Evaluation and optimal computation of angular momentum matrix elements: An information theory approach

MARIA EUGENIA CASTRO, ALFONSO NIÑO, CAMELIA MUÑOZ-CARO
Grupo de Química Computacional y Computación de Alto Rendimiento
Escuela Superior de Informática. Universidad de Castilla-La Mancha
Paseo de la Universidad 4, 13071, Ciudad Real
SPAIN
{mariaeugenia.castro,alfonso.nino,camelia.munoz}@uclm.es http://qcycar-uclm.esi.uclm.es

Abstract: In this work, we determine all possible angular momentum matrix elements arising in the variational treatment of the rovibrational molecular Hamiltonian. In addition, the logic of the associated computing process is organized in a series of decision tables. Using Shwayder´s approach, information theory is applied to obtain optimal computing codes from the decision tables. The needed decision rules apparition frequencies are computed as a function of the rotational quantum number *J*. Using these values, we show that the codes obtained are optimal for any value of *J*. In all cases, the optimal codes exhibit an efficiency of at least a 97% of the theoretical maximum. In addition, pessimal codes are obtained as a counterpart of the optimal ones. We find that the efficiency difference between the optimal and pessimal codes reaches quickly a limit for increasing values of the *J* quantum number.

Key-Words: - Molecular rovibrational Hamiltonian; Angular momentum operators; Matrix elements; Decision tables; Information theory; Information entropy.

1 Introduction

The construction and resolution of the molecular Hamiltonian is an important topic in spectroscopy since the first times of quantum mechanics [1]. The simulation and interpretation of molecular rotational-vibrational (rovibrational) spectra needs to solve the corresponding Hamiltonian. This rovibrational Hamiltonian is defined by the kinetic and potential energy operators for the nuclei motion [1-4]. For a complete description, the three components of the kinetic operator are needed, namely: the pure rotational, the pure vibrational, and the rotation-vibration coupling. These three elements involve the effect of angular momentum operators. Within a variational framework [5, 6], the efficient computation of the different angular momentum matrix elements will allow to analyze molecules of arbitrary complexity. Therefore, we need a way to organize and optimize the logic of this process.

A classical Decision Table (DT) is a tabular form displaying the full decision logic of a problem [7]. Thus, the associated DT describes the existing set of conditions, as well as the set of actions to take according to these conditions. The conditions define an upper block of rows (conditions matrix), whereas the actions define a lower block (actions matrix). The columns of the DT define the combination of conditions corresponding to the different actions (decision rules). DT´s are a tool dating back to the early 60´s of the 20th century [7-15]. However, in several formats, they are still used for the optimization of processes in different research areas [16-20].

DT's can be used to generate an optimized computer code for solving the problem at hand. This optimization refers to a reduction of machine execution time, required machine memory, or number of decision rules. Along the years, several algorithms have been developed in order to reach these goals [7-15]. A key work dealing with the conversion of DT's to computer code was presented by Pollack in 1965 [9]. In this work, Pollack proposed two algorithms: one for reducing computer storage, and other for reducing total computer running time. Both algorithms are based on the localization of indifferences in decision rules [9]. Although this procedure is adequate for cases where indifferent conditions do exist, it can be ambiguous for problems with only independent cases (no indifferences). Here, a more general and formally sound method was developed by Shwayder [11]. The method applies an information theory approach [21, 22] relying in the concept of information entropy. Information theory is based in the seminal work of Shannon [21], and deals with the efficient coding of messages and communicating data [22]. Here, the information entropy (*H*) quantifies the number of bits needed for

representing the result of an uncertain event (i.e., the information contained in a message) [21, 22].

In this work, we focus in the computation of the angular momentum matrix elements for overall rotation appearing in any variational treatment of the molecular rovibrational Hamiltonian. The calculation of these matrix elements is carried out relying in the use and code conversion of decision tables. A preliminary version of this work was presented in reference [23]. To generalize the variational treatment of rovibrational Hamiltonians, we develop in this work the optimal algorithms for computing angular momentum matrix elements. Thus, we calculate the value of the matrix elements, define the corresponding DT´s, and generate optimized codes from them. In addition, the efficiencies of the codes are quantified as proposed by Shwayder's [11].

