A symmetry-adapted Lanczos method for calculating energy levels with different symmetries from a single set of iterations

Xiao-Gang Wang and Tucker Carrington, Jr.a)
Département de Chimie, Université de Montréal, Case Postal 6128, succursale Centre-ville, Montréal (Québec) H3C 3J7, Canada

(Received 16 August 2000; accepted 17 October 2000)

We present a symmetry-adapted Lanczos method that uses projection operators to calculate energy levels with different symmetries from a single sequence of matrix–vector products. Compared with the conventional Lanczos method, this method has the advantage that energy levels are computed more efficiently and with symmetry labels. High-lying stretching energy levels of a coupled Morse oscillator Hamiltonian describing $^{28}\text{SiH}_4$ are calculated to demonstrate the advantage of the symmetry-adapted method. © 2001 American Institute of Physics. [DOI: 10.1063/1.1331357]

I. INTRODUCTION

Several new techniques enable experimentalists to accurately determine high-lying energy levels of small molecules. Both the development of better ab initio methods and the availability of faster computers with more memory has permitted ab initio quantum chemists to determine potential energy surfaces of useful accuracy close to the minimum. The experimental and ab initio advances drive the development of methods for calculating vibrational and rovibrational energy levels of small polyatomic molecules.1–3 Almost invariably energy levels are calculated by computing eigenvalues of a large matrix representation of the Hamiltonian operator. The principal impediment to the calculation of energy levels is the size of the matrix required to obtain converged results. Large matrices cause trouble for two reasons. First, standard, explicit, diagonalization algorithms modify the matrix and therefore require that it be stored in the core memory of the computer. Second, the cost of explicit diagonalization scales as $N^3$, where $N$ is the size of the matrix. Even when it is possible to store a large matrix, calculating its eigenvalues is costly. No matter how clever one is at choosing basis functions to represent wave functions compactly, it is clear that the size of the Hamiltonian matrix is a serious obstacle.

The alternative to using an explicit diagonalization algorithm is to use an iterative method to compute some of the eigenvalues of the huge Hamiltonian matrix. The simplest iterative method for calculating eigenvalues is the Lanczos algorithm.4 It has been applied successfully to calculate vibrational and rovibrational energy levels using a variety of coordinates for molecules with three and four atoms (including all degrees of freedom) and molecules with more than four atoms (including some degrees of freedom).5–13 Using the Lanczos algorithm one computes eigenvalues of a matrix by evaluating $M_L$ matrix–vector products to obtain a $M_L \times M_L$ tridiagonal matrix. Among the eigenvalues of the tridiagonal matrix are eigenvalues of the original matrix.

If one uses an explicit diagonalization algorithm to calculate energy levels it is straightforward to take advantage of symmetry. By choosing basis functions that transform as irreducible representations of the symmetry group of the molecule one can calculate energy levels of different symmetries separately and therefore reduce the size of the matrix required to obtain converged energy levels. Using symmetry-adapted basis functions has two important advantages: (1) symmetry labels are automatically obtained for the energy levels; (2) because the size of the matrix diagonalized for a single irreducible representation is smaller than the size of the matrix one would diagonalize if symmetry were not exploited, less memory is required to calculate energy levels and the cost of the calculation is reduced.

If one uses the Lanczos algorithm to calculate energy levels it is not as easy to exploit symmetry to reduce the cost of the calculation. One wishes to take advantage of symmetry to obtain symmetry labels and to reduce the CPU time (and perhaps) the memory required to obtain converged energy levels. The number of matrix–vector products required to compute converged energy levels using the Lanczos algorithm depends on the spacing of the energy levels. If degenerate or nearly degenerate energy levels can be relegated to different matrices (or blocks) the number of matrix–vector products required can be greatly reduced.

