. One possibility is to adapt the mean by increasing or by decreasing p in (17) instead of sorting the scaled vector to find the largest components (like in CC2 or CCW2), in order to reach a target number of nodes to add. The goal here is to bound the time spent in the update step. This approach will increase the number of iterations but the size of the final space is smaller than those obtained from other strategies. Let N_T the target number of basis vectors (nodes) we want to add, δ the tolerance criterion we accept on this number, then we adjust p such that $|size(A^{(j)})/N_T - 1| \leq \delta$, where $size(A^{(j)})$ is the number of elements of $A^{(j)}$. When the value of p is changed we reconstruct $A^{(j)}$ by (20). We call this strategy aCCW(N_T, δ) for automatic collective component-wise selection. The complexity of this procedure is only linear with respect to m_R compared to $m_R \ln(m_R)$ with sort operation.

4 Results and discussion

The presented method has been tested on the 6-atom acetonitrile molecule, CH_3CN, and on a more challenging system of 7 atoms: the ethylene oxide molecule, C_2H_4O. Two implementations of A-VCI have been used: a Python/C prototype and a full C++ version. In the Python/C software all the basis enlargement methods are implemented whereas the C++ driver only provides the collective strategies. These codes use the iterative ARPACK solver [31] to compute the lowest F eigenvalues of the discretized Hamiltonian matrix at each iteration of A-VCI. The initial guess of the eigensolver is a random vector ($IDO = 0$). Moreover, we set the number of Arnoldi vectors to $NCV = 2F + 1$ and we use the ARPACK default convergence criteria based on the machine precision. Finally, both implementations of A-VCI take advantage of the OpenMP fork-join paradigm for parallelization.

The results are obtained on two platforms:

(i) A 24-core Haswell Intel Xeon E5-2680 processors running at 2.8 GHz with 128 GB of shared memory. The Intel compiler (2016 update 3) with the following options: `-mkl=parallel -march=native -axCORE-AVX2,CORE-AVX-I,AVX -qopenmp` is used. We refer to it as *plafrim* in the sequel.

(ii) A 12-core Nehalem Intel Xeon L5640 nodes running at 2.27 GHz with 24 GB of shared memory. Python 3.5.1 and the open-source GNU 5.1.0 compiler with the options `-O3 -mtune=native -fopenmp` were used. We refer to it as *pyrene* in the sequel.

4.1 Acetonitrile molecule, CH_3CN

The convergence of the A-VCI algorithm depends on how we choose the approximation space, the threshold and the basis expansion method. What would be the most efficient way, for a required precision, to set all these parameters, that is to say to obtain the smallest basis or the lowest CPU time and memory consumption

? In this part, we study this problem in the case of the well-known acetonitrile molecule. The CH_3CN force constants come from Ref. [1] and are based on the quartic PES originally developed by Bégué *et al.* [7] The PES is composed of 299 anharmonic terms (108 cubic and 199 quartic terms). We consider the 12 harmonic frequencies sorted by ascending order

$$\omega_1 = \omega_2 = 361, \omega_3 = 920,$$
$$\omega_4 = \omega_5 = 1\,061, \omega_6 = 1\,413,$$
$$\omega_7 = \omega_8 = 1\,487, \omega_9 = 2\,297,$$
$$\omega_{10} = 3\,065, \omega_{11} = \omega_{12} = 3\,149\,(\text{cm}^{-1}).$$

First, we compare several approaches to expand the $B^{(j)}$ basis at each iteration j, then we show how the threshold and the size of the approximation space $\Pi_{\mathbf{d}}$ influence the accuracy of the eigenvalues. Finally, we compute the frequencies up to $3\,000\,\text{cm}^{-1}$ and compare them with a reference calculation [1].

4.1.1 Influence of different strategies to expand the A-VCI space

Here we compare the different strategies to expand the A-VCI space presented in Subsection 3.4 when we compute the first 121 eigenvalues as in Ref. [23]. We remind that CW(1) and CW2 are component-wise whereas CCW(p) and CCW2 are based on the collective residue vector $\mathbf{R_A}^{(j)}$. In this section, the calculations are done with the Python/C driver on *pyrene* and we consider the same degree of approximation in each direction fixed to 30 for the approximation space $\Pi_{\mathbf{d}}$, the threshold is $\varepsilon = 7.5 \times 10^{-3}$. The initial basis $B^{(0)}$ is given by the first 121 functions of $\Pi_{\mathbf{d}}$ (with $\mathbf{d} = [7,6,2,2,2,1,1,1,1,0,0,0]$) sorted by ascending energies in the harmonic approximation.

Table 1 reports the size and the number of non-zero elements in the matrices H and H_R obtained for different strategies, with the corresponding CPU time. The absolute error denoted *Err* is computed with respect to a calculation of Avila and Carrington[1] with a basis of 743 103 elements. The first two rows of Table 1 provide the results of the component-wise methods while the others are related to collective strategies. The first line corresponds to the A-VCI approach used in Ref. [23], leading to the largest basis and CPU time but also to the best accuracy. The basis grows smaller when expanding it with the nodes which have the largest residual components, with both collective (line 7) and individual (line 2) strategies. In these cases, the error can be large (more than $1\,\text{cm}^{-1}$) and the CPU time is higher than with the CCW methods. We observe that the most efficient strategies are CCW(2) and CCW(3), with equivalent performances. From $p = 4$ the cost of the additional iterations overcomes the gain due to the basis reduction, so rising p from this point becomes no longer interesting. The matrix storage cost for CCW strategies is low and then building the full sparse matrix to perform efficient matrix-vector product is a good option. For instance, the number of non-zero elements divided by the total number of elements (sparsity of the matrix) is 0.45% for $H^{(j)}$ matrix and 3 per million for $H_R^{(j)}$ matrix for the CCW(2) strategy. Table 1 shows that a trade-off between the basis size and the number of elements added at each iteration must be considered. Thanks to the exponent p of the generalized mean (17) we can balance this trade-off. The higher its value is, the less basis functions are added, resulting in more iterations and a smaller final basis.

4.1.2 Convergence study of the A-VCI algorithm

In order to present the convergence properties of the A-VCI algorithm with respect to the threshold ε and the approximation space $\Pi_{\mathbf{d}}$, we search the 239 lowest eigenvalues of CH_3CN. The corresponding frequency range is around 0–$3\,000\,\text{cm}^{-1}$ and has been studied in many papers [1, 3, 13]. We perform a very accurate A-VCI calculation with a small threshold $\varepsilon_{ref} = 2.5 \times 10^{-4}$ and the CCW(2) strategy to obtain a reference spectrum. The maximal Hermite degrees per direction of the related discretized space $\Pi_{\mathbf{d}}$ corresponding to a maximal energy of $15\,000\,\text{cm}^{-1}$ are

$$\mathbf{d} = [42, 42, 17, 15, 15, 11, 11, 11, 7, 5, 5, 5].$$

Table 1: Convergence of A-VCI for CH_3CN with different basis expansion strategies. The used parameters are $F = 121$ and $\varepsilon = 7.5 \times 10^{-3}$. The approximation space $\Pi_{\mathbf{d}}$ is defined by $d_i = 30$, $i = 1 \dots 6$. The absolute error Err is evaluated with respect to Ref. [1]. The cost of the algorithm is given by the number of iterations, the sizes of the final spaces and the computational time. The number of non-zero elements (NNZ) in $H^{(j)}$ and $H_R^{(j)}$ at the convergence is also provided.

Expand strategy	Number of iterations	$B^{(j)}$ size	$B_R^{(j)}$ size	$H^{(j)}$ NNZ	$H_R^{(j)}$ NNZ	Err (cm^{-1})	Time (s)
CW(1)	6	86 238	5 421 360	13 166 360	51 581 366	0.305	5 637
CW2	19	17 839	1 819 345	1 593 955	10 573 541	1.333	1 230
CCW(1)	6	26 206	2 360 202	2 835 020	15 430 825	0.766	711
CCW(2)	7	24 158	2 168 898	2 607 190	14 201 050	0.997	667
CCW(3)	8	22 477	2 064 762	2 357 655	13 212 455	0.998	669
CCW(4)	10	22 432	2 082 831	2 339 942	13 197 787	0.803	723
CCW2	17	17 793	1 723 782	1 744 565	10 385 001	0.999	902

Moreover, the initial basis $B^{(0)}$ is composed of the first 239 elements of $\Pi_{\mathbf{d}}$, which largest degrees are given by $[8, 8, 3, 2, 2, 2, 2, 2, 1, 0, 0, 0]$. From now we will use these functions as the A-VCI starting space of CH_3CN.

For this computation, the C++ driver on plafrim is used. The algorithm reaches the convergence in 11 iterations producing a basis of 2 488 511 elements. We denote the associated eigenvalues by E_i^{ref}, for $i = 0 \dots F - 1$. The corresponding frequencies and their assignments are provided in the Supplementary Material. The maximal degrees of the final basis are $[18, 18, 17, 12, 10, 11, 9, 7, 7, 5, 5, 5]$. This final set is embedded in the binomial space BP(23). In the following sections, we compare our results to this reference calculation.

Influence of the threshold on the convergence: First we present the influence of the threshold on the accuracy of the eigenvalues and the basis size. We consider the same parameters as our reference computation described above. Let $Err(E)$ denote the global error on all eigenvalues

$$Err(E) = \max_{i \in \{0, F-1\}} |E_i^{\varepsilon} - E_i^{ref}| \tag{22}$$

where E_i^{ε} is the eigenvalue obtained with the threshold ε.

Figure 1 shows the evolution of the maximal error on the eigenvalues according to the size of the final basis, and the evolution of the latter with the threshold, on a log-log scale. The maximal error nearly evolves as a decreasing power function of the size of the final basis. We deduce that the maximal error quickly and uniformly diminishes when the size of the final basis increases. Moreover, the behavior of the size of the basis with respect to the threshold parameter ε is similar, making the threshold a good parameter to control the growth of the basis. For this molecule, this figure shows that it is possible to determine an approximate basis size from a required accuracy on the eigenvalues, and deduce from this size a maximal threshold value needed to reach this precision. For example, an error below 1 cm^{-1} at least requires a basis size of order 5×10^4, for which the threshold needs to be no more than 7×10^{-3}.

Influence of the truncation parameter of the research space: By using the approximation parameter E_{max} introduced in Section 3.2, we reduce the size of the approximation space and of the research space to find new directions.

On Table 2 are reported the maximal Hermite degrees per coordinate corresponding to E_{max} ranging from 8 500 to 15 000 cm^{-1}. The maximal error of the first 239 eigenvalues with respect to the threshold ε is presented in Figure 2 for different values of E_{max}. Due to the log-log scales the curves for E_{max} between 9 000 cm^{-1} and 12 000 cm^{-1} are not visible and give an error above 1 cm^{-1}. Except for the green curve $E_{max} = 14 000$ cm^{-1} all
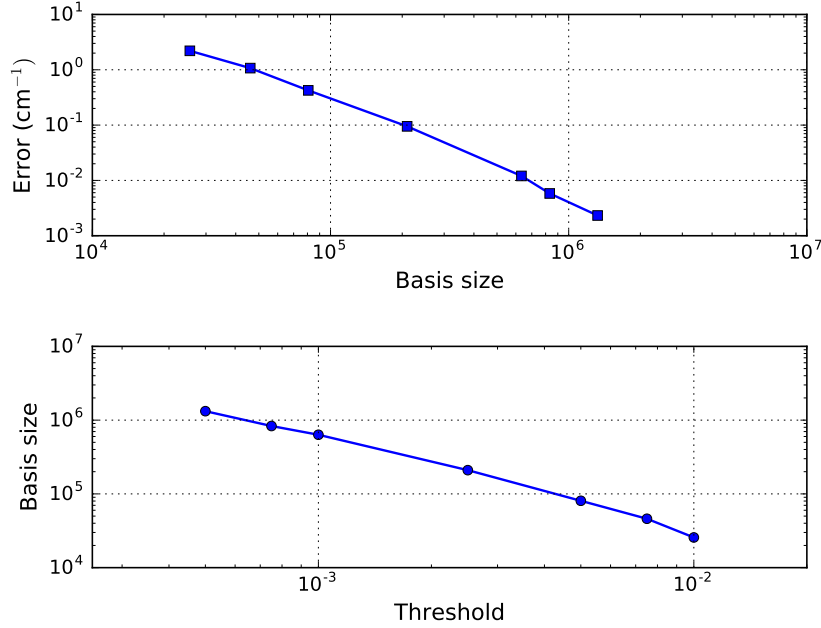
Figure 1: Influence of the threshold ($\log_{10}$ scale) on the final basis size ($\log_{10}$ scale) and of the basis size on the maximum error ($\log_{10}$ scale) for $E_{max} = 15\,000\,\text{cm}^{-1}$.

Table 2: Maximal Hermite degrees of the product space $\Pi_{\mathbf{d}}$ corresponding to a given energy above the ground state E_{max}.

E_{max} (cm^{-1})	$\mathbf{d}$ in $\Pi_{\mathbf{d}}$
8 500	$[24, 24, 10, 9, 9, 7, 6, 6, 4, 3, 3, 3]$
10 000	$[28, 28, 11, 10, 10, 8, 7, 7, 5, 4, 4, 4]$
11 000	$[31, 31, 12, 11, 11, 8, 8, 8, 5, 4, 4, 4]$
12 000	$[34, 34, 14, 12, 12, 9, 9, 9, 6, 4, 4, 4]$
12 500	$[35, 35, 14, 12, 12, 9, 9, 9, 6, 5, 4, 4]$
13 000	$[37, 37, 15, 13, 13, 10, 9, 9, 6, 5, 5, 5]$
14 000	$[39, 39, 16, 14, 14, 10, 10, 10, 7, 5, 5, 5]$
15 000	$[42, 42, 17, 15, 15, 11, 11, 11, 7, 5, 5, 5]$

the curves present a plateau that means that even if we lower the threshold we cannot improve the accuracy on the eigenvalues. This plateau decreases with the size of the approximation space. Moreover, the same holds the other way around: for ε higher than 2.5×10^{-3} there is no need to increase E_{max} above $12\,500$ cm^{-1}. Finally, the green curve does not have a plateau because for all thresholds the eigenvectors and the Hamiltonian are well described in the space $\Pi_{\mathbf{d}}$.