2 Theory

The kinetic energy operator of the rovibrational molecular Hamiltonian [24, 25] can be expressed as a function of the angular momentum for overall rotation, $\mathbf{J}= (\partial T/\omega)$, and for the change on vibrational coordinates, $\mathbf{p}= (\partial T/\dot{q})$. Thus,

$$\hat{T} = -\frac{\hbar^2}{2}\left(\mathbf{J}\,\mathbf{G}_{\mathbf{R}}\,\mathbf{J} + \mathbf{J}\,\mathbf{G}_{\mathbf{RV}}\,\mathbf{p} + \mathbf{p}\,\mathbf{G}_{\mathbf{RV}}^{\mathbf{T}}\,\mathbf{J} + \mathbf{p}\,\mathbf{G}_{\mathbf{v}}\,\mathbf{p}\right) + U \quad (1)$$

where $\mathbf{G}_R$, $\mathbf{G}_{RV}$, and $\mathbf{G}_V$ are components of the rovibrational **G** matrix [26-28]:

$$\mathrm{G} = \begin{pmatrix} \mathrm{G}_{\mathrm{R}} & \mathrm{G}_{\mathrm{RV}} \\ \mathrm{G}_{\mathrm{RV}}^{T} & \mathrm{G}_{\mathrm{V}} \end{pmatrix} = \begin{pmatrix} \mathrm{I} & \mathrm{X} \\ \mathrm{X}^{T} & \mathrm{Y} \end{pmatrix}^{-1} \quad (2)$$

for pure rotation, rovibrational coupling and pure vibration, respectively. U is the pseudopotential term usually included in the potential function. I, X and Y are the inertial tensor matrix, the rotation-vibration matrix and pure vibration matrix, respectively.

Considering the angular momentum operator, **J**, along the molecule-fixed axis (x, y, z), its components $\mathbf{J}_x$, $\mathbf{J}_y$, and $\mathbf{J}_z$, satisfy the following commutation relations [4]:

$$\begin{aligned} \left[\mathrm{J}_x, \mathrm{J}_y\right] &= -i\hbar \mathrm{J}_z, \\ \left[\mathrm{J}_y, \mathrm{J}_z\right] &= -i\hbar \mathrm{J}_x, \\ \left[\mathrm{J}_x, \mathrm{J}_z\right] &= -i\hbar \mathrm{J}_y, \\ \left[\mathrm{J}_z, \mathrm{J}_{\pm}^{m}\right] &= \mp\hbar \mathrm{J}_{\pm}^{m}, \end{aligned} \quad (3)$$

where:

$$\mathrm{J}_{\pm}^{m} = \mathrm{J}_x \pm i\mathrm{J}_y \quad (4)$$

are the ladder operators.

Using equations (3) and (4), and a complete set of basis functions, we can obtain all the matrix elements arising in any variational treatment of the rovibrational Hamiltonian. These elements have the form:

$$\left\langle J',k',m'\middle|\hat{\mathrm{O}}\middle|J,k,m\right\rangle \quad (5)$$

Here, $\hat{O}$ represents any angular momentum operator and J, k, and m (and their primed counterparts) refer to the usual rotational quantum numbers:

$$\begin{aligned} J &= 0,\ 1,\ 2,.... \\ k &= -J, -J+1, -J+2, ..., J-2, J-1, J \\ m &= -J, -J+1, -J+2, ..., J-2, J-1, J \end{aligned} \quad (6)$$

Using symmetric rotor eigenfunctions, we can obtain the non-vanishing matrix elements applying the technique described in [3, 29, 30]. Due to the orthogonality of the rotational eigenfunctions all the elements with $J' \neq J$ disappear. For the rest, it is interesting to note that only three different cases can appear:

1) Operators with the form $\mathbf{J}_a$, where a indicates the x, y, or z component.
2) Operators with the form $(\mathbf{J}_a)^2$.
3) Operators with the form $\mathbf{J}_a\mathbf{J}_b$, where b= x, y or z component different to a.

Case 1) corresponds to the rovibrational coupling, whereas cases 2) and 3) correspond to pure rotation. The results we obtain are collected in Table 1.