The most obvious way to use the Lanczos algorithm and exploit symmetry is to construct symmetry adapted basis functions, i.e., to use the symmetry adapted basis functions one would use with an explicit diagonalization method.14 This is often not the best way to exploit symmetry because the matrix–vector products one must evaluate to use the Lanczos algorithm are most efficient if the basis is a product basis.9,15,16 A product basis is a basis each of whose functions is a product of functions of a single variable. Because symmetry-adapted basis functions are frequently not product basis functions it is difficult to efficiently evaluate matrix–vector products with a symmetry-adapted basis. As discussed by Bramley and Carrington,9 symmetry-adapted basis functions can be used efficiently if a symmetry operation affects only one coordinate. However, symmetry-adapted basis functions for a symmetry operation that mixes or exchanges coordinates are linear combinations of the original product.
basis functions and the corresponding matrix–vector product is difficult to evaluate efficiently. For some molecules it is possible to define coordinates so that each symmetry operation affects only one coordinate. In this fashion we used the Lanczos algorithm while taking full advantage of the symmetry of H₂O to calculate its vibrational energy levels from a kinetic energy operator in symmetrized Radau coordinates.¹¹

For many molecules of interest it is not possible to choose coordinates so that each symmetry operation affects only one coordinate. How can one then exploit symmetry to label energy levels and accelerate their calculation? Is it possible to take advantage of symmetry without relinquishing the efficiency of matrix–vector products computed with sequential summation for a product basis? Bramley and Carrington⁹ proposed using as the Lanczos starting vector a symmetrized combination of unsymmetrized vectors to calculate only energy levels with the same symmetry as the starting vector. If the number of iterations is small one obtains only energy levels of the same symmetry as the starting vector, but if the number of iterations is large enough, energy levels that transform like different irreducible representations are also converged. These “wrong symmetry” levels are converged because of the finite precision of computer arithmetic. In Ref. 17 we developed a method for retaining energy levels whose wave functions have the same symmetry as the starting vector and discarded the wrong symmetry levels. In some cases one must be careful to avoid discarding energy levels with the same symmetry as the starting vector but whose wave functions have very small overlaps with the starting vector.

Chen and Guo⁸⁻¹⁰ developed a symmetry-adapted filter diagonalization method (SAFD). From a single set of matrix–vector products they simultaneously compute correlation functions for all of the irreducible representations. By analyzing (separately) the symmetry-adapted correlation functions they obtain symmetry-labeled energy levels. If energy levels are closely spaced then using filter diagonalization one requires more matrix–vector products to generate a correlation function from which they can be accurately extracted. Computing separate correlation functions for each symmetry block therefore reduces the number of matrix–vector products required to obtain converged energy levels. The purpose of this paper is to develop a Lanczos method for calculating energy levels of different symmetries separately, but from one set of matrix–vector products.

II. THEORY

The conventional Lanczos iteration is based on a simple three-term recurrence relation

\[ \beta_{i+1}v_{i+1} = Hv_i - \alpha_i v_i - \beta_i v_{i-1}. \]  

with \( \beta_i = 0 \). In exact arithmetic the Lanczos vectors, \( v_i \), calculated using Eq. (1) are orthogonal and the Hamiltonian matrix in the basis of the Lanczos vectors is tridiagonal. The \( \beta_i \) and \( \alpha_i \) are elements of the tridiagonal matrix. If one requires more than a small number of iterations the orthogonality of the Lanczos vectors is lost due to the finite precision of computer arithmetic. However, it is still possible, using a scheme suggested by Cullum and Willoughby, to extract accurate eigenvalues from the tridiagonal matrix.⁴

We develop a symmetry-adapted Lanczos (SAL) method with which we calculate Lanczos vectors \( v^{(m)}_i \), and \( \alpha^{(m)}_i \) and \( \beta^{(m)}_i \) coefficients for all irreducible representations \( m \) from a single set of matrix–vector products. Each set of Lanczos vectors is computed from the usual three-term recurrence relation

\[ \beta^{(m)}_{i+1} v^{(m)}_{i+1} = H v^{(m)}_i - \alpha^{(m)}_i v^{(m)}_i - \beta^{(m)}_i v^{(m)}_{i-1}, \]  

but rather than applying \( H \) to symmetry-adapted Lanczos vectors to compute the first term on the right-hand side of this equation we use projection operators.