4.1.3   New results

Our computations highlight a significant gap with the results in Ref. [1]. For instance, the 150th frequency is estimated at $2\,653.053$ cm^{-1} whereas our reference calculation locates it at $2\,651.593$ cm^{-1}. The difference between the values calculated in the P_{24} and P_{27} spaces is around 2 cm^{-1} whereas it stays under 1 cm^{-1} for all other frequencies (see Table I. in Ref. [1]). It seems the convergence of this particular eigenvalue is hard to reach.
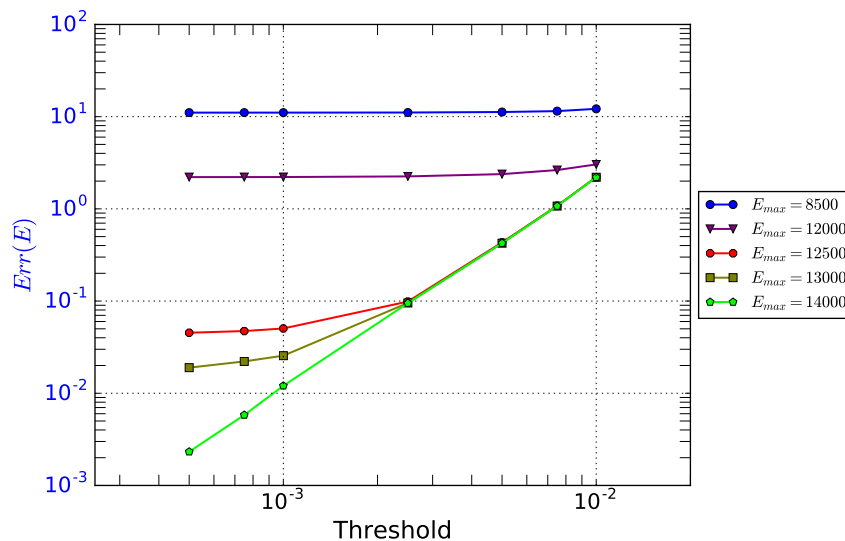
Figure 2: Influence of the threshold ($\log_{10}$ scale) on the eigenvalues error ($\log_{10}$ scale) for different approximation spaces defined by E_{max}.

Moreover, 35 of the 195 first eigenvalues have a difference above $0.1\,\mathrm{cm}^{-1}$ compared to our reference calculation. Finally, A-VCI calculates 238 frequencies under $3\,000\,\mathrm{cm}^{-1}$ whereas only 202 are reported in Ref. [1]. We see that A-VCI is able to retrieve the last 6 frequencies computed by Avila and Carrington [1] and to catch some energetic levels missing from their paper, represented by frequencies 196–199, 202–228, 230–232 and 235–237.

In order to explain these discrepancies between eigenvalues in the high frequency range of the spectrum, and to validate that our algorithm has good convergence properties, the direct product search basis $\Pi_{\mathbf{d}}$ has been replaced by a space with the same pruning condition as in Ref. [1], for different values of the convergence criterion b. The use of the A-VCI algorithm in these pruned basis allows us to reach high values of b since the full pruned basis is never constructed. The pruning condition is added in (15) to discard nodes just before constructing the keys. By doing this, we are able to find an optimal basis in the pruned space.

Tab. 3 shows how the A-VCI procedure applied in a pruned basis with increasing values of b brings close to the converged value of this frequency given by our reference calculation: $2\,651.593\,\mathrm{cm}^{-1}$. Here the A-VCI parameters are $F = 239$, $\varepsilon = 5.0 \times 10^{-3}$, CCW(2) and we use the pruning condition of Ref. [1] written in our ordering:

$$n_1 + n_2 + 3n_3 + 3n_4 + 3n_5 + 3n_6 + 4n_7 + 4n_8 + 4n_9 + 3n_{10} + 3n_{11} + 3n_{12} \leq b \tag{23}$$

As shown in Tab. 3, a pruned basis with $b = 41$ seems necessary for the A-VCI algorithm to reach the convergence on the 150th frequency. Solving the eigenvalue problem in the corresponding space P_{41} with either a direct or iterative method is difficult due to its size whereas the A-VCI method succeeds to extract the relevant states in a basis of $82\,099$ elements.

The final A-VCI basis in the product space defined by $E_{max} = 12\,500\,\mathrm{cm}^{-1}$ counts $79\,133$ elements and leads to the same results that using the pruning condition (23) with $b = 41$: compared to our reference calculation the 150th frequency has an error of $0.272\,\mathrm{cm}^{-1}$, and the largest error on all computed eigenvalues is $0.425\,\mathrm{cm}^{-1}$. The full comparison between this latter calculation, the A-VCI reference calculation and Avila and Carrington [1] is provided in the Supplementary Material.

Table 3: Absolute error on the 150th frequency of CH_3CN using the A-VCI procedure in different pruned basis P_b. The pruning condition is given by Eq. (23). The number of computed eigenvalues is $F = 239$ and the threshold is $\varepsilon = 5 \times 10^{-3}$. The error is evaluated with respect to the reference calculation described in Part 4.1.2. The last column corresponds to an A-VCI calculation with the same parameters performed in the product basis $\Pi_{\mathbf{d}}$ defined by $E_{max} = 12\,500\,cm^{-1}$.

Basis	P_{27}	P_{33}	P_{37}	P_{40}	P_{41}	P_{45}	$\Pi_{\mathbf{d}}(12\,500)$
Final basis size	58 122	79 409	79 743	81 594	82 099	82 405	79 133
150th freq. (cm^{-1})	2 653.405	2 652.101	2 651.908	2 651.884	2 651.869	2 651.867	2 651.862
Error (cm^{-1})	1.815	0.511	0.318	0.294	0.279	0.277	0.272

4.2 Ethylene oxide molecule, C_2H_4O

We have shown that A-VCI method accurately computes the spectrum of CH_3CN in a large frequency range of interest. Let us now consider a more difficult problem with the ethylene oxide molecule. To study the vibrational levels of this 7-atom system we use the PES described in Bégué *et al.* [8]. The anharmonic part of this potential is composed of 180 cubic terms and 445 quartic terms. The 15 harmonic frequencies are sorted by ascending order:

$$
\begin{aligned}
\omega_1 &= 815.515, \omega_2 = 850.180, \omega_3 = 899.564, \\
\omega_4 &= 1\,052.231, \omega_5 = 1\,156.802, \omega_6 = 1\,157.906, \\
\omega_7 &= 1\,174.993, \omega_8 = 1\,176.045, \omega_9 = 1\,300.108, \\
\omega_{10} &= 1\,512.350, \omega_{11} = 1\,549.074, \omega_{12} = 3\,109.459, \\
\omega_{13} &= 3\,117.878, \omega_{14} = 3\,196.560, \omega_{15} = 3\,211.265\,(cm^{-1}).
\end{aligned}
$$

In this section, we consider the automatic collective component wise (aCCW) strategy to expand the basis. All computations were run on *plafrim*.

4.2.1 The first 50 eigenvalues

We focus here on the 50 lowest energetic levels of C_2H_4O. The corresponding eigenvalues were computed by Brown and Carrington [13] in a basis of 2 955 289 functions thanks to an adaptive procedure called "O Expand". Using this strategy they proved to be more efficient than with a binomial basis BP(11) of 7 726 160 elements. As far as we know, this "O Expand" computation is the most accurate one to date on the first 50 eigenvalues of C_2H_4O. In this part, the absolute error on the A-VCI eigenvalues is denoted Err and evaluated with respect to this reference calculation.

The A-VCI initial basis $B^{(0)}$ is made of the first 50 harmonic configurations of the product space defined by the Hermite degrees $\mathbf{d} = [6, 5, 5, 4, 4, 4, 4, 4, 3, 3, 3, 1, 1, 1, 1]$. The approximation spaces $\Pi_{\mathbf{d}}(E_{max})$ and BP(b) with different values of ε are considered. Table 4 shows how the choice of these parameters influences the accuracy on eigenvalues, the basis size and the computational time. The first three lines refer to the aCCW automatic selection of A-VCI to limit the number of nodes added at each iteration. The two product spaces $\Pi_{\mathbf{d}}$ defined for a maximal energy of $10\,000\,cm^{-1}$ and $14\,000\,cm^{-1}$ lead to maximal Hermite degree $\mathbf{d} = [13, 12, 12, 10, 9, 9, 9, 9, 8, 7, 7, 4, 4, 4, 4]$ and $\mathbf{d} = [18, 17, 16, 14, 13, 13, 12, 12, 11, 10, 10, 5, 5, 5, 5]$, respectively. The last two lines of Table 4 present a pruning approach in the binomial spaces BP(11) and BP(16) with two different ways to expand the basis.

In the considered computations a threshold of 5×10^{-3} leads to an error of $0.4\,cm^{-1}$ and final bases almost ten times smaller than the "O Expand" set of Ref. [13], with less than 350 000 functions. For this level of precision using the binomial basis BP(11) with the CCW(5) strategy (line 4) is faster than the aCCW expansion method applied in the product space $\Pi_{\mathbf{d}}(10\,000)$ (line 1). This is no longer true when we decrease the threshold to

2.5×10^{-3} to reach an accuracy around $0.04\,\mathrm{cm}^{-1}$. In this case, the $\Pi_{\mathbf{d}}$ spaces give smaller basis and CPU times than the BP(b) spaces. A threshold of 2.5×10^{-3} with $E_{max} = 10\,000\,\mathrm{cm}^{-1}$ is enough to obtain an error below $0.1\,\mathrm{cm}^{-1}$. A larger product space corresponding to $E_{max} = 14\,000\,\mathrm{cm}^{-1}$ leads to an accuracy of $0.047\,\mathrm{cm}^{-1}$ in a basis twice smaller than in Ref. [13] (see line 3). In this later calculation, the maximal Hermite degrees per direction of the the final basis are $[11, 9, 8, 8, 8, 10, 7, 7, 9, 6, 7, 5, 5, 5, 5]$ and the final basis is embedded in the binomial space BP(17).

Table 4: Convergence of A-VCI for the 50 lowest eigenvalues of C_2H_4O. Several approximation spaces and thresholds are used with the aCCW and CCW(5) basis expansion strategies. The corresponding basis sizes and number of iterations are reported. The reference calculation for the absolute error is provided in Brown and Carrington [13].

Space	Threshold	Iterations	Err	m	m_R	Strategy	Time (s)
$\Pi_{\mathbf{d}}(10\,000)$	5.0×10^{-3}	11	0.42	325 177	32 469 708	aCCW(30 000, 0.6)	4 601
$\Pi_{\mathbf{d}}(10\,000)$	2.5×10^{-3}	23	0.080	1 240 302	92 205 491	aCCW(50 000, 0.6)	47 366
$\Pi_{\mathbf{d}}(14\,000)$	2.5×10^{-3}	25	0.047	1 362 866	107 840 100	aCCW(50 000, 0.6)	57 529
BP(11)	5.0×10^{-3}	13	0.41	346 033	7 216 711	CCW(5)	2 920
BP(16)	2.5×10^{-3}	27	0.042	1 439 356	105 416 196	aCCW(50 000, 0.6)	64 310

Figure 3 presents how the mean order p and the number of points in $A^{(j)}$ fluctuate according to the iterations. The green curve shows how the number of nodes varies with the CCW(5) strategy. At the beginning, a small number of nodes are added in the basis, and it increases strongly during the iterations. On the opposite, with the aCCW strategy, we quickly reach the target number of nodes to add, but this strategy needs more iterations than CCW to obtain the convergence. Finally, the aCCW strategy generally gives the smallest final basis sets.

4.2.2 The first 200 eigenvalues

For the mid-IR spectrum of C_2H_4O, the domain of spectroscopic interest is between $2\,800$ and $3\,200\,\mathrm{cm}^{-1}$. In this frequency range, the accurate assignment of active bands is still a challenging matter [33, 39, 46]. In Thomas and Carrington [46], they compute the first 200 eigenvalues by a tensor-type approach and the last calculated frequency is $3\,233\,\mathrm{cm}^{-1}$. Brown and Carrington [13] showed that the computation of Ref. [46] is not accurate for the first 50 lowest frequencies and would be certainly worse for higher frequencies. The ground state calculated by Thomas and Carrington [46] is $12\,461.860\,\mathrm{cm}^{-1}$ whereas the "O Expand" basis of Ref. [13] provides a converged value of $12\,461.467\,\mathrm{cm}^{-1}$.

In this section, the starting subspace $B^{(0)}$ contains the first 200 elements of $\Pi_{\mathbf{d}}$, with

$$\mathbf{d} = [5, 5, 4, 4, 4, 4, 3, 3, 3, 3, 3, 2, 2, 2, 2]$$

corresponding to the lowest vibrational states, and we expand the basis with the aCCW strategy in order to minimize the memory used by the algorithm. Table 5 presents a preliminary computation to evaluate which approximation space $\Pi_{\mathbf{d}}$ gives a good accuracy for a threshold of 5×10^{-3}. The ground state obtained is always $12\,461.480\,\mathrm{cm}^{-1}$ except for $\Pi_{\mathbf{d}}(12\,000)$. This means that this space is too small to accurately approximate the eigenfunctions. This result is confirmed in Brown et. al. [13] when they compare the maximal error on their eigenvalues with those of Thomas et.al. [46]). For $E_{max} = 12\,000\,\mathrm{cm}^{-1}$, the largest error is $0.300, \mathrm{cm}^{-1}$ (resp. $23.124, \mathrm{cm}^{-1}$). while with $E_{max} \geq 13\,000\,\mathrm{cm}^{-1}$, the error stabilizes at $0.258, \mathrm{cm}^{-1}$ (resp. $23.142, \mathrm{cm}^{-1}$). These first results also confirm that the computations presented in Thomas's paper [46] have a low accuracy.

According to the previous results, we set E_{max} to $13\,500\,\mathrm{cm}^{-1}$ to perform a more accurate computation with a threshold of 3×10^{-3} and the aCCW strategy with $N_T = 100\,000$ for the targeted number of nodes to add at each iteration and $\delta = 0.4$.