3 Methods and Implementation

3.1. Conversion to optimized code

The results for the three cases shown in Table 1 are transformed in three different DT's, as shown in Tables 2, 3 and 4. In these DT's the C's are the conditions, the A's are the actions, and the R's represent the decision rules (i.e., the matrix elements collected in Table 1).

To convert the DT's in a computer code, optimizing the execution time, we use the Shwayder approach [11], which is based in Shannons's noiseless coding theorem [21, 22]. In a noiseless channel the information (messages) are transmitted with no possibility of error. For transmitting the messages through the noiseless channel as fast as possible, it must be selected a sequence of code characters for each message (code word) so that the average length of code words is as small as possible [11, 21, 22].

Using the noiseless coding theorem of Shannon it is possible to establish a measure of efficiency as [21]:

$$\text{efficiency} = \frac{\left(-\sum_{i=1}^{m} p_i \log_2 p_i\right) \Big/ \log_2 2}{\text{average word length}} \quad (7)$$

In equation (7) the numerator is a measure of the variability of the set of messages, and is called information entropy, H. The p_i are the relative frequencies of the messages. The unit of measurement of entropy is bits, with $\log_2 2=1$ bit. The average word length in the denominator is obtained as the sum of the products of the code-word length times the frequency of each message. Applying this reasoning to the decision logic problem in a computer code, we can use the similarity between the coding problem and the decision table construction.

Table 1. All possible non-vanishing ($J'=J$) matrix elements for angular momentum operators.

Operator ($\hat{O}$)	k′	Matrix element	Label
$\mathbf{J}_x$	k–1	$\hbar/2[(J+k)(J-k+1)]^{1/2}$	x1
	k+1	$\hbar/2[(J-k)(J+k+1)]^{1/2}$	x2
$\mathbf{J}_y$	k–1	$-i\hbar/2[(J+k)(J-k+1)]^{1/2}$	y1
	k+1	$i\hbar/2[(J-k)(J+k+1)]^{1/2}$	y2
$\mathbf{J}_z$	k	$\hbar\, k$	z1
$\mathbf{J}_x^2$	k	$\hbar^2/2[J^2+J-k^2]$	xx1
	k–2	$\hbar^2/4[J^2+J-k^2-k]^{1/2}[J^2+J-k^2-3k-2]^{1/2}$	xx2
	k+2	$\hbar^2/4[J^2+J-k^2+k]^{1/2}[J^2+J-k^2+3k-2]^{1/2}$	xx3
$\mathbf{J}_y^2$	k	$\hbar^2/2[J^2+J-k^2]$	yy1
	k–2	$-\hbar^2/4[J^2+J-k^2-k]^{1/2}[J^2+J-k^2-3k-2]^{1/2}$	yy2
	k+2	$-\hbar^2/4[J^2+J-k^2+k]^{1/2}[J^2+J-k^2+3k-2]^{1/2}$	yy3
$\mathbf{J}_z^2$	k	$\hbar^2\, k^2$	zz1
$\mathbf{J}_x\mathbf{J}_y$	k	$-i\hbar^2/2\ k$	xy1
	k–2	$-i\hbar^2/4[J^2+J-k^2+k]^{1/2}[J^2+J-k^2+3k-2]^{1/2}$	xy2
	k+2	$i\hbar^2/4[J^2+J-k^2-k]^{1/2}[J^2+J-k^2-3k-2]^{1/2}$	xy3
$\mathbf{J}_y\mathbf{J}_x$	k	$i\hbar^2/2\ k$	yx1
	k–2	$-i\hbar^2/4[J^2+J-k^2+k]^{1/2}[J^2+J-k^2+3k-2]^{1/2}$	yx2
	k+2	$i\hbar^2/4[J^2+J-k^2-k]^{1/2}[J^2+J-k^2-3k-2]^{1/2}$	yx3
$\mathbf{J}_x\mathbf{J}_z$	k–1	$\hbar^2/2(k)[J^2+J-k^2+k]^{1/2}$	xz1
	k+1	$\hbar^2/2(k)[J^2+J-k^2-k]^{1/2}$	xz2
$\mathbf{J}_z\mathbf{J}_x$	k–1	$\hbar^2/2(k-1)[J^2+J-k^2+k]^{1/2}$	zx1
	k+1	$\hbar^2/2(k+1)[J^2+J-k^2-k]^{1/2}$	zx2
$\mathbf{J}_y\mathbf{J}_z$	k–1	$-i\hbar^2/2(k)[J^2+J-k^2+k]^{1/2}$	yz1
	k+1	$i\hbar^2/2(k)[J^2+J-k^2-k]^{1/2}$	yz2
$\mathbf{J}_z\mathbf{J}_y$	k–1	$-i\hbar^2/2(k-1)[J^2+J-k^2+k]^{1/2}$	zy1
	k+1	$i\hbar^2/2(k+1)[J^2+J-k^2-k]^{1/2}$	zy2