The projection operator used in this article is defined as

\[ \hat{P}^{(m)}_{ii} = \frac{d_m}{\hbar} \sum_R D^{(m)}_{ii}(R) \hat{R}, \]  

where \( m \) labels an irreducible representation, \( i \) labels a basis function, \( \phi^{(m)}_i \), that transforms like irreducible representation \( m \) (irreducible representations may be degenerate), \( \hat{R} \) is a symmetry operation in the group, \( \hbar \) is the order of the group, \( d_m \) is the dimension of irreducible representation \( m \), and \( D^{(m)}_{ii}(R) \) is the \( i \)th diagonal element of the matrix representation of \( \hat{R} \) in the \( \phi^{(m)}_i \) basis. To calculate energy levels \( \beta^{(m)}_i \) is only required for one \( i \) and we shall write \( \hat{P}^{(m)}_{ii} \) as \( \hat{P}^{(m)}_i \). If an irreducible representation is multidimensional an index \( i \) is implied. The projection operator \( \hat{P}^{(m)}_i \) projects the \( m \)th symmetry component from a general vector with \( N \) symmetry components without changing the amplitude and sign of the component. For example,

\[ \hat{P}^{(m)} \sum_{n=1}^{N} v^{(n)} = v^{(m)}. \]  

Projection operators commute with the Hamiltonian, i.e.,

\[ \hat{P}^{(m)} H - H \hat{P}^{(m)} = 0. \]  

Combining Eq. (4) with Eq. (5) it is clear that

\[ H v^{(m)} = \hat{P}^{(m)} \left[ \sum_{n=1}^{N} v^{(n)} \right]. \]  

Therefore, rather than computing \( H v^{(m)} \) in Eq. (2) directly one can obtain it from Eq. (6). At each iteration one applies \( H \) to the sum of the Lanczos vectors computed at the previous iteration. Note that in this fashion \( H v^{(m)} \) is obtained for all \( m \) from one set of matrix vector products.

We observe that in some cases after computing \( \beta^{(m)}_{i+1} v^{(m)}_{i+1} \) it is necessary to purify it by applying the projection operator to it, i.e., \( \beta^{(m)}_{i+1} v^{(m)}_{i+1} \) is replaced by \( \hat{P}^{(m)} \beta^{(m)}_{i+1} v^{(m)}_{i+1} \). This purification step is not always required but it is necessary for the local mode example presented in the next section. Without the purification step, for some symmetries, energy levels are computed correctly, but for other symmetries, the energy levels are never copied and some energy levels are incorrect. If we use Eq. (6) to obtain the first term on the right-hand side of Eq. (2) and do not purify, contributions from wrong symmetry vectors creep into our Lanczos vectors, \( v^{(m)}_i \). To confirm the importance of the purification step we computed
a contamination amplitude, \( (1 - |\beta^{(m)}| z^2 |z|) \), where \( z = \beta^{(m)}_i v_{i+1} \) (which would be exactly zero if the precision of the computer’s arithmetic were infinite). It is usually of the order of \( 10^{-15} \) (we use double precision) but for the silane example of the next section, without purification, it can be as large as 50\%. The contamination prevents the calculation of accurate energy levels of \( A_1 \) and \( F_2 \) symmetries. Purifying \( \beta^{(m)}_i v_{i+1} \) resolves the contamination problem and allows us to calculate accurate energy levels.

Using the Lanczos algorithm to calculate separately energy levels in all \( N \) irreducible representations by using symmetry-adapted starting vectors would require a factor of \( N \) more matrix–vector products. It is noted that the SAL method can be used to calculate energy levels in any number of symmetry species. One might wish to calculate energy levels of some but not all of the symmetries. It is easy and inexpensive to apply a projection operator because it only involves interchanging vector indexes and multiplying vector components by coefficients; the cost scales as \( N g n^f \), where \( g \) is the number of operations for which \( \hat{K} \mathbf{v} \) must be computed.

The most efficient way to implement the symmetry adapted Lanczos algorithm entails storing Lanczos vectors for each irreducible representation. To reduce storage requirements one can store only sums of Lanczos vectors for all the symmetries and recompute the symmetry-adapted Lanczos vectors by applying the projection operators to the sums.