This large computation is done on a 72 cores platform with 768 Gb of memory. This run stopped after nearly 5 days of computation at iteration 44 due to a walltime limitation on the batch scheduler and needed
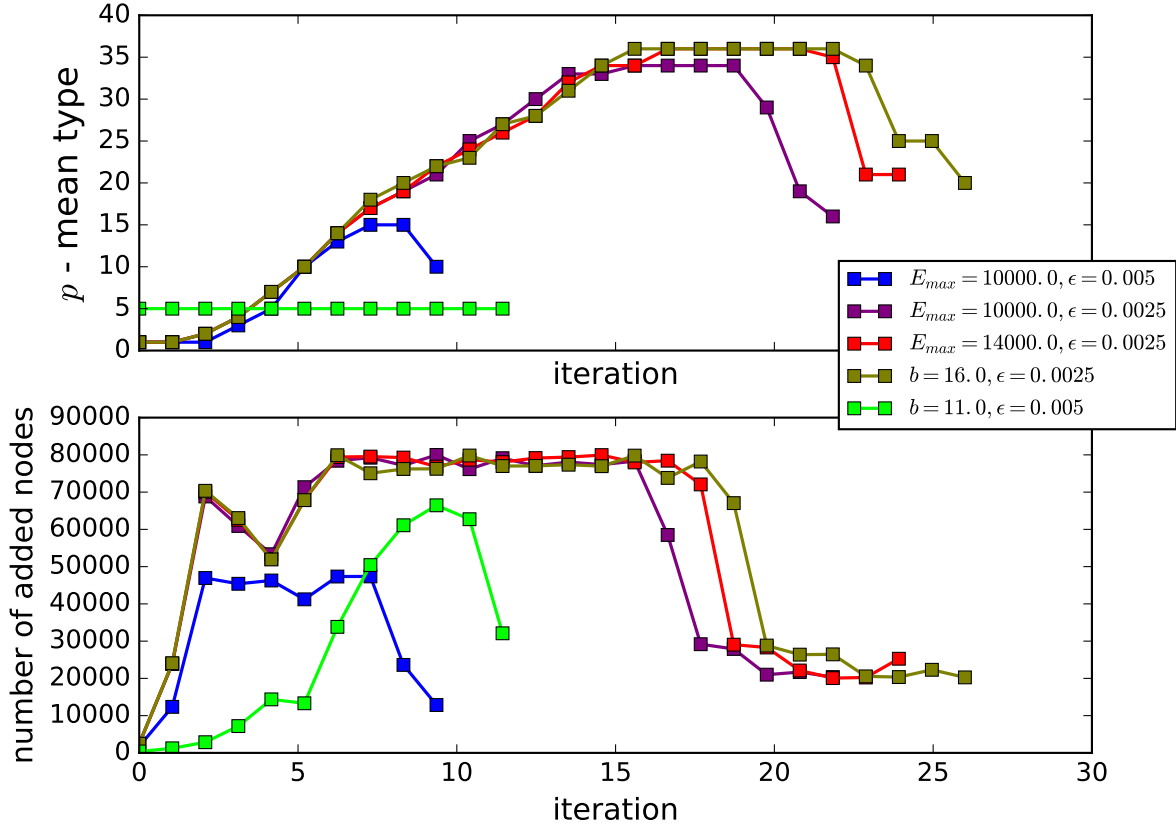
Figure 3: Evolution of the average type p during the A-VCI iterations for C_2H_4O ($F = 50$). The basis expansion strategy is aCCW, except for the green curve obtained with CCW(5). The used approximation spaces are $\Pi_{\mathbf{d}}(E_{max})$ and BP(b).

Table 5: Convergence of the 200 lowest frequencies of C_2H_4O computed with A-VCI in $\Pi_{\mathbf{d}}$, with E_{max} ranging from $12\,000$ to $15\,000\,\text{cm}^{-1}$. The used threshold is 5×10^{-3}.

E_{max} (cm^{-1})	Strategy aCCW(N_T, δ)	Iterations	m	m_R	Time (s)
$12\,000$	$(40000, 0.6)$	24	$1\,469\,373$	$108\,175\,774$	$173\,625$
$13\,000$	$(60000, 0.6)$	26	$1\,593\,865$	$124\,495\,044$	$217\,053$
$13\,500$	$(60000, 0.6)$	26	$1\,593\,892$	$124\,497\,716$	$216\,692$
$14\,000$	$(60000, 0.6)$	26	$1\,593\,892$	$124\,497\,874$	$222\,639$
$15\,000$	$(60000, 0.6)$	26	$1\,593\,892$	$124\,497\,898$	$217\,667$

less than 200 Gb of memory. The basis size obtained at this step counts $5\,328\,639$ elements and the related research space size is $308\,316\,882$. We reach the convergence for 197 eigenvalues over 200 but the largest residue is lower than 3.6×10^{-3}. Except for the 187th, 194th and 199th frequencies, the relative residues are all below 3×10^{-3} as shown in Fig. 4. In the bottom of the spectrum frequencies are well-converged and A-VCI stands very close to the 50 eigenvalues computed by Brown and Carrington [13], with a maximal error of $0.1052\,\text{cm}^{-1}$. Compared to the $2\,955\,289$ basis elements used in Ref. [13], the A-VCI basis is larger, but is able to calculate

more eigenstates than Brown and Carrington. As expected the discrepancies between A-VCI and Thomas and Carrington [46] are more important: the largest error on the 200 computed eigenvalues is $23.504\,\mathrm{cm^{-1}}$, and the mean error is $6.948\,\mathrm{cm^{-1}}$. This is not surprising since the highest frequencies are generally the most difficult to converge: we see on Fig. 4 that they correspond to the largest residues. Nevertheless, even on the first 50 eigenvalues, the largest error between A-VCI and Ref. [46] is $17.050\,\mathrm{cm^{-1}}$ and the mean error is $2.428\,\mathrm{cm^{-1}}$, whereas the results of Brown and Carrington are in good accordance with A-VCI. Finally, the ground state evaluated at $12\,461.474\,\mathrm{cm^{-1}}$ by A-VCI is close to the value found by Brown and Carrington [13] and assess of the accuracy of this calculation. The corresponding eigenvalues, frequencies and assignments are reported in the Supplementary Material.
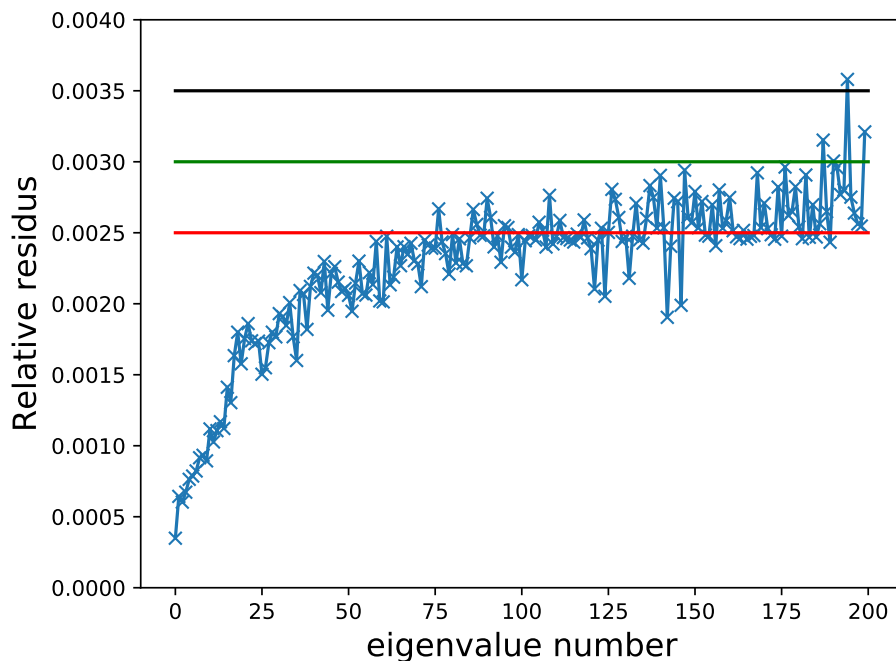
Figure 4: Final relative residues at iteration 44 for C_2H_4O ($\varepsilon = 3 \times 10^{-3}$ and $E_{max} = 13\,500\,\mathrm{cm^{-1}}$).

5   Conclusion

The A-VCI algorithm[23] is an adaptative procedure to compute iteratively a spectrum at any desired accuracy. It is applied to the resolution of the vibrational Schrödinger equation of two molecular systems (CH_3CN and C_2H_4O), for which sum of product potential energy surfaces, generated by Bégué *et al.* [7, 8], have been used by several authors[1, 25, 3, 13, 30, 46] to produce the most accurate results currently available.

This iterative method builds nested spaces from an initial active space inside a much larger approximation space. The use of the Bauer-Fike theorem led us to define an *a-posteriori* criterion to control the convergence of the algorithm and to select significant basis elements to add at each iteration. Three parameters, namely ε the threshold to control the convergence, E_{max} to customize the size of the approximation space and p to adjust the selection criterion at each iteration, have been introduced to increase the flexibility of the method. A full study of the influence of these parameters was carried out. We are now able to choose efficiently these parameters to reach the desired precision for a given PES and a given frequency range. In addition to providing

a great accuracy for a large frequency range ($0 - 3000$ cm^{-1} for CH$_3$CN and $0 - 3200$ cm^{-1} for C$_2$H$_4$O), the constraints on the parameters can be relaxed to reach an accuracy of $\simeq 1$ cm^{-1} on the same energetic range, with very small resulting basis sets.

Our approach to manage the basis elements and the matrix allows us to handle molecular systems up to 7 atoms, in less than a day on a 24-core computer with 128 GB of memory capacity (except for the largest C$_2$H$_4$O results). This limit can be postponed either by using a matrix free approach for the H_R matrix elements to reduce the memory foot-print or by using a distributed memory programming (Message Passing Interface) to spread the workload over different platforms.

The A-VCI algorithm extracts eigenvalues from an Hamiltonian operator with an accuracy we are able to estimate. Thus, a comparison with experimental data could allow to deduce the adequacy between an operator (in particular its potential energy part) and the chemical system studied. Moreover, additions such as Coriolis effects or more sophisticated rotational-vibrational couplings will ultimately be useful to help investigate new interpretations of experimental data.

acknowledgments

A   Hermite functions

The 1-D normalized Hermite function of degree n writes

$$\psi_n(q) = \left(\frac{1}{\pi}\right)^{1/4} \frac{1}{\sqrt{2^n\, n!}} \cdot e^{-\frac{q^2}{2}} \cdot \mathrm{He}_n(q),$$

where He_n is the normalized Hermite polynomial of degree n. These polynomials are orthogonal for the weight $e^{-\frac{q^2}{2}}$. Moreover we have

$$\int_{\mathbb{R}} \psi_n(x)\psi_m(x) \ = \delta_{n,m}.$$

The sequence of Hermite polynomials satisfies the recursion

$$\mathrm{He}_{n+1}(q) = q\mathrm{He}_n - n\mathrm{He}_{n-1},$$

and ψ_n are the eigenfunctions of the 1-D harmonic Hamiltonian operator defined by

$$\mathcal{H}_0(q) = -\frac{\partial^2}{\partial q^2} + q^2.$$

The properties of Hermite functions provide an analytic formula to calculate the Hamiltonian matrix coefficients as shown in the following propositions.

Proposition 1 (see [49]) *The indices m such that $\langle \phi_n(q)|q^s\phi_m(q)\rangle \neq 0$ are given by*

- *If $s = 2p$, then $m = n + 2l$ for $-s/2 \leq l \leq s/2$ if $s = 2p$.*

- *If $s = 2p + 1$, then $m = n + 2l + 1$ for $-(s-1)/2 \leq l \leq (s-1)/2$.*

Proposition 2 (see [49]) *The tensor moment $\mathcal{M}(s, n, m) = \langle \phi_n(q), q^s\phi_m(q)\rangle$ has the following analytic expression*

$$\langle \phi_n(q)|q^s\phi_m(q)\rangle = \begin{cases} 0 & \text{if } s - n - m \text{ is odd,} \\ \dfrac{s!}{2^r}\sqrt{\dfrac{2^{m+n}}{m!n!}} \displaystyle\sum_{p=\max(0,-r)}^{\min(n,m)} C_p^n C_p^m \dfrac{p!}{2^p(r+p)!} & \text{otherwise,} \end{cases} \tag{24}$$

with $r = (s - n - m)/2$.

B Supplementary Material

B.1 Acetonitrile molecule

Table 6: Frequencies and assignments for two A-VCI computations. The first column corresponds to the more accurate one with a final basis set of $2\,488\,511$ elements while in the second computation, the basis size is only of $79\,133$ elements

Frequency Number	$\Pi_{\mathbf{d}}(15\,000)$, $\varepsilon = 2.5\,10^{-4}$ (cm^{-1})	$\Pi_{\mathbf{d}}(12\,500)$, $\varepsilon = 0.005$ (cm^{-1})	Reference [1] (cm^{-1})	Assignment
1	360.990	361.048	360.991	$\omega_2(0.97)$, $\omega_2 + \omega_{10}(0.20)$
2	360.990	361.059	360.991	$\omega_1(0.97)$, $\omega_1 + \omega_{10}(0.20)$
3	723.179	723.240	723.181	$\omega_1 + \omega_2(0.97)$, $\omega_1 + \omega_2 + \omega_{10}(0.20)$
4	723.179	723.253	723.181	$2\omega_2(0.69)$, $2\omega_1(0.69)$
5	723.825	723.886	723.827	$2\omega_1(0.68)$, $2\omega_2(0.68)$
6	900.657	900.722	900.662	$\omega_3(0.95)$, $\omega_3 + \omega_{10}(0.19)$
7	1034.125	1034.231	1034.126	$\omega_5(0.97)$, $\omega_5 + \omega_{10}(0.21)$
8	1034.126	1034.271	1034.126	$\omega_4(0.97)$, $\omega_4 + \omega_{10}(0.21)$
9	1086.552	1086.657	1086.554	$\omega_1 + 2\omega_2(0.84)$, $3\omega_1(0.48)$
10	1086.552	1086.661	1086.554	$2\omega_1 + \omega_2(0.84)$, $3\omega_2(0.48)$
11	1087.774	1087.881	1087.776	$3\omega_2(0.83)$, $2\omega_1 + \omega_2(0.48)$
12	1087.774	1087.888	1087.776	$3\omega_1(0.83)$, $\omega_1 + 2\omega_2(0.48)$
13	1259.807	1259.916	1259.822	$\omega_2 + \omega_3(0.94)$, $\omega_2 + 2\omega_3(0.21)$
14	1259.807	1259.921	1259.822	$\omega_1 + \omega_3(0.94)$, $\omega_1 + 2\omega_3(0.21)$
15	1388.969	1389.103	1388.973	$\omega_6(0.73)$, $\omega_2 + \omega_5(0.45)$
16	1394.681	1394.811	1394.689	$\omega_1 + \omega_5(0.68)$, $\omega_2 + \omega_4(0.68)$
17	1394.681	1394.815	1394.689	$\omega_2 + \omega_5(0.68)$, $\omega_1 + \omega_4(0.68)$
18	1394.900	1395.032	1394.907	$\omega_2 + \omega_4(0.68)$, $\omega_1 + \omega_5(0.68)$
19	1397.681	1397.813	1397.687	$\omega_6(0.64)$, $\omega_1 + \omega_4(0.51)$
20	1451.093	1451.194	1451.101	$\omega_1 + 3\omega_2(0.68)$, $3\omega_1 + \omega_2(0.68)$
21	1451.093	1451.251	1451.101	$2\omega_1 + 2\omega_2(0.84)$, $4\omega_1(0.34)$
22	1452.818	1452.920	1452.827	$3\omega_1 + \omega_2(0.68)$, $\omega_1 + 3\omega_2(0.68)$
23	1452.818	1452.957	1452.827	$4\omega_2(0.68)$, $4\omega_1(0.68)$
24	1453.394	1453.540	1453.403	$4\omega_1(0.58)$, $4\omega_2(0.58)$
25	1483.220	1483.396	1483.229	$\omega_8(0.97)$, $\omega_8 + \omega_{10}(0.20)$
26	1483.220	1483.426	1483.229	$\omega_7(0.97)$, $\omega_7 + \omega_{10}(0.20)$
27	1620.196	1620.278	1620.222	$2\omega_2 + \omega_3(0.66)$, $2\omega_1 + \omega_3(0.66)$
28	1620.196	1620.313	1620.222	$\omega_1 + \omega_2 + \omega_3(0.93)$, $\omega_1 + \omega_2 + 2\omega_3(0.22)$
29	1620.740	1620.859	1620.767	$2\omega_1 + \omega_3(0.65)$, $2\omega_2 + \omega_3(0.65)$
30	1749.520	1749.697	1749.530	$\omega_2 + \omega_6(0.74)$, $2\omega_2 + \omega_5(0.50)$
31	1749.520	1749.707	1749.530	$\omega_1 + \omega_6(0.74)$, $2\omega_1 + \omega_4(0.50)$
32	1756.414	1756.576	1756.426	$\omega_1 + \omega_2 + \omega_5(0.68)$, $2\omega_1 + \omega_4(0.48)$
33	1756.414	1756.579	1756.426	$\omega_1 + \omega_2 + \omega_4(0.68)$, $2\omega_2 + \omega_5(0.48)$
34	1757.121	1757.300	1757.133	$2\omega_1 + \omega_5(0.83)$, $\omega_1 + \omega_2 + \omega_4(0.39)$
35	1757.121	1757.338	1757.133	$2\omega_2 + \omega_4(0.83)$, $\omega_1 + \omega_2 + \omega_5(0.38)$
36	1759.762	1759.922	1759.772	$\omega_2 + \omega_6(0.62)$, $2\omega_2 + \omega_5(0.60)$
37	1759.762	1759.934	1759.772	$\omega_1 + \omega_6(0.62)$, $2\omega_1 + \omega_4(0.60)$
38	1785.098	1785.180	1785.207	$2\omega_3(0.89)$, $3\omega_3(0.31)$