Table 2. DT for the $\mathbf{J}_a$ angular momentum operators.

	$\mathbf{J}_a$	R1	R2	R3	R4	R5
C1	a=x	1	1	0	0	0
C2	a=y	0	0	1	1	0
C3	k′=k–1	1	0	1	0	0
	f (J=50)	0.199	0.199	0.199	0.199	0.201
A1	x1	X	-	-	-	-
A2	x2	-	X	-	-	-
A3	y1	-	-	X	-	-
A4	y2	-	-	-	X	-
A5	z1	-	-	-	-	X

Table 3. DT for the $(\mathbf{J}_a)^2$ angular momentum operators.

	$(\mathbf{J}_a)^2$	R1	R2	R3	R4	R5	R6	R7
C1	a=x	1	1	1	0	0	0	0
C2	a=y	0	0	0	1	1	1	0
C3	k′=k	1	0	0	1	0	0	1
C4	k′=k–2	0	1	0	0	1	0	0
	f (J=50)	0.144	0.141	0.141	0.144	0.141	0.141	0.144
A1	xx1	X	-	-	X	-	-	-
A2	xx2	-	X	-	-	-	-	-
A3	xx3	-	-	X	-	-	-	-
A4	yy2	-	-	-	-	X	-	-
A5	yy3	-	-	-	-	-	X	-
A6	zz1	-	-	-	-	-	-	X

Table 4. DT for the $\mathbf{J}_a\mathbf{J}_b$ angular momentum operators.

	$\mathbf{J}_a\mathbf{J}_b$	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14
C1	a=x	1	1	1	1	1	0	0	0	0	0	0	0	0	0
C2	a=y	0	0	0	0	0	1	1	1	1	1	0	0	0	0
C3	b=x	0	0	0	0	0	1	1	1	0	0	1	1	0	0
C4	b=y	1	1	1	0	0	0	0	0	0	0	0	0	1	1
C5	k′=k	1	0	0	0	0	1	0	0	0	0	0	0	0	0
C6	k′=k–1	0	0	0	1	0	0	0	0	1	0	1	0	1	0
C7	k′=k–2	0	1	0	0	0	0	1	0	0	0	0	0	0	0
	f(J=50)	0.072	0.070	0.070	0.071	0.071	0.072	0.070	0.070	0.071	0.071	0.071	0.071	0.071	0.071
A1	xy1	X	-	-	-	-	-	-	-	-	-	-	-	-	-
A2	xy2	-	X	-	-	-	-	X	-	-	-	-	-	-	-
A3	xy3	-	-	X	-	-	-	-	X	-	-	-	-	-	-
A4	xz1	-	-	-	X	-	-	-	-	-	-	-	-	-	-
A5	xz2	-	-	-	-	X	-	-	-	-	-	-	-	-	-
A6	yx1	-	-	-	-	-	X	-	-	-	-	-	-	-	-
A7	yz1	-	-	-	-	-	-	-	-	X	-	-	-	-	-
A8	yz2	-	-	-	-	-	-	-	-	-	X	-	-	-	-
A9	zx1	-	-	-	-	-	-	-	-	-	-	X	-	-	-
A10	zx2	-	-	-	-	-	-	-	-	-	-	-	X	-	-
A11	zy1	-	-	-	-	-	-	-	-	-	-	-	-	X	-
A12	zy2	-	-	-	-	-	-	-	-	-	-	-	-	-	X