If one is interested only in a single symmetry species one can use the conventional Lanczos algorithm with a starting vector that transforms like the irreducible representation of interest. However, to avoid converging energy levels of other symmetries it is necessary to apply the corresponding projection operator \( \hat{P} \beta^{(m)} \) to the vector \( \beta^{(m)}_i v_{i+1} \) at each iteration. This is certainly better than using the method of Ref. 17 to distinguish between correct symmetry and wrong symmetry energy levels. Projecting is especially important if \( \beta^{(m)}_i v_{i+1} \) is small. If \( \beta^{(m)}_i v_{i+1} \) is very small, very small components of \( \beta^{(m)}_i v_{i+1} \) which would be zero if the computer were infinitely precise are magnified when the vector is normalized. The projection eliminates contributions from wrong symmetry vectors. For example, if we calculate energy levels of one symmetry of silane without projecting, the contamination amplitude, with \( z = \beta^{(m)}_i v_{i+1} \) can become as large as \( 10^{-7} \) when \( \beta^{(m)}_i \) is as small as \( 10^{-5} \) \( \text{cm}^{-1} \). Using a symmetry-adapted starting vector and applying the projector to \( \beta^{(m)}_i v_{i+1} \) would be better than applying the SAL method presented above because the SAL method would require two projections [one when \( H \mathbf{v}^{(m)} \) is computed and one to purify] rather than just one.

iii. results

To confirm the efficiency of the SAL algorithm we apply it to calculate stretching energy levels of silane, a five-atom tetrahedral molecule. The appropriate group is \( T_d \). When stretch and bend degrees of freedom are only weakly coupled a coupled Morse oscillator model does a good job of describing the stretch overtones. In this local mode model, the Hamiltonian is

\[
H = \sum_{i=1}^{4} \left( -\frac{1}{2 \mu} \frac{\partial^2}{\partial r_{i}^2} + D \chi_{i}^2 \right) + \sum_{i<j} \left( \frac{1}{3 m_s} \frac{\partial}{\partial r_i} \frac{\partial}{\partial r_j} + \frac{f_{yy}}{a^2} y_i y_j \right),
\]

with

\[
y_i = 1 - \exp(-a(r_i - r_i^e)),
\]

where \( r_i \) is the length of bond \( i \), \( r_i^e \) is the equilibrium bond length of bond \( i \), \( \mu = m_{12} m_{j} / (m_{12} + m_{j}) \), and \( D \) and \( a \) define the Morse oscillator. The kinetic coupling term is the vibrational \( G \) matrix element with the bond angle fixed at its equilibrium value. The potential coupling term is a quadratic term with a coefficient \( f_{yy} \). The vibrational levels of this Hamiltonian are two-fold or three-fold degenerate and the local mode \( A_1 / F_2 \) pair states are very nearly degenerate. The existence of degenerate and nearly degenerate energy levels makes their calculation with the Lanczos method difficult.

We use the potential surface determined from a recent global fit\(^{23} \) of \( T_d \) and \( C_{3v} \) silane isotopomers: \( D_z = 37.616.406.460 \text{ cm}^{-1} \), \( a = 1.396.717 \text{ Å}^{-1} \), and \( f_{yy} = 1651.710 \text{ cm}^{-1} \text{ Å}^{-2} \).

We use a basis of products of Morse oscillator functions, \( |n_1,n_2,n_3,n_4,\rangle \), and calculate all matrix elements using closed-form expressions. See Ref. 22 for details. The basis size is determined by the condition \( n_i \leq 12 \). The basis size is \( 12^2 = 20.736 \). For \( ^{28} \text{SiH}_4 \), the spectral range of our Hamiltonian matrix is \( 92.685-4477 \text{ cm}^{-1} \).