Table 6 (continued)

Frequency Number	$\Pi_{\mathbf{d}}(15\,000)$, $\varepsilon = 2.5\,10^{-4}$ (cm^{-1})	$\Pi_{\mathbf{d}}(12\,500)$, $\varepsilon = 0.005$ (cm^{-1})	Reference [1] (cm^{-1})	Assignment
39	1816.786	1816.963	1816.799	$2\omega_1 + 3\omega_2(0.76)$, $4\omega_1 + \omega_2(0.54)$
40	1816.786	1816.973	1816.799	$3\omega_1 + 2\omega_2(0.76)$, $\omega_1 + 4\omega_2(0.54)$
41	1818.938	1819.117	1818.952	$\omega_1 + 4\omega_2(0.71)$, $5\omega_1(0.53)$
42	1818.939	1819.126	1818.952	$4\omega_1 + \omega_2(0.71)$, $5\omega_2(0.53)$
43	1820.016	1820.191	1820.031	$5\omega_2(0.75)$, $2\omega_1 + 3\omega_2(0.47)$
44	1820.016	1820.200	1820.031	$5\omega_1(0.75)$, $3\omega_1 + 2\omega_2(0.47)$
45	1844.245	1844.478	1844.258	$\omega_1 + \omega_8(0.68)$, $\omega_2 + \omega_7(0.68)$
46	1844.317	1844.523	1844.330	$\omega_2 + \omega_7(0.68)$, $\omega_1 + \omega_8(0.68)$
47	1844.318	1844.551	1844.330	$\omega_2 + \omega_8(0.68)$, $\omega_1 + \omega_7(0.68)$
48	1844.677	1844.880	1844.690	$\omega_1 + \omega_7(0.68)$, $\omega_2 + \omega_8(0.68)$
49	1931.513	1931.657	1931.547	$\omega_3 + \omega_5(0.94)$, $2\omega_3 + \omega_5(0.21)$
50	1931.513	1931.668	1931.547	$\omega_3 + \omega_4(0.94)$, $2\omega_3 + \omega_4(0.21)$
51	1981.811	1981.932	1981.849	$\omega_1 + 2\omega_2 + \omega_3(0.80)$, $3\omega_1 + \omega_3(0.46)$
52	1981.812	1981.942	1981.849	$2\omega_1 + \omega_2 + \omega_3(0.80)$, $3\omega_2 + \omega_3(0.46)$
53	1982.812	1982.943	1982.857	$3\omega_1 + \omega_3(0.79)$, $\omega_1 + 2\omega_2 + \omega_3(0.45)$
54	1982.812	1982.944	1982.857	$3\omega_2 + \omega_3(0.79)$, $2\omega_1 + \omega_2 + \omega_3(0.45)$
55	2057.051	2057.244	2057.068	$2\omega_5(0.68)$, $2\omega_4(0.68)$
56	2065.269	2065.409	2065.286	$2\omega_4(0.68)$, $2\omega_5(0.68)$
57	2065.269	2065.559	2065.286	$\omega_4 + \omega_5(0.96)$, $\omega_4 + \omega_5 + \omega_{10}(0.22)$
58	2111.365	2111.567	2111.380	$\omega_1 + \omega_2 + \omega_6(0.75)$, $2\omega_1 + \omega_2 + \omega_4(0.43)$
59	2111.365	2111.650	2111.380	$2\omega_2 + \omega_6(0.53)$, $2\omega_1 + \omega_6(0.53)$
60	2112.282	2112.485	2112.297	$2\omega_1 + \omega_6(0.53)$, $2\omega_2 + \omega_6(0.53)$
61	2119.309	2119.542	2119.327	$2\omega_1 + \omega_2 + \omega_4(0.59)$, $\omega_1 + 2\omega_2 + \omega_5(0.59)$
62	2119.309	2119.548	2119.327	$2\omega_1 + \omega_2 + \omega_5(0.59)$, $\omega_1 + 2\omega_2 + \omega_4(0.59)$
63	2120.523	2120.728	2120.541	$3\omega_1 + \omega_5(0.67)$, $3\omega_2 + \omega_4(0.67)$
64	2120.523	2120.760	2120.541	$\omega_1 + 2\omega_2 + \omega_4(0.57)$, $2\omega_1 + \omega_2 + \omega_5(0.57)$
65	2120.891	2121.106	2120.910	$3\omega_2 + \omega_4(0.58)$, $3\omega_1 + \omega_5(0.58)$
66	2122.818	2123.020	2122.834	$\omega_1 + \omega_2 + \omega_6(0.61)$, $2\omega_1 + \omega_2 + \omega_4(0.52)$
67	2122.818	2123.086	2122.834	$3\omega_2 + \omega_5(0.45)$, $3\omega_1 + \omega_4(0.45)$
68	2123.283	2123.490	2123.301	$3\omega_2 + \omega_5(0.45)$, $3\omega_1 + \omega_4(0.45)$
69	2142.348	2142.586	2142.614	$\omega_1 + 2\omega_3(0.87)$, $\omega_1 + 3\omega_3(0.32)$
70	2142.348	2142.600	2142.614	$\omega_2 + 2\omega_3(0.87)$, $\omega_2 + 3\omega_3(0.32)$
71	2183.618	2183.752	2183.635	$2\omega_1 + 4\omega_2(0.66)$, $4\omega_1 + 2\omega_2(0.66)$
72	2183.618	2183.795	2183.635	$3\omega_1 + 3\omega_2(0.76)$, $5\omega_1 + \omega_2(0.42)$
73	2186.117	2186.260	2186.138	$\omega_1 + 5\omega_2(0.67)$, $5\omega_1 + \omega_2(0.67)$
74	2186.117	2186.269	2186.138	$2\omega_1 + 4\omega_2(0.53)$, $4\omega_1 + 2\omega_2(0.53)$
75	2187.618	2187.762	2187.642	$6\omega_2(0.64)$, $6\omega_1(0.64)$
76	2187.618	2187.799	2187.642	$3\omega_1 + 3\omega_2(0.57)$, $5\omega_1 + \omega_2(0.52)$
77	2188.119	2188.269	2188.144	$6\omega_2(0.52)$, $6\omega_1(0.52)$
78	2206.608	2206.813	2206.626	$\omega_1 + \omega_2 + \omega_8(0.68)$, $2\omega_2 + \omega_7(0.48)$
79	2206.616	2206.917	2206.633	$\omega_1 + \omega_2 + \omega_7(0.68)$, $2\omega_2 + \omega_8(0.48)$
80	2206.758	2206.959	2206.776	$2\omega_1 + \omega_8(0.73)$, $\omega_1 + \omega_2 + \omega_7(0.62)$
81	2206.758	2207.071	2206.776	$2\omega_2 + \omega_7(0.73)$, $\omega_1 + \omega_2 + \omega_8(0.62)$
82	2207.542	2207.753	2207.559	$2\omega_2 + \omega_8(0.82)$, $2\omega_1 + \omega_8(0.41)$
83	2207.542	2207.835	2207.559	$2\omega_1 + \omega_7(0.82)$, $2\omega_2 + \omega_7(0.41)$
84	2250.710	2250.870	2250.746	$\omega_9(0.90)$, $\omega_3 + \omega_6(0.30)$
85	2287.846	2288.038	2287.939	$\omega_2 + \omega_3 + \omega_5(0.60)$, $\omega_1 + \omega_3 + \omega_4(0.60)$
86	2290.118	2290.313	2290.222	$\omega_1 + \omega_3 + \omega_4(0.66)$, $\omega_2 + \omega_3 + \omega_5(0.66)$

Table 6 (continued)

Frequency Number	$\Pi_{\mathbf{d}}(15\,000)$, $\varepsilon = 2.5\,10^{-4}$ (cm^{-1})	$\Pi_{\mathbf{d}}(12\,500)$, $\varepsilon = 0.005$ (cm^{-1})	Reference [1] (cm^{-1})	Assignment
87	2290.118	2290.362	2290.222	$\omega_1 + \omega_3 + \omega_5(0.66)$, $\omega_2 + \omega_3 + \omega_4(0.66)$
88	2290.281	2290.524	2290.384	$\omega_2 + \omega_3 + \omega_4(0.66)$, $\omega_1 + \omega_3 + \omega_5(0.66)$
89	2297.130	2297.299	2297.189	$\omega_3 + \omega_6(0.83)$, $\omega_1 + \omega_3 + \omega_4(0.26)$
90	2344.642	2344.871	2344.748	$\omega_1 + 3\omega_2 + \omega_3(0.65)$, $3\omega_1 + \omega_2 + \omega_3(0.64)$
91	2344.642	2344.875	2344.748	$2\omega_1 + 2\omega_2 + \omega_3(0.79)$, $4\omega_1 + \omega_3(0.32)$
92	2346.006	2346.214	2346.127	$4\omega_2 + \omega_3(0.63)$, $4\omega_1 + \omega_3(0.63)$
93	2346.006	2346.251	2346.127	$3\omega_1 + \omega_2 + \omega_3(0.63)$, $\omega_1 + 3\omega_2 + \omega_3(0.63)$
94	2346.461	2346.689	2346.588	$4\omega_2 + \omega_3(0.54)$, $4\omega_1 + \omega_3(0.54)$
95	2384.790	2384.918	2384.873	$\omega_3 + \omega_8(0.95)$, $\omega_3 + \omega_8 + \omega_{10}(0.20)$
96	2384.791	2385.105	2384.873	$\omega_3 + \omega_7(0.95)$, $\omega_3 + \omega_7 + \omega_{10}(0.20)$
97	2414.963	2415.178	2415.022	$\omega_2 + 2\omega_5(0.68)$, $\omega_5 + \omega_6(0.51)$
98	2414.963	2415.228	2415.022	$\omega_1 + 2\omega_4(0.68)$, $\omega_4 + \omega_6(0.51)$
99	2420.083	2420.294	2420.135	$\omega_1 + 2\omega_5(0.65)$, $\omega_4 + \omega_6(0.60)$
100	2420.083	2420.331	2420.135	$\omega_2 + 2\omega_4(0.65)$, $\omega_5 + \omega_6(0.60)$
101	2425.374	2425.570	2425.443	$\omega_2 + \omega_4 + \omega_5(0.67)$, $\omega_1 + 2\omega_4(0.48)$
102	2425.408	2425.648	2425.478	$\omega_1 + \omega_4 + \omega_5(0.68)$, $\omega_2 + 2\omega_4(0.48)$
103	2427.910	2428.109	2427.963	$\omega_4 + \omega_6(0.55)$, $\omega_2 + \omega_4 + \omega_5(0.55)$
104	2427.910	2428.153	2427.963	$\omega_5 + \omega_6(0.55)$, $\omega_1 + \omega_4 + \omega_5(0.55)$
105	2474.470	2474.699	2474.505	$\omega_1 + 2\omega_2 + \omega_6(0.65)$, $\omega_1 + 3\omega_2 + \omega_5(0.39)$
106	2474.470	2474.720	2474.505	$2\omega_1 + \omega_2 + \omega_6(0.65)$, $3\omega_1 + \omega_2 + \omega_4(0.39)$
107	2476.137	2476.369	2476.173	$3\omega_1 + \omega_6(0.65)$, $4\omega_1 + \omega_4(0.45)$
108	2476.137	2476.374	2476.173	$3\omega_2 + \omega_6(0.65)$, $4\omega_2 + \omega_5(0.45)$
109	2483.351	2483.574	2483.414	$2\omega_1 + 2\omega_2 + \omega_5(0.58)$, $3\omega_1 + \omega_2 + \omega_4(0.48)$
110	2483.351	2483.619	2483.414	$2\omega_1 + 2\omega_2 + \omega_4(0.58)$, $3\omega_1 + \omega_2 + \omega_5(0.48)$
111	2485.027	2485.290	2485.091	$3\omega_1 + \omega_2 + \omega_5(0.62)$, $4\omega_2 + \omega_4(0.53)$
112	2485.028	2485.316	2485.092	$\omega_1 + 3\omega_2 + \omega_4(0.62)$, $4\omega_1 + \omega_5(0.53)$
113	2485.806	2486.124	2485.871	$4\omega_2 + \omega_4(0.75)$, $2\omega_1 + 2\omega_2 + \omega_4(0.38)$
114	2485.806	2486.126	2485.871	$4\omega_1 + \omega_5(0.75)$, $2\omega_1 + 2\omega_2 + \omega_5(0.38)$
115	2486.912	2487.135	2486.958	$\omega_1 + 2\omega_2 + \omega_6(0.52)$, $\omega_1 + 3\omega_2 + \omega_5(0.48)$
116	2486.912	2487.162	2486.958	$2\omega_1 + \omega_2 + \omega_6(0.52)$, $3\omega_1 + \omega_2 + \omega_4(0.48)$
117	2487.740	2487.978	2487.791	$4\omega_1 + \omega_4(0.57)$, $3\omega_1 + \omega_6(0.51)$
118	2487.740	2487.978	2487.791	$4\omega_2 + \omega_5(0.57)$, $3\omega_2 + \omega_6(0.51)$
119	2500.898	2501.101	2501.264	$2\omega_1 + 2\omega_3(0.60)$, $2\omega_2 + 2\omega_3(0.60)$
120	2500.898	2501.230	2501.264	$\omega_1 + \omega_2 + 2\omega_3(0.85)$, $\omega_1 + \omega_2 + 3\omega_3(0.34)$
121	2501.287	2501.630	2501.686	$2\omega_2 + 2\omega_3(0.59)$, $2\omega_1 + 2\omega_3(0.59)$
122	2519.081	2519.330	2519.143	$\omega_5 + \omega_8(0.68)$, $\omega_4 + \omega_7(0.68)$
123	2519.772	2520.120	2519.834	$\omega_5 + \omega_7(0.68)$, $\omega_4 + \omega_8(0.68)$
124	2522.051	2522.296	2522.114	$\omega_4 + \omega_8(0.68)$, $\omega_5 + \omega_7(0.68)$
125	2522.051	2522.393	2522.114	$\omega_4 + \omega_7(0.68)$, $\omega_5 + \omega_8(0.68)$
126	2551.573	2551.746	2551.630	$4\omega_1 + 3\omega_2(0.71)$, $2\omega_1 + 5\omega_2(0.55)$
127	2551.573	2551.770	2551.630	$3\omega_1 + 4\omega_2(0.71)$, $5\omega_1 + 2\omega_2(0.55)$
128	2554.335	2554.509	2554.398	$2\omega_1 + 5\omega_2(0.61)$, $6\omega_1 + \omega_2(0.59)$
129	2554.335	2554.532	2554.398	$5\omega_1 + 2\omega_2(0.61)$, $\omega_1 + 6\omega_2(0.59)$
130	2556.177	2556.365	2556.245	$\omega_1 + 6\omega_2(0.60)$, $7\omega_1(0.53)$
131	2556.180	2556.382	2556.248	$6\omega_1 + \omega_2(0.60)$, $7\omega_2(0.53)$
132	2557.101	2557.291	2557.171	$7\omega_2(0.68)$, $2\omega_1 + 5\omega_2(0.45)$
133	2557.101	2557.313	2557.171	$7\omega_1(0.68)$, $5\omega_1 + 2\omega_2(0.45)$
134	2570.088	2570.360	2570.145	$2\omega_1 + \omega_2 + \omega_7(0.59)$, $\omega_1 + 2\omega_2 + \omega_8(0.59)$

Table 6 (continued)

Frequency Number	$\Pi_{\mathbf{d}}(15\,000)$, $\varepsilon = 2.5\,10^{-4}$ (cm^{-1})	$\Pi_{\mathbf{d}}(12\,500)$, $\varepsilon = 0.005$ (cm^{-1})	Reference [1] (cm^{-1})	Assignment
135	2570.088	2570.392	2570.145	$2\omega_1 + \omega_2 + \omega_8(0.60)$, $\omega_1 + 2\omega_2 + \omega_7(0.59)$
136	2570.395	2570.667	2570.452	$3\omega_1 + \omega_8(0.52)$, $3\omega_2 + \omega_7(0.51)$
137	2570.395	2570.708	2570.452	$\omega_1 + 2\omega_2 + \omega_7(0.67)$, $2\omega_1 + \omega_2 + \omega_8(0.66)$
138	2571.173	2571.457	2571.231	$3\omega_2 + \omega_7(0.59)$, $3\omega_1 + \omega_8(0.59)$
139	2571.575	2571.851	2571.633	$2\omega_1 + \omega_2 + \omega_7(0.52)$, $\omega_1 + 2\omega_2 + \omega_8(0.51)$
140	2571.575	2571.861	2571.633	$3\omega_2 + \omega_8(0.67)$, $3\omega_1 + \omega_7(0.66)$
141	2572.014	2572.305	2572.072	$3\omega_2 + \omega_8(0.59)$, $3\omega_1 + \omega_7(0.59)$
142	2609.197	2609.508	2609.256	$\omega_1 + \omega_9(0.90)$, $\omega_1 + \omega_3 + \omega_6(0.29)$
143	2609.197	2609.548	2609.256	$\omega_2 + \omega_9(0.90)$, $\omega_2 + \omega_3 + \omega_6(0.29)$
144	2646.939	2647.249	2647.073	$2\omega_2 + \omega_3 + \omega_5(0.65)$, $\omega_1 + \omega_2 + \omega_3 + \omega_4(0.50)$
145	2646.939	2647.265	2647.073	$2\omega_1 + \omega_3 + \omega_4(0.65)$, $\omega_1 + \omega_2 + \omega_3 + \omega_5(0.50)$
146	2649.944	2650.259	2650.084	$\omega_1 + \omega_2 + \omega_3 + \omega_5(0.65)$, $2\omega_1 + \omega_3 + \omega_4(0.46)$
147	2649.945	2650.290	2650.086	$\omega_1 + \omega_2 + \omega_3 + \omega_4(0.65)$, $2\omega_2 + \omega_3 + \omega_5(0.46)$
148	2650.502	2650.801	2650.647	$2\omega_2 + \omega_3 + \omega_4(0.79)$, $\omega_1 + \omega_2 + \omega_3 + \omega_5(0.34)$
149	2650.502	2650.895	2650.647	$2\omega_1 + \omega_3 + \omega_5(0.79)$, $\omega_1 + \omega_2 + \omega_3 + \omega_4(0.34)$
150	2651.593	2651.862	2653.053	$3\omega_3(0.78)$, $4\omega_3(0.41)$
151	2656.657	2656.961	2656.793	$\omega_1 + \omega_3 + \omega_6(0.80)$, $2\omega_1 + \omega_3 + \omega_4(0.33)$
152	2656.657	2656.992	2656.793	$\omega_2 + \omega_3 + \omega_6(0.80)$, $2\omega_2 + \omega_3 + \omega_5(0.33)$
153	2708.676	2708.935	2708.821	$2\omega_1 + 3\omega_2 + \omega_3(0.71)$, $4\omega_1 + \omega_2 + \omega_3(0.50)$
154	2708.676	2709.047	2708.821	$3\omega_1 + 2\omega_2 + \omega_3(0.71)$, $\omega_1 + 4\omega_2 + \omega_3(0.50)$
155	2710.305	2710.563	2710.482	$\omega_1 + 4\omega_2 + \omega_3(0.65)$, $5\omega_1 + \omega_3(0.48)$
156	2710.307	2710.669	2710.484	$4\omega_1 + \omega_2 + \omega_3(0.65)$, $5\omega_2 + \omega_3(0.48)$
157	2711.121	2711.365	2711.318	$5\omega_1 + \omega_3(0.67)$, $3\omega_1 + 2\omega_2 + \omega_3(0.42)$
158	2711.121	2711.477	2711.318	$5\omega_2 + \omega_3(0.67)$, $2\omega_1 + 3\omega_2 + \omega_3(0.42)$
159	2743.916	2744.107	2744.028	$\omega_1 + \omega_3 + \omega_8(0.66)$, $\omega_2 + \omega_3 + \omega_7(0.66)$
160	2743.966	2744.140	2744.078	$\omega_2 + \omega_3 + \omega_7(0.66)$, $\omega_1 + \omega_3 + \omega_8(0.66)$
161	2743.966	2744.154	2744.078	$\omega_2 + \omega_3 + \omega_8(0.66)$, $\omega_1 + \omega_3 + \omega_7(0.66)$
162	2744.356	2744.527	2744.468	$\omega_1 + \omega_3 + \omega_7(0.66)$, $\omega_2 + \omega_3 + \omega_8(0.66)$
163	2768.017	2768.282	2768.055	$2\omega_6(0.77)$, $\omega_2 + \omega_5 + \omega_6(0.36)$
164	2775.077	2775.379	2775.162	$\omega_1 + \omega_2 + 2\omega_5(0.52)$, $\omega_1 + \omega_2 + 2\omega_4(0.52)$
165	2775.077	2775.455	2775.162	$2\omega_2 + 2\omega_5(0.50)$, $2\omega_1 + 2\omega_4(0.50)$
166	2776.518	2776.828	2776.592	$2\omega_1 + 2\omega_4(0.47)$, $2\omega_2 + 2\omega_5(0.47)$
167	2777.356	2777.679	2777.430	$\omega_2 + \omega_4 + \omega_6(0.54)$, $\omega_1 + \omega_5 + \omega_6(0.54)$
168	2781.940	2782.250	2782.020	$2\omega_1 + 2\omega_5(0.49)$, $2\omega_2 + 2\omega_4(0.49)$
169	2782.232	2782.519	2782.313	$\omega_1 + \omega_5 + \omega_6(0.43)$, $\omega_2 + \omega_4 + \omega_6(0.43)$
170	2782.232	2782.576	2782.313	$2\omega_1 + 2\omega_5(0.51)$, $2\omega_2 + 2\omega_4(0.51)$
171	2786.666	2787.002	2786.752	$2\omega_2 + \omega_4 + \omega_5(0.48)$, $\omega_1 + \omega_2 + 2\omega_5(0.48)$
172	2786.666	2787.006	2786.752	$\omega_1 + \omega_2 + \omega_4 + \omega_5(0.67)$, $2\omega_1 + 2\omega_4(0.34)$
173	2789.026	2789.283	2789.109	$2\omega_2 + 2\omega_5(0.45)$, $2\omega_1 + 2\omega_4(0.44)$
174	2789.026	2789.381	2789.109	$2\omega_1 + \omega_4 + \omega_5(0.58)$, $2\omega_2 + \omega_4 + \omega_5(0.57)$
175	2790.219	2790.536	2790.298	$\omega_2 + \omega_4 + \omega_6(0.41)$, $\omega_1 + \omega_5 + \omega_6(0.41)$
176	2790.667	2790.958	2790.743	$\omega_1 + \omega_2 + \omega_4 + \omega_5(0.50)$, $\omega_1 + \omega_4 + \omega_6(0.43)$
177	2812.024	2812.320	2812.500	$2\omega_3 + \omega_5(0.87)$, $3\omega_3 + \omega_5(0.32)$
178	2812.024	2812.342	2812.500	$2\omega_3 + \omega_4(0.87)$, $3\omega_3 + \omega_4(0.32)$
179	2838.803	2839.142	2838.870	$\omega_1 + 3\omega_2 + \omega_6(0.53)$, $3\omega_1 + \omega_2 + \omega_6(0.53)$
180	2838.803	2839.155	2838.870	$2\omega_1 + 2\omega_2 + \omega_6(0.65)$, $3\omega_1 + 2\omega_2 + \omega_4(0.36)$
181	2841.089	2841.398	2841.159	$4\omega_2 + \omega_6(0.53)$, $4\omega_1 + \omega_6(0.53)$
182	2841.089	2841.454	2841.159	$3\omega_1 + \omega_2 + \omega_6(0.53)$, $\omega_1 + 3\omega_2 + \omega_6(0.53)$

Table 6 (continued)

Frequency Number	$\Pi_{\mathbf{d}}(15\,000)$, $\varepsilon = 2.5\,10^{-4}$ (cm^{-1})	$\Pi_{\mathbf{d}}(12\,500)$, $\varepsilon = 0.005$ (cm^{-1})	Reference [1] (cm^{-1})	Assignment
183	2841.869	2842.208	2841.940	$4\omega_1 + \omega_6(0.46)$, $4\omega_2 + \omega_6(0.46)$
184	2848.525	2848.865	2848.604	$2\omega_1 + 3\omega_2 + \omega_4(0.53)$, $3\omega_1 + 2\omega_2 + \omega_5(0.53)$
185	2848.525	2848.892	2848.604	$3\omega_1 + 2\omega_2 + \omega_4(0.53)$, $2\omega_1 + 3\omega_2 + \omega_5(0.53)$
186	2850.580	2850.926	2850.664	$5\omega_2 + \omega_4(0.40)$, $5\omega_1 + \omega_5(0.40)$
187	2850.580	2850.978	2850.664	$4\omega_1 + \omega_2 + \omega_5(0.58)$, $\omega_1 + 4\omega_2 + \omega_4(0.58)$
188	2851.765	2852.077	2851.852	$5\omega_1 + \omega_5(0.64)$, $5\omega_2 + \omega_4(0.64)$
189	2851.765	2852.185	2851.852	$\omega_1 + 4\omega_2 + \omega_4(0.45)$, $4\omega_1 + \omega_2 + \omega_5(0.45)$
190	2852.081	2852.422	2852.155	$\omega_1 + 3\omega_2 + \omega_6(0.42)$, $3\omega_1 + \omega_2 + \omega_6(0.42)$
191	2852.081	2852.441	2852.155	$2\omega_1 + 2\omega_2 + \omega_6(0.52)$, $3\omega_1 + 2\omega_2 + \omega_4(0.45)$
192	2852.146	2852.495	2852.233	$5\omega_1 + \omega_5(0.52)$, $5\omega_2 + \omega_4(0.52)$
193	2853.166	2853.516	2853.249	$5\omega_2 + \omega_5(0.45)$, $5\omega_1 + \omega_4(0.45)$
194	2853.167	2853.564	2853.249	$4\omega_1 + \omega_2 + \omega_4(0.43)$, $\omega_1 + 4\omega_2 + \omega_5(0.43)$
195	2853.526	2853.916	2853.611	$5\omega_2 + \omega_5(0.41)$, $5\omega_1 + \omega_4(0.41)$
196	2860.740	2860.972		$\omega_1 + 2\omega_2 + 2\omega_3(0.72)$, $3\omega_1 + 2\omega_3(0.42)$
197	2860.741	2861.017		$2\omega_1 + \omega_2 + 2\omega_3(0.72)$, $3\omega_2 + 2\omega_3(0.42)$
198	2861.403	2861.688		$3\omega_1 + 2\omega_3(0.69)$, $\omega_1 + 2\omega_2 + 2\omega_3(0.40)$
199	2861.403	2861.723		$3\omega_2 + 2\omega_3(0.69)$, $2\omega_1 + \omega_2 + 2\omega_3(0.40)$
200	2873.300	2873.602	2873.365	$\omega_6 + \omega_8(0.81)$, $\omega_2 + \omega_5 + \omega_8(0.37)$
201	2873.300	2873.638	2873.365	$\omega_6 + \omega_7(0.81)$, $\omega_1 + \omega_4 + \omega_7(0.37)$
202	2880.051	2880.354		$\omega_2 + \omega_4 + \omega_7(0.66)$, $\omega_2 + \omega_5 + \omega_8(0.52)$
203	2880.051	2880.374		$\omega_1 + \omega_5 + \omega_8(0.66)$, $\omega_1 + \omega_4 + \omega_7(0.52)$
204	2881.231	2881.559		$\omega_1 + \omega_5 + \omega_7(0.73)$, $\omega_1 + \omega_4 + \omega_8(0.40)$
205	2881.231	2881.574		$\omega_2 + \omega_4 + \omega_8(0.73)$, $\omega_2 + \omega_5 + \omega_7(0.40)$
206	2882.644	2882.945		$\omega_1 + \omega_4 + \omega_7(0.48)$, $\omega_2 + \omega_4 + \omega_8(0.48)$
207	2882.698	2883.024		$\omega_2 + \omega_4 + \omega_7(0.48)$, $\omega_1 + \omega_5 + \omega_7(0.48)$
208	2884.122	2884.416		$\omega_2 + \omega_5 + \omega_8(0.53)$, $\omega_1 + \omega_4 + \omega_8(0.51)$
209	2884.122	2884.443		$\omega_1 + \omega_4 + \omega_7(0.53)$, $\omega_2 + \omega_5 + \omega_7(0.51)$
210	2920.637	2920.840		$3\omega_1 + 5\omega_2(0.63)$, $5\omega_1 + 3\omega_2(0.63)$
211	2920.638	2920.840		$4\omega_1 + 4\omega_2(0.71)$, $2\omega_1 + 6\omega_2(0.45)$
212	2923.574	2923.798		$2\omega_1 + 6\omega_2(0.62)$, $6\omega_1 + 2\omega_2(0.62)$
213	2923.574	2923.826		$\omega_1 + 7\omega_2(0.49)$, $7\omega_1 + \omega_2(0.49)$
214	2925.674	2925.948		$\omega_1 + 7\omega_2(0.61)$, $7\omega_1 + \omega_2(0.61)$
215	2925.674	2925.953		$4\omega_1 + 4\omega_2(0.51)$, $8\omega_1(0.43)$
216	2926.934	2927.198		$8\omega_2(0.60)$, $8\omega_1(0.60)$
217	2926.934	2927.266		$5\omega_1 + 3\omega_2(0.48)$, $3\omega_1 + 5\omega_2(0.48)$
218	2927.354	2927.651		$8\omega_2(0.47)$, $8\omega_1(0.47)$
219	2934.730	2935.093		$2\omega_1 + 2\omega_2 + \omega_7(0.59)$, $3\omega_1 + \omega_2 + \omega_8(0.48)$
220	2934.730	2935.105		$2\omega_1 + 2\omega_2 + \omega_8(0.59)$, $3\omega_1 + \omega_2 + \omega_7(0.48)$
221	2935.184	2935.550		$3\omega_1 + \omega_2 + \omega_8(0.60)$, $2\omega_1 + 2\omega_2 + \omega_7(0.56)$
222	2935.186	2935.560		$\omega_1 + 3\omega_2 + \omega_7(0.59)$, $2\omega_1 + 2\omega_2 + \omega_8(0.56)$
223	2936.542	2936.958		$4\omega_2 + \omega_7(0.72)$, $\omega_1 + 3\omega_2 + \omega_8(0.40)$
224	2936.542	2936.965		$4\omega_1 + \omega_8(0.72)$, $\omega_1 + 3\omega_2 + \omega_7(0.40)$
225	2936.702	2937.095		$\omega_1 + 3\omega_2 + \omega_8(0.59)$, $4\omega_1 + \omega_7(0.53)$
226	2936.724	2937.125		$3\omega_1 + \omega_2 + \omega_7(0.59)$, $4\omega_2 + \omega_8(0.53)$
227	2937.644	2938.041		$4\omega_1 + \omega_7(0.75)$, $2\omega_1 + 2\omega_2 + \omega_7(0.40)$
228	2937.644	2938.064		$4\omega_2 + \omega_8(0.75)$, $2\omega_1 + 2\omega_2 + \omega_8(0.39)$
229	2947.207	2947.512	2947.052	$\omega_{10}(0.58)$, $2\omega_8(0.51)$
230	2950.918	2951.180		$\omega_3 + 2\omega_5(0.65)$, $\omega_3 + 2\omega_4(0.65)$