We choose to study the spectral window from 17 341 to 18 377 \text{ cm}^{-1} which includes the following 19 bands: 3210 \( (A_1 + A_2 + 2E + 3F_1 + 3F_2) \), 3111 \( (A_1 + F_2) \), 2220\( (A_1 + F_2) \), 2211 \( (A_1 + E + F_2) \), and 7000 \( (A_1 + F_2) \). This window contains the lowest \( A_2 \) state and a nearly degenerate local mode pair state for which the difference between the partners \( |7000;A_1\rangle \) and \( |7000;F_2\rangle \) is only \( 1.7 \times 10^{-7} \text{ cm}^{-1} \).

The \( T_d \) projection operator we used for the SAL calculation is given in Table I by listing all the components of \( \beta^{(m)} |n_1,n_2,n_3,n_4,\rangle \). For multidimensional symmetry species such as \( E, F_1, \) and \( F_2, \) it suffices to use only one component of the projection operator. The projection operator in Table I is obtained using the representation matrices of the operators of \( T_d \) given in Ref. 24.

The SAL and conventional Lanczos results for the chosen spectral window are given in Table II where we compare the number of iterations required to copy each energy level. If an energy level is copied it is certain to be converged. The number of iterations necessary to obtain a copy is therefore a measure of the number of iterations required to converge an energy level. Energy levels may be converged before they are copied. We compare our energy levels to levels computed using Householder’s method and a basis symmetrized for \( T_d \) group. After symmetrization, the size of \( F_2 \) symmetrized basis is largest—4095.

The levels calculated using the two Lanczos methods and the conventional Householder method agree to within the last figure given in Table II. The results clearly indicate that to obtain copied energy levels (which are equal to the
exact Householder energy levels) many fewer iterations are needed with the SAL algorithm than with the conventional Lanczos algorithm. In particular, for the (3210:A2) level, only 70 iterations are needed with the SAL algorithm whereas 3165 are necessary with the conventional Lanczos algorithm. For this level the SAL algorithm is much more efficient because the (3210; A2) level is the lowest A2 level. The conventional Lanczos algorithm does a surprisingly poor job of calculating the (7000; F2) local mode pair states although the spacing is only $1.7 \times 10^{-7}$ cm$^{-1}$ and the (7000; F2) levels are triply degenerate. They are copied after only 1680 iterations. This may be due to the fact that this cluster of four levels is well-separated from its nearest-neighbors which are located at 409 cm$^{-1}$ above and 892 cm$^{-1}$ below.

It should be noted that although we compare the number of iterations required to copy a level once, accurate energy levels actually appear long before a level is copied. For example, using the SAL algorithm the (3210; A2) level is identical to the value reported in Table II after only 14 iterations but it is copied only at iteration 70.

The SAL and the SAFD$^{18,19}$ algorithms both employ projection operators. The SAL algorithm is easier to use because there are fewer parameters to choose and also has the advantage that one never misses energy values. It may also be less expensive.$^{25}$

### IV. CONCLUSION

We have developed a symmetry-adapted Lanczos method to calculate energy levels of different symmetries from a single set of matrix–vector products. The SAL method has two important advantages: (1) using it one obtains symmetry-labeled energy levels, (2) it significantly reduces the number of matrix–vector products required to achieve convergence. There are two alternatives to the method we suggest: (1) choose coordinates so that each symmetry operation affects only one coordinate, this is frequently impossible; (2) use symmetry adapted starting vectors and do one iteration for each symmetry species; this is costly.

Our scheme involves applying as many projection operators as there are irreducible representations for which one wishes to calculate energy levels. In the 4D coupled Morse oscillator problem which has two- and three-dimensional irreducible representations and nearly degenerate four-fold clusters of levels, the SAL algorithm reduces the number of iterations required to obtain converged energy levels by a factor 2.5 to 45.

The SAL algorithm could also be used to calculate symmetry-adapted wave functions. A good way to do this would be to apply our projection operator idea to the Lanczos-based method presented in Ref. 26.

---

**TABLE I.** The projection operators, $P^{m}$, for the $T_{d}$ group represented by listing all 24 components of $P^{m}$ (1234), where (1234) stands for $|a_{1}, a_{2}, a_{3}, a_{4}$.)