Table 6 (continued)

Frequency Number	$\Pi_{\mathbf{d}}(15\,000)$, $\varepsilon = 2.5\,10^{-4}$ (cm^{-1})	$\Pi_{\mathbf{d}}(12\,500)$, $\varepsilon = 0.005$ (cm^{-1})	Reference [1] (cm^{-1})	Assignment
231	2959.285	2959.487		$\omega_3 + 2\omega_4(0.66)$, $\omega_3 + 2\omega_5(0.66)$
232	2959.285	2959.661		$\omega_3 + \omega_4 + \omega_5(0.93)$, $2\omega_3 + \omega_4 + \omega_5(0.22)$
233	2967.878	2968.078	2967.959	$\omega_7 + \omega_8(0.96)$, $\omega_7 + \omega_8 + \omega_{10}(0.21)$
234	2967.879	2968.228	2967.959	$2\omega_7(0.68)$, $2\omega_8(0.68)$
235	2968.902	2969.108		$2\omega_1 + \omega_9(0.63)$, $2\omega_2 + \omega_9(0.63)$
236	2968.902	2969.162		$\omega_1 + \omega_2 + \omega_9(0.90)$, $\omega_1 + \omega_2 + \omega_3 + \omega_6(0.28)$
237	2969.544	2969.760		$2\omega_2 + \omega_9(0.63)$, $2\omega_1 + \omega_9(0.63)$
238	2981.009	2981.257	2980.774	$\omega_{10}(0.68)$, $2\omega_8(0.45)$

B.2   Ethylene oxide molecule

Table 7: Eigenvalues, frequencies and assignments for the first 200 eigenvalues of obtained by an A-VCI computation in $\Pi_{\mathbf{d}}(13\,500\,\mathrm{cm}^{-1})$ with a threshold of $\varepsilon = 3 \times 10^{-3}$. The final basis set contains $5\,328\,639$ elements

Number	Eigenvalue	Frequency	Assignment
GS	12461.474	0.000	$\omega_0(0.98)$, $\omega_9(0.13)$
1	13254.106	792.632	$\omega_1(0.97)$, $\omega_1 + \omega_9(0.14)$
2	13283.381	821.907	$\omega_2(0.96)$, $\omega_2 + \omega_9(0.16)$
3	13339.748	878.275	$\omega_3(0.96)$, $\omega_3 + \omega_9(0.17)$
4	13478.613	1017.140	$\omega_4(0.97)$, $\omega_4 + \omega_9(0.14)$
5	13582.643	1121.169	$\omega_6(0.96)$, $\omega_6 + \omega_9(0.14)$
6	13585.096	1123.623	$\omega_5(0.97)$, $\omega_5 + \omega_9(0.14)$
7	13607.198	1145.725	$\omega_7(0.97)$, $\omega_7 + \omega_9(0.15)$
8	13609.432	1147.958	$\omega_8(0.97)$, $\omega_8 + \omega_9(0.14)$
9	13732.256	1270.783	$\omega_9(0.94)$, $2\omega_9(0.22)$
10	13928.809	1467.336	$\omega_{10}(0.97)$, $\omega_9 + \omega_{10}(0.13)$
11	13956.634	1495.160	$\omega_{11}(0.94)$, $2\omega_1(0.20)$
12	14048.548	1587.074	$2\omega_1(0.94)$, $\omega_{11}(0.18)$
13	14072.382	1610.909	$\omega_1 + \omega_2(0.96)$, $\omega_1 + \omega_2 + \omega_9(0.16)$
14	14102.380	1640.906	$2\omega_2(0.92)$, $2\omega_2 + \omega_9(0.18)$
15	14131.644	1670.171	$\omega_1 + \omega_3(0.96)$, $\omega_1 + \omega_3 + \omega_9(0.18)$
16	14156.376	1694.903	$\omega_2 + \omega_3(0.94)$, $\omega_2 + \omega_3 + \omega_9(0.19)$
17	14216.146	1754.673	$2\omega_3(0.94)$, $2\omega_3 + \omega_9(0.21)$
18	14266.714	1805.241	$\omega_1 + \omega_4(0.96)$, $\omega_1 + \omega_4 + \omega_9(0.15)$
19	14293.552	1832.078	$\omega_2 + \omega_4(0.94)$, $\omega_2 + \omega_4 + \omega_9(0.17)$
20	14350.346	1888.872	$\omega_3 + \omega_4(0.91)$, $\omega_1 + \omega_5(0.27)$
21	14368.049	1906.576	$\omega_1 + \omega_5(0.90)$, $\omega_3 + \omega_4(0.28)$
22	14369.968	1908.495	$\omega_1 + \omega_6(0.93)$, $\omega_2 + \omega_4(0.14)$
23	14388.498	1927.024	$\omega_1 + \omega_8(0.82)$, $\omega_2 + \omega_6(0.45)$
24	14398.737	1937.263	$\omega_1 + \omega_7(0.86)$, $\omega_2 + \omega_5(0.43)$
25	14400.360	1938.886	$\omega_2 + \omega_6(0.81)$, $\omega_1 + \omega_8(0.47)$
26	14404.607	1943.134	$\omega_2 + \omega_5(0.85)$, $\omega_1 + \omega_7(0.43)$

Table 7 (continued)

Number	Eigenvalue	Frequency	Assignment
27	14422.567	1961.093	$\omega_2 + \omega_7(0.94), \omega_2 + \omega_7 + \omega_9(0.17)$
28	14431.332	1969.858	$\omega_2 + \omega_8(0.96), \omega_2 + \omega_8 + \omega_9(0.17)$
29	14458.131	1996.657	$\omega_3 + \omega_6(0.95), \omega_3 + \omega_6 + \omega_9(0.18)$
30	14459.815	1998.342	$\omega_3 + \omega_5(0.96), \omega_3 + \omega_5 + \omega_9(0.18)$
31	14481.886	2020.412	$\omega_3 + \omega_7(0.96), \omega_3 + \omega_7 + \omega_9(0.19)$
32	14484.815	2023.341	$\omega_3 + \omega_8(0.96), \omega_3 + \omega_8 + \omega_9(0.18)$
33	14492.455	2030.981	$2\omega_4(0.96), 2\omega_4 + \omega_9(0.16)$
34	14521.233	2059.760	$\omega_1 + \omega_9(0.93), \omega_1 + 2\omega_9(0.23)$
35	14547.330	2085.856	$\omega_2 + \omega_9(0.91), \omega_2 + 2\omega_9(0.25)$
36	14588.244	2126.771	$\omega_4 + \omega_6(0.92), \omega_4 + \omega_6 + \omega_9(0.15)$
37	14594.652	2133.178	$\omega_4 + \omega_5(0.93), \omega_4 + \omega_5 + \omega_9(0.15)$
38	14601.685	2140.211	$\omega_3 + \omega_9(0.90), \omega_3 + 2\omega_9(0.27)$
39	14612.482	2151.008	$\omega_4 + \omega_7(0.92), \omega_5 + \omega_6(0.22)$
40	14625.703	2164.230	$\omega_4 + \omega_8(0.96), \omega_4 + \omega_8 + \omega_9(0.15)$
41	14693.480	2232.006	$2\omega_6(0.83), 2\omega_5(0.41)$
42	14700.820	2239.346	$\omega_5 + \omega_6(0.91), \omega_4 + \omega_7(0.24)$
43	14706.736	2245.263	$\omega_1 + \omega_{10}(0.83), \omega_6 + \omega_8(0.41)$
44	14708.585	2247.111	$2\omega_5(0.86), 2\omega_6(0.43)$
45	14724.212	2262.738	$\omega_5 + \omega_8(0.70), \omega_1 + \omega_{11}(0.57)$
46	14727.200	2265.726	$\omega_5 + \omega_7(0.85), \omega_4 + \omega_9(0.36)$
47	14727.210	2265.736	$\omega_6 + \omega_7(0.92), \omega_5 + \omega_8(0.18)$
48	14734.658	2273.184	$\omega_6 + \omega_8(0.85), \omega_1 + \omega_{10}(0.43)$
49	14742.750	2281.276	$\omega_1 + \omega_{11}(0.67), \omega_5 + \omega_8(0.64)$
50	14749.045	2287.571	$\omega_2 + \omega_{10}(0.88), 2\omega_7(0.35)$
51	14750.250	2288.776	$\omega_4 + \omega_9(0.84), \omega_5 + \omega_7(0.40)$
52	14750.562	2289.089	$2\omega_7(0.83), \omega_2 + \omega_{10}(0.36)$
53	14753.334	2291.860	$\omega_7 + \omega_8(0.95), \omega_7 + \omega_8 + \omega_9(0.16)$
54	14755.939	2294.466	$2\omega_8(0.91), 2\omega_7(0.31)$
55	14771.517	2310.043	$\omega_2 + \omega_{11}(0.90), 3\omega_2(0.19)$
56	14806.374	2344.901	$\omega_3 + \omega_{10}(0.96), \omega_3 + \omega_9 + \omega_{10}(0.17)$
57	14831.168	2369.695	$\omega_3 + \omega_{11}(0.90), 2\omega_1 + \omega_3(0.19)$
58	14839.316	2377.843	$3\omega_1(0.88), \omega_1 + \omega_{11}(0.29)$
59	14849.458	2387.984	$\omega_6 + \omega_9(0.90), \omega_6 + 2\omega_9(0.22)$
60	14851.107	2389.634	$\omega_5 + \omega_9(0.92), \omega_5 + 2\omega_9(0.23)$
61	14862.045	2400.572	$2\omega_1 + \omega_2(0.92), 2\omega_1 + \omega_2 + \omega_9(0.16)$
62	14875.493	2414.019	$\omega_7 + \omega_9(0.92), \omega_7 + 2\omega_9(0.24)$
63	14876.364	2414.891	$\omega_8 + \omega_9(0.93), \omega_8 + 2\omega_9(0.23)$
64	14888.515	2427.042	$\omega_1 + 2\omega_2(0.90), \omega_1 + 2\omega_2 + \omega_9(0.18)$
65	14918.575	2457.102	$3\omega_2(0.85), \omega_2 + \omega_{11}(0.24)$
66	14924.962	2463.488	$2\omega_1 + \omega_3(0.92), 2\omega_1 + \omega_3 + \omega_9(0.18)$
67	14941.781	2480.307	$\omega_4 + \omega_{10}(0.96), \omega_4 + \omega_9 + \omega_{10}(0.14)$
68	14944.438	2482.964	$\omega_1 + \omega_2 + \omega_3(0.93), \omega_1 + \omega_2 + \omega_3 + \omega_9(0.20)$
69	14967.053	2505.579	$\omega_4 + \omega_{11}(0.92), 2\omega_1 + \omega_4(0.21)$
70	14970.762	2509.289	$2\omega_2 + \omega_3(0.86), 2\omega_2 + \omega_3 + \omega_9(0.20)$

Table 7 (continued)

Number	Eigenvalue	Frequency	Assignment
71	14998.054	2536.580	$2\omega_9(0.88)$, $3\omega_9(0.30)$
72	15007.201	2545.727	$\omega_1 + 2\omega_3(0.94)$, $\omega_1 + 2\omega_3 + \omega_9(0.21)$
73	15027.926	2566.452	$\omega_2 + 2\omega_3(0.90)$, $\omega_2 + 2\omega_3 + \omega_9(0.22)$
74	15043.649	2582.176	$\omega_6 + \omega_{10}(0.92)$, $\omega_5 + \omega_{11}(0.22)$
75	15047.520	2586.046	$\omega_5 + \omega_{10}(0.91)$, $\omega_6 + \omega_{11}(0.28)$
76	15053.109	2591.635	$2\omega_1 + \omega_4(0.89)$, $\omega_4 + \omega_{11}(0.19)$
77	15059.793	2598.319	$\omega_7 + \omega_{10}(0.85)$, $\omega_8 + \omega_{11}(0.39)$
78	15061.847	2600.373	$\omega_8 + \omega_{10}(0.86)$, $\omega_7 + \omega_{11}(0.38)$
79	15071.796	2610.322	$\omega_6 + \omega_{11}(0.74)$, $\omega_1 + \omega_2 + \omega_4(0.45)$
80	15075.991	2614.517	$\omega_1 + \omega_2 + \omega_4(0.79)$, $\omega_6 + \omega_{11}(0.47)$
81	15076.066	2614.593	$\omega_5 + \omega_{11}(0.89)$, $\omega_6 + \omega_{10}(0.24)$
82	15090.633	2629.159	$3\omega_3(0.92)$, $3\omega_3 + \omega_9(0.24)$
83	15101.083	2639.610	$\omega_8 + \omega_{11}(0.82)$, $\omega_7 + \omega_{10}(0.42)$
84	15101.469	2639.995	$\omega_7 + \omega_{11}(0.84)$, $\omega_8 + \omega_{10}(0.41)$
85	15105.885	2644.412	$2\omega_2 + \omega_4(0.87)$, $2\omega_2 + \omega_4 + \omega_9(0.18)$
86	15134.791	2673.318	$\omega_1 + \omega_3 + \omega_4(0.82)$, $2\omega_1 + \omega_5(0.42)$
87	15151.200	2689.726	$2\omega_1 + \omega_5(0.76)$, $\omega_1 + \omega_3 + \omega_4(0.46)$
88	15156.069	2694.595	$2\omega_1 + \omega_6(0.86)$, $\omega_1 + \omega_2 + \omega_4(0.22)$
89	15160.323	2698.849	$\omega_2 + \omega_3 + \omega_4(0.88)$, $\omega_1 + \omega_2 + \omega_5(0.22)$
90	15169.040	2707.567	$2\omega_1 + \omega_8(0.76)$, $\omega_1 + \omega_2 + \omega_6(0.43)$
91	15179.362	2717.889	$\omega_1 + \omega_2 + \omega_5(0.84)$, $2\omega_1 + \omega_7(0.26)$
92	15186.772	2725.299	$\omega_1 + \omega_2 + \omega_6(0.77)$, $2\omega_1 + \omega_8(0.45)$
93	15193.776	2732.302	$2\omega_1 + \omega_7(0.89)$, $\omega_1 + \omega_2 + \omega_5(0.28)$
94	15197.648	2736.175	$\omega_9 + \omega_{10}(0.93)$, $2\omega_9 + \omega_{10}(0.22)$
95	15202.616	2741.143	$\omega_1 + \omega_2 + \omega_8(0.66)$, $2\omega_2 + \omega_6(0.58)$
96	15210.689	2749.215	$\omega_1 + \omega_2 + \omega_7(0.87)$, $2\omega_2 + \omega_5(0.30)$
97	15216.441	2754.967	$\omega_1 + \omega_2 + \omega_8(0.64)$, $2\omega_2 + \omega_6(0.62)$
98	15219.835	2758.361	$2\omega_2 + \omega_5(0.85)$, $\omega_1 + \omega_2 + \omega_7(0.33)$
99	15220.620	2759.147	$2\omega_3 + \omega_4(0.88)$, $\omega_1 + \omega_3 + \omega_5(0.31)$
100	15223.444	2761.970	$\omega_9 + \omega_{11}(0.88)$, $2\omega_9 + \omega_{11}(0.22)$
101	15234.937	2773.464	$2\omega_2 + \omega_7(0.88)$, $2\omega_2 + \omega_7 + \omega_9(0.19)$
102	15242.891	2781.418	$\omega_1 + \omega_3 + \omega_5(0.87)$, $2\omega_3 + \omega_4(0.31)$
103	15244.243	2782.769	$\omega_1 + \omega_3 + \omega_6(0.91)$, $\omega_1 + \omega_3 + \omega_6 + \omega_9(0.18)$
104	15249.888	2788.414	$2\omega_2 + \omega_8(0.91)$, $2\omega_2 + \omega_8 + \omega_9(0.19)$
105	15262.167	2800.693	$\omega_1 + \omega_3 + \omega_8(0.73)$, $\omega_2 + \omega_3 + \omega_6(0.54)$
106	15270.730	2809.256	$\omega_2 + \omega_3 + \omega_5(0.72)$, $\omega_1 + \omega_3 + \omega_7(0.59)$
107	15272.124	2810.650	$\omega_2 + \omega_3 + \omega_6(0.71)$, $\omega_1 + \omega_3 + \omega_8(0.57)$
108	15273.665	2812.191	$\omega_1 + 2\omega_4(0.92)$, $\omega_1 + 2\omega_4 + \omega_9(0.16)$
109	15275.704	2814.230	$\omega_1 + \omega_3 + \omega_7(0.74)$, $\omega_2 + \omega_3 + \omega_5(0.58)$
110	15292.354	2830.880	$\omega_2 + \omega_3 + \omega_7(0.90)$, $\omega_2 + \omega_3 + \omega_7 + \omega_9(0.20)$
111	15299.704	2838.230	$\omega_2 + 2\omega_4(0.91)$, $\omega_1 + \omega_4 + \omega_6(0.18)$
112	15301.384	2839.910	$\omega_2 + \omega_3 + \omega_8(0.93)$, $\omega_2 + \omega_3 + \omega_8 + \omega_9(0.20)$
113	15311.327	2849.854	$2\omega_1 + \omega_9(0.87)$, $2\omega_1 + 2\omega_9(0.22)$
114	15331.701	2870.227	$2\omega_3 + \omega_6(0.92)$, $2\omega_3 + \omega_6 + \omega_9(0.21)$

Table 7 (continued)

Number	Eigenvalue	Frequency	Assignment
115	15332.157	2870.683	$\omega_1 + \omega_2 + \omega_9(0.89)$, $\omega_1 + \omega_2 + 2\omega_9(0.26)$
116	15332.595	2871.121	$2\omega_3 + \omega_5(0.93)$, $2\omega_3 + \omega_5 + \omega_9(0.21)$
117	15354.656	2893.182	$2\omega_3 + \omega_7(0.93)$, $2\omega_3 + \omega_7 + \omega_9(0.22)$
118	15357.436	2895.962	$\omega_3 + 2\omega_4(0.84)$, $\omega_1 + \omega_4 + \omega_5(0.37)$
119	15358.166	2896.692	$2\omega_3 + \omega_8(0.93)$, $2\omega_3 + \omega_8 + \omega_9(0.22)$
120	15360.828	2899.355	$2\omega_2 + \omega_9(0.83)$, $2\omega_2 + 2\omega_9(0.27)$
121	15367.786	2906.312	$\omega_{12}(0.62)$, $\omega_{10} + \omega_{11}(0.49)$
122	15371.685	2910.211	$\omega_1 + \omega_4 + \omega_5(0.64)$, $\omega_3 + 2\omega_4(0.35)$
123	15372.337	2910.863	$\omega_1 + \omega_4 + \omega_6(0.72)$, $\omega_{10} + \omega_{11}(0.35)$
124	15378.145	2916.671	$2\omega_{10}(0.64)$, $\omega_{13}(0.47)$
125	15389.839	2928.365	$\omega_1 + \omega_3 + \omega_9(0.88)$, $\omega_1 + \omega_3 + 2\omega_9(0.27)$
126	15392.130	2930.656	$\omega_2 + \omega_4 + \omega_6(0.69)$, $\omega_1 + \omega_4 + \omega_8(0.50)$
127	15397.446	2935.972	$\omega_1 + \omega_4 + \omega_7(0.74)$, $\omega_2 + \omega_4 + \omega_5(0.45)$
128	15404.363	2942.889	$\omega_1 + \omega_4 + \omega_8(0.76)$, $\omega_2 + \omega_4 + \omega_6(0.51)$
129	15407.391	2945.917	$\omega_2 + \omega_4 + \omega_5(0.76)$, $\omega_1 + \omega_4 + \omega_7(0.51)$
130	15412.967	2951.494	$\omega_2 + \omega_3 + \omega_9(0.86)$, $\omega_2 + \omega_3 + 2\omega_9(0.29)$
131	15413.868	2952.394	$2\omega_{10}(0.61)$, $\omega_{13}(0.51)$
132	15424.388	2962.915	$\omega_2 + \omega_4 + \omega_7(0.79)$, $\omega_{13}(0.25)$
133	15440.016	2978.542	$\omega_2 + \omega_4 + \omega_8(0.92)$, $\omega_2 + \omega_4 + \omega_8 + \omega_9(0.18)$
134	15450.720	2989.247	$\omega_{10} + \omega_{11}(0.63)$, $\omega_{12}(0.52)$
135	15455.329	2993.856	$2\omega_{11}(0.71)$, $2\omega_1 + \omega_{11}(0.37)$
136	15457.443	2995.970	$\omega_3 + \omega_4 + \omega_6(0.87)$, $\omega_1 + \omega_5 + \omega_6(0.19)$
137	15459.743	2998.270	$\omega_3 + \omega_4 + \omega_5(0.64)$, $\omega_1 + 2\omega_5(0.50)$
138	15468.361	3006.887	$\omega_3 + \omega_4 + \omega_5(0.63)$, $\omega_1 + 2\omega_6(0.49)$
139	15469.680	3008.206	$2\omega_3 + \omega_9(0.85)$, $2\omega_3 + 2\omega_9(0.31)$
140	15475.932	3014.458	$\omega_1 + \omega_5 + \omega_6(0.82)$, $\omega_1 + \omega_4 + \omega_7(0.23)$
141	15482.167	3020.694	$\omega_3 + \omega_4 + \omega_7(0.88)$, $\omega_3 + \omega_5 + \omega_6(0.20)$
142	15486.806	3025.332	$\omega_{14}(0.85)$, $\omega_{13} + \omega_{14}(0.20)$
143	15487.621	3026.147	$\omega_1 + 2\omega_5(0.64)$, $\omega_1 + 2\omega_6(0.62)$
144	15491.762	3030.289	$\omega_1 + \omega_6 + \omega_8(0.54)$, $2\omega_1 + \omega_{10}(0.53)$
145	15492.388	3030.915	$\omega_3 + \omega_4 + \omega_8(0.72)$, $\omega_1 + \omega_5 + \omega_8(0.53)$
146	15498.388	3036.914	$\omega_{15}(0.80)$, $\omega_1 + 2\omega_8(0.26)$
147	15499.312	3037.838	$\omega_2 + 2\omega_6(0.72)$, $2\omega_1 + \omega_{10}(0.38)$
148	15499.969	3038.495	$\omega_3 + \omega_4 + \omega_8(0.59)$, $\omega_1 + \omega_5 + \omega_8(0.58)$
149	15500.793	3039.319	$3\omega_4(0.93)$, $3\omega_4 + \omega_9(0.16)$
150	15507.666	3046.192	$\omega_1 + \omega_5 + \omega_7(0.80)$, $\omega_1 + \omega_4 + \omega_9(0.28)$
151	15509.516	3048.042	$\omega_2 + \omega_5 + \omega_6(0.72)$, $\omega_1 + \omega_6 + \omega_7(0.35)$
152	15512.803	3051.330	$\omega_1 + \omega_6 + \omega_7(0.83)$, $\omega_2 + \omega_5 + \omega_6(0.24)$
153	15517.932	3056.458	$\omega_1 + \omega_6 + \omega_8(0.62)$, $2\omega_1 + \omega_{10}(0.51)$
154	15520.625	3059.151	$2\omega_1 + \omega_{11}(0.54)$, $\omega_1 + \omega_5 + \omega_8(0.41)$
155	15522.167	3060.693	$\omega_1 + \omega_2 + \omega_{10}(0.68)$, $\omega_2 + \omega_6 + \omega_8(0.44)$
156	15522.842	3061.369	$\omega_2 + 2\omega_5(0.85)$, $\omega_2 + 2\omega_6(0.33)$
157	15526.483	3065.009	$\omega_1 + \omega_7 + \omega_8(0.66)$, $\omega_2 + \omega_6 + \omega_7(0.58)$
158	15526.976	3065.503	$\omega_1 + 2\omega_8(0.75)$, $\omega_1 + \omega_2 + \omega_{10}(0.40)$

Table 7 (continued)