<table>
<thead>
<tr>
<th>m</th>
<th>1234</th>
<th>1243</th>
<th>1324</th>
<th>1342</th>
<th>1423</th>
<th>1432</th>
<th>2134</th>
<th>2143</th>
<th>2314</th>
<th>2341</th>
<th>2413</th>
<th>2431</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_{1}$</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>$A_{2}$</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>$E_{a}$</td>
<td>+1</td>
<td>+1</td>
<td>-1/2</td>
<td>-1/2</td>
<td>-1/2</td>
<td>-1/2</td>
<td>+1</td>
<td>-1/2</td>
<td>-1/2</td>
<td>-1/2</td>
<td>-1/2</td>
<td>+1</td>
</tr>
<tr>
<td>$F_{1c}$</td>
<td>+1</td>
<td>-1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>-1</td>
<td>+1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$F_{2c}$</td>
<td>+1</td>
<td>+1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>+1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

**TABLE II.** Comparison of the SAL and conventional Lanczos algorithms for high-lying stretch energy levels of $^{28}$SiH$_{4}$.

<table>
<thead>
<tr>
<th>State</th>
<th>Exact</th>
<th>SAL</th>
<th>Lanczos</th>
</tr>
</thead>
<tbody>
<tr>
<td>3210:A1</td>
<td>17 341.600 856 191</td>
<td>604</td>
<td>2749</td>
</tr>
<tr>
<td>3210:F2</td>
<td>17 342.300 163 103</td>
<td>1023</td>
<td>3295</td>
</tr>
<tr>
<td>3210:E</td>
<td>17 343.534 545 396</td>
<td>406</td>
<td>3402</td>
</tr>
<tr>
<td>3210:F2</td>
<td>17 343.929 473 801</td>
<td>1025</td>
<td>3386</td>
</tr>
<tr>
<td>3210:F1</td>
<td>17 344.596 340 347</td>
<td>366</td>
<td>3337</td>
</tr>
<tr>
<td>3210:F2</td>
<td>17 346.516 200 963</td>
<td>1024</td>
<td>3366</td>
</tr>
<tr>
<td>3210:F1</td>
<td>17 347.149 038 744</td>
<td>379</td>
<td>3402</td>
</tr>
<tr>
<td>3210:E</td>
<td>17 347.602 996 301</td>
<td>399</td>
<td>3378</td>
</tr>
<tr>
<td>3210:F1</td>
<td>17 348.621 483 667</td>
<td>378</td>
<td>3337</td>
</tr>
<tr>
<td>3210:A2</td>
<td>17 349.225 775 916</td>
<td>70</td>
<td>3165</td>
</tr>
<tr>
<td>3111:A1</td>
<td>17 341.641 082 776</td>
<td>696</td>
<td>2326</td>
</tr>
<tr>
<td>2220:A1</td>
<td>17 341.969 071 004</td>
<td>697</td>
<td>2685</td>
</tr>
<tr>
<td>3111:F3</td>
<td>17 341.975 843 641</td>
<td>1023</td>
<td>2685</td>
</tr>
<tr>
<td>2220:F2</td>
<td>17 341.081 834 451</td>
<td>1022</td>
<td>2571</td>
</tr>
<tr>
<td>2211:A1</td>
<td>17 477.663 721 031</td>
<td>597</td>
<td>1589</td>
</tr>
<tr>
<td>2211:F2</td>
<td>17 482.387 790 410</td>
<td>726</td>
<td>1621</td>
</tr>
<tr>
<td>2211:E</td>
<td>17 484.761 020 391</td>
<td>357</td>
<td>1598</td>
</tr>
<tr>
<td>7000:A1</td>
<td>18 376.192 488 332</td>
<td>513</td>
<td>1680</td>
</tr>
<tr>
<td>7000:F2</td>
<td>18 376.192 488 506</td>
<td>667</td>
<td>1680</td>
</tr>
</tbody>
</table>

$^{a}$The levels (in cm$^{-1}$) calculated using the two Lanczos methods compared in this table and the conventional Householder method agree to within the last figure given. The zero-point energy 4477.169 311 537 cm$^{-1}$ is not subtracted.

$^{b}$The number of iterations required to copy the level.