Number	Eigenvalue	Frequency	Assignment
159	15533.150	3071.676	$\omega_1 + \omega_4 + \omega_9(0.83)$, $\omega_1 + \omega_5 + \omega_7(0.34)$
160	15537.494	3076.020	$\omega_1 + \omega_2 + \omega_{11}(0.57)$, $\omega_2 + \omega_5 + \omega_8(0.49)$
161	15538.773	3077.299	$\omega_2 + \omega_5 + \omega_7(0.69)$, $\omega_1 + 2\omega_7(0.40)$
162	15542.509	3081.035	$\omega_1 + \omega_7 + \omega_8(0.60)$, $\omega_2 + \omega_6 + \omega_7(0.51)$
163	15543.705	3082.232	$\omega_1 + 2\omega_7(0.85)$, $\omega_2 + \omega_5 + \omega_7(0.28)$
164	15551.329	3089.855	$\omega_2 + \omega_6 + \omega_8(0.79)$, $\omega_1 + \omega_2 + \omega_{10}(0.41)$
165	15557.571	3096.098	$\omega_2 + \omega_5 + \omega_8(0.75)$, $\omega_1 + \omega_2 + \omega_{11}(0.52)$
166	15559.373	3097.899	$\omega_2 + 2\omega_7(0.90)$, $\omega_2 + 2\omega_7 + \omega_9(0.18)$
167	15559.907	3098.433	$\omega_2 + \omega_4 + \omega_9(0.76)$, $\omega_2 + \omega_5 + \omega_7(0.46)$
168	15565.653	3104.179	$\omega_3 + 2\omega_6(0.79)$, $\omega_3 + 2\omega_5(0.44)$
169	15567.112	3105.639	$2\omega_2 + \omega_{10}(0.90)$, $2\omega_2 + \omega_9 + \omega_{10}(0.17)$
170	15568.263	3106.789	$\omega_2 + \omega_7 + \omega_8(0.92)$, $\omega_2 + \omega_7 + \omega_8 + \omega_9(0.18)$
171	15572.098	3110.624	$\omega_3 + \omega_5 + \omega_6(0.88)$, $\omega_3 + \omega_4 + \omega_7(0.23)$
172	15577.098	3115.624	$\omega_2 + 2\omega_8(0.93)$, $\omega_2 + 2\omega_8 + \omega_9(0.18)$
173	15579.734	3118.260	$\omega_3 + 2\omega_5(0.83)$, $\omega_3 + 2\omega_6(0.45)$
174	15582.380	3120.906	$\omega_1 + \omega_3 + \omega_{10}(0.77)$, $\omega_3 + \omega_6 + \omega_8(0.43)$
175	15585.225	3123.751	$2\omega_2 + \omega_{11}(0.82)$, $4\omega_2(0.24)$
176	15588.303	3126.830	$2\omega_4 + \omega_6(0.87)$, $\omega_1 + \omega_4 + \omega_5(0.18)$
177	15596.549	3135.076	$\omega_3 + \omega_5 + \omega_7(0.76)$, $\omega_3 + \omega_4 + \omega_9(0.43)$
178	15597.261	3135.787	$\omega_3 + \omega_5 + \omega_8(0.71)$, $\omega_1 + \omega_3 + \omega_{11}(0.51)$
179	15598.807	3137.333	$2\omega_4 + \omega_5(0.87)$, $\omega_4 + \omega_6 + \omega_7(0.17)$
180	15599.090	3137.616	$\omega_3 + \omega_6 + \omega_7(0.89)$, $\omega_3 + \omega_5 + \omega_8(0.21)$
181	15607.750	3146.277	$\omega_3 + \omega_6 + \omega_8(0.79)$, $\omega_1 + \omega_3 + \omega_{10}(0.49)$
182	15613.910	3152.436	$2\omega_4 + \omega_7(0.85)$, $\omega_4 + \omega_5 + \omega_6(0.25)$
183	15614.965	3153.491	$\omega_3 + \omega_4 + \omega_9(0.69)$, $\omega_3 + \omega_5 + \omega_7(0.50)$
184	15615.813	3154.339	$\omega_1 + \omega_3 + \omega_{11}(0.68)$, $\omega_3 + \omega_5 + \omega_8(0.57)$
185	15621.233	3159.759	$\omega_3 + 2\omega_7(0.70)$, $\omega_2 + \omega_3 + \omega_{10}(0.58)$
186	15621.722	3160.249	$\omega_2 + \omega_3 + \omega_{10}(0.68)$, $\omega_3 + 2\omega_7(0.57)$
187	15622.426	3160.952	$4\omega_1(0.75)$, $2\omega_1 + \omega_{11}(0.38)$
188	15624.986	3163.512	$\omega_3 + \omega_7 + \omega_8(0.93)$, $\omega_3 + \omega_7 + \omega_8 + \omega_9(0.20)$
189	15628.101	3166.628	$\omega_3 + 2\omega_8(0.91)$, $\omega_3 + 2\omega_7(0.23)$
190	15629.515	3168.042	$\omega_1 + \omega_5 + \omega_9(0.84)$, $\omega_3 + \omega_4 + \omega_9(0.24)$
191	15633.060	3171.587	$\omega_1 + \omega_6 + \omega_9(0.86)$, $\omega_1 + \omega_6 + 2\omega_9(0.22)$
192	15637.449	3175.976	$2\omega_4 + \omega_8(0.94)$, $2\omega_4 + \omega_8 + \omega_9(0.17)$
193	15641.670	3180.196	$\omega_2 + \omega_3 + \omega_{11}(0.85)$, $\omega_2 + \omega_3 + \omega_9 + \omega_{11}(0.18)$
194	15646.530	3185.057	$3\omega_1 + \omega_2(0.84)$, $\omega_1 + \omega_2 + \omega_{11}(0.25)$
195	15651.151	3189.677	$\omega_1 + \omega_8 + \omega_9(0.71)$, $\omega_2 + \omega_6 + \omega_9(0.48)$
196	15660.065	3198.592	$\omega_2 + \omega_5 + \omega_9(0.74)$, $\omega_1 + \omega_7 + \omega_9(0.48)$
197	15660.707	3199.233	$\omega_2 + \omega_6 + \omega_9(0.69)$, $\omega_1 + \omega_8 + \omega_9(0.53)$
198	15665.851	3204.377	$\omega_1 + \omega_7 + \omega_9(0.75)$, $\omega_2 + \omega_5 + \omega_9(0.48)$
199	15674.352	3212.878	$2\omega_1 + 2\omega_2(0.83)$, $2\omega_1 + \omega_{11}(0.20)$

References

[1] G. Avila and T. Carrington Jr. Using nonproduct quadrature grids to solve the vibrational schrödinger equation in 12d. *J. Chem. Phys.*, 134(5):054126, 2011.

[2] Gustavo Avila and Tucker Carrington. Solving the vibrational schrödinger equation using bases pruned to include strongly coupled functions and compatible quadratures. *The Journal of chemical physics*, 137(17):174108, November 2012.

[3] Gustavo Avila and Tucker Carrington. Pruned bases that are compatible with iterative eigensolvers and general potentials: New results for CH3CN. *Chemical Physics*, pages 1–16, 2016.

[4] Gustavo Avila and Tucker Carrington Jr. Solving the schrödinger equation using smolyak interpolants. *The Journal of Chemical Physics*, 139(13):134114, 2013.

[5] I Baraille, C Larrieu, A Dargelos, and M Chaillet. Calculation of non-fundamental {IR} frequencies and intensities at the anharmonic level. i. the overtone, combination and difference bands of diazomethane, {H2CN2}. *Chemical Physics*, 273(2–3):91–101, 2001.

[6] F.L. Bauer and C.T. Fike. Norms and exclusion theorems. *Numer. Math*, 2(1):137–141, 1960.

[7] D. Begue, P. Carbonniere, and C. Pouchan. Calculations of vibrational energy levels by using a hybrid ab initio and dft quartic force field: Application to acetonitrile. *J. Phys. Chem. A*, 109(20):4611–4616, 2005.

[8] D. Bégué, N. Gohaud, C. Pouchan, P. Cassam-Chenai, and J. Liévin. A comparison of two methods for selecting vibrational configuration interaction spaces on a heptatomic system: Ethylene oxide. *J. Chem. Phys.*, 127(16):164115, 2007.

[9] J.M. Bowman, T. Carrington, and H.-D. Meyer. Variational quantum approaches for computing vibrational energies of polyatomic molecules. *Mol. Phys.*, 106(16-18):2145–2182, 2008.

[10] J.M. Bowman, K. Christoffel, and F. Tobin. Application of scf-si theory to vibrational motion in polyatomic molecules. *J. Phys. Chem.*, 83(8):905–920, 1979.

[11] Matthew J. Bramley and Tucker Carrington Jr. A general discrete variable method to calculate vibrational energy levels of three and four atom molecules. *The Journal of Chemical Physics*, 99(11):8519–8541, 1993.

[12] Matthew J. Bramley, John W. Tromp, Tucker Carrington Jr., and Gregory C. Corey. Efficient calculation of highly excited vibrational energy levels of floppy molecules: The band origins of h+3 up to 35 000 cm^{-1}. *The Journal of Chemical Physics*, 100(9):6175–6194, 1994.

[13] James Brown and Tucker Carrington. Using an expanding nondirect product harmonic basis with an iterative eigensolver to compute vibrational energy levels with as many as seven atoms. *The Journal of Chemical Physics*, 145(14):144104, 2016.

[14] James Brown and Tucker Carrington Jr. Using an iterative eigensolver to compute vibrational energies with phase-spaced localized basis functions. *The Journal of Chemical Physics*, 143(4):044104, 2015.

[15] S. Carter and N.C. Handy. The variational method for the calculation of ro-vibrational energy levels. *Computer Physics Reports*, 5(3):117 – 171, 1986.

[16] Stuart Carter, Joel M. Bowman, and Amit R. Sharma. The multimode approach to ro-vibrational spectroscopy. *AIP Conference Proceedings*, 1504(1):465–466, 2012.

[17] Stuart Carter, Susan J. Culik, and Joel M. Bowman. Vibrational self-consistent field method for many-mode systems: A new approach and application to the vibrations of co adsorbed on cu(100). *The Journal of Chemical Physics*, 107(24):10458–10469, 1997.

[18] P. Cassam-Chenaï and J. Liévin. Alternative perturbation method for the molecular vibration-rotation problem. *International Journal of Quantum Chemistry*, 93(3):245–264, 2003.

[19] Johnny Chang and Robert E. Wyatt. Influence of rotational states on molecular multiphoton excitation. *The Journal of Chemical Physics*, 85(4):1840–1847, 1986.

[20] Jason Cooper and Tucker Carrington. Computing vibrational energy levels by using mappings to fully exploit the structure of a pruned product basis. *The Journal of Chemical Physics*, 130(21):214110, 2009.

[21] Richard Dawes and Tucker Carrington Jr. How to choose one-dimensional basis functions so that a very efficient multidimensional basis may be extracted from a direct product of the one-dimensional functions: Energy levels of coupled systems with as many as 16 coordinates. *The Journal of Chemical Physics*, 122(13):134101, 2005.

[22] Richard Dawes and Tucker Carrington Jr. Using simultaneous diagonalization and trace minimization to make an efficient and simple multidimensional basis for solving the vibrational schrödinger equation. *The Journal of Chemical Physics*, 124(5):054102, 2006.

[23] Romain Garnier, Marc Odunlami, Vincent Le Bris, Didier Bégué, Isabelle Baraille, and Olivier Coulaud. Adaptive vibrational configuration interaction (a-vci): A posteriori error estimation to efficiently compute anharmonic ir spectra. *J. Chem. Phys.*, 144(20):204123, 2016.

[24] L. Halonen, D. W. Noid, and M.S. Child. Local mode predictions for excited stretching vibrational states of hccd and h12c13ch. *The Journal of Chemical Physics*, 78(5):2803–2804, 1983.

[25] Thomas Halverson and Bill Poirier. Large scale exact quantum dynamics calculations: Ten thousand quantum states of acetonitrile. *Chemical Physics Letters*, 624:37–42, 2015.

[26] Thomas Halverson and Bill Poirier. One million quantum states of benzene. *The Journal of Physical Chemistry A*, 119(50):12417–12433, 2015. PMID: 26418314.

[27] Bernd Hartke. Propagation with distributed gaussians as a sparse, adaptive basis for higher-dimensional quantum dynamics. *Phys. Chem. Chem. Phys.*, 8:3627–3635, 2006.

[28] Roger A. Horn and Charles R. Johnson. *Topics in matrix analysis*. Cambridge University Press, Cambridge, New York, Melbourne, 1994.

[29] Michael E. Kellman and Vivian Tyng. The dance of molecules: New dynamical perspectives on highly excited molecular vibrations. *Accounts of Chemical Research*, 40(4):243–250, 2007. PMID: 17256881.

[30] A. Leclerc and T. Carrington. Calculating vibrational spectra with sum of product basis functions without storing full-dimensional vectors or matrices. *J. Chem. Phys.*, 140(17):174111, 2014.

[31] R B Lehoucq, D C Sorensen, and C Yang. ARPACK Users ' Guide : Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods . *Communication*, 6:147, 1998.

[32] John C Light and Tucker Carrington Jr. Discrete-variable representations and their utilization. *Advances in Chemical Physics*, 114:263–310, 2000.

[33] R. C. Lord and B. Nolin. Vibrational spectra of ethylene oxide and ethylene oxide-d$_4$. *The Journal of Chemical Physics*, 24:656–658, apr 1956.

[34] Shai Machnes, Elie Assémat, Henrik R. Larsson, and David J. Tannor. Quantum dynamics in phase space using projected von neumann bases. *The Journal of Physical Chemistry A*, 120(19):3296–3308, 2016. PMID: 26977715.

[35] U. Manthe and H. Köppel. New method for calculating wave packet dynamics: Strongly coupled surfaces and the adiabatic basis. *The Journal of Chemical Physics*, 93(1):345–356, 1990.

[36] Michael Neff and Guntram Rauhut. Toward large scale vibrational configuration interaction calculations. *The Journal of Chemical Physics*, 131(12), 2009.

[37] Bill Poirier. Using wavelets to extend quantum dynamics calculations to ten or more degrees of freedom. *Journal of Theoretical and Computational Chemistry*, 02(01):65–72, 2003.

[38] Claude Pouchan and Khalil Zaki. Ab initio configuration interaction determination of the overtone vibrations of methyleneimine in the region 2800-3200 cm-1. *The Journal of Chemical Physics*, 107(2):342–345, 1997.

[39] Cristina Puzzarini, Malgorzata Biczysko, Julien Bloino, and Vincenzo Barone. Accurate Spectroscopic Characterization of Oxirane: a Valuable Route To Its Identification in Titan'S Atmosphere and the Assignment of Unidentified Infrared Bands. *The Astrophysical journal*, 785(2), 2014.

[40] Martin Quack. Spectra and dynamics of coupled vibrations in polyatomic molecules. *Annual Review of Physical Chemistry*, 41(1):839–874, 1990.

[41] Guntram Rauhut. Configuration selection as a route towards efficient vibrational configuration interaction calculations. *The Journal of Chemical Physics*, 127(18), 2007.

[42] Hubert Romanowski, Joel M. Bowman, and Lawrence B. Harding. Vibrational energy levels of formaldehyde. *The Journal of Chemical Physics*, 82(9):4155–4165, 1985.

[43] Yohann Scribano and David M. Benoit. Iterative active-space selection for vibrational configuration interaction calculations using a reduced-coupling {VSCF} basis. *Chemical Physics Letters*, 458(4-6):384 – 387, 2008.

[44] Asaf Shimshovitz, Zlatko Bačič, and David J. Tannor. The von neumann basis in non-cartesian coordinates: Application to floppy triatomic molecules. *The Journal of Chemical Physics*, 141(23), 2014.

[45] J. Sielk, H. F. von Horsten, F. Kruger, R. Schneider, and B. Hartke. Quantum-mechanical wavepacket propagation in a sparse, adaptive basis of interpolating gaussians with collocation. *Phys. Chem. Chem. Phys.*, 11:463–475, 2009.

[46] Phillip S. Thomas and T. Carrington Jr. Using nested contractions and a hierarchical tensor format to compute vibrational spectra of molecules with seven atoms. *J. Phys. Chem. A*, 119(52):13074–13091, 2015.

[47] Alexandra Viel and Claude Leforestier. Six-dimensional calculation of the vibrational spectrum of the hfco molecule. *The Journal of Chemical Physics*, 112(3):1212–1220, 2000.

[48] Xiao-Gang Wang and Tucker Carrington. The utility of constraining basis function indices when using the lanczos algorithm to calculate vibrational energy levels. *The Journal of Physical Chemistry A*, 105(12):2575–2581, 2001.

[49] R. M. Wilcox. Matrix Elements of General Potentials in the HarmonicOscillator Representation. *The Journal of Chemical Physics*, 45(1966):3312, 1966.

[50] Hua-Gen Yu. An exact variational method to calculate vibrational energies of five atom molecules beyond the normal mode approach. *The Journal of Chemical Physics*, 117(5):2030–2037, 2002.