So, the Schwayder´s approach [11] establishes that the optimal code is obtained by maximizing the associated information entropy (*H*). Therefore, we start with a given DT, selecting the condition with the highest entropy. Then, for each branch of the condition, we have two sub-DT's. One of the sub-DT's corresponds to the rules with the initial condition set to true (or 1), and the other to the rules with the condition set to false (or 0). For each sub-DT we select again the condition with the highest entropy, and the process is applied recursively until no conditions left. In this form, we obtain the optimal testing order of the conditions. Considering that we have no indifferences and that we use two coding characters (1-0 or true-false), the maximum information entropy for a given condition is obtained as [11]:

$$H = -\left(P(1)\log_2 P(1) + P(0)\log_2 P(0) \right) \quad (8)$$

In equation (8), *P(1)* represents the probability of the condition being true (or 1), and *P(0)* the probability of being false (or 0). For computing *P(0)* and *P(1)* we need the frequency of apparition (*f*) of each decision rule (columns of the DT). Therefore,

$$P(1) = \Sigma f_i \text{ (with condition equal 1)} \quad (9)$$

$$P(0) = 1 - P(1)$$

To determine the frequencies for the different decision rules in Tables 2, 3 and 4, we consider that for each *J* rotational quantum number we have 2*J*+1 *k* and *m* quantum number values, see equation (6). Therefore, we obtain:

$$f(k'=k) = (2J+1)/T$$

$$f(k'=k\pm1) = 2J/T \quad (10)$$

$$f(k'=k\pm2) = (2J-1)/T$$

where *T* is the total number of the integrals evaluated for a given *J* rotational quantum number for the possible values of *k*, i.e., *6J*.

The frequency of each decision rule is computed as a function of *J*, for values ranging from 0 to 50. The results for the three DT's considered are shown in Figure 1. Here, we observe that when *J* increases the decision rules frequencies converge to a similar value. The frequencies differ at most in the second decimal place. In addition, for the entire *J* interval, the relative position of the different decision rules is maintained. These facts indicate that the optimal code resulting from the DT's should be the same independently of *J*. Tables 2, 3, and 4 collect the results for *J*=50.

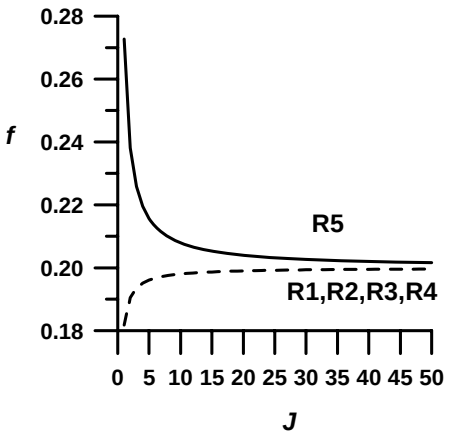
With the frequencies values, we can apply the described procedure to transform the three DT's in the corresponding optimized code.

Case 1.

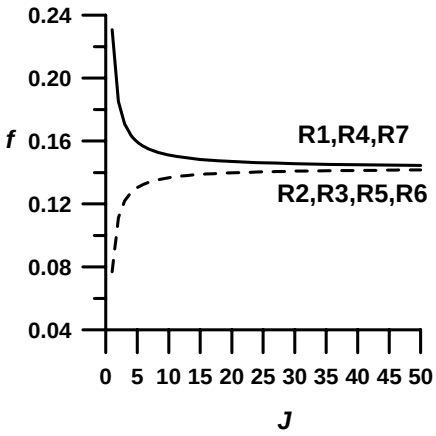
The DT in Table 2 is the simplest one. Three conditions (C) are evaluated to generate the five decision rules (R). In the conditions, values of *a*= *x*, and *y* are taken into account. The *a*= *z* component, is implicitly taken into account. In the same form, only a condition for k'= *k*+1 or *k*-1 is needed to generate all the remaining decision rules. Five decision rules (R) are evaluated corresponding to the five different matrix elements (the actions, A), see Table 1. Chart1 shows the resulting code that maximizes information entropy. On the other hand, for the sake of

comparison, we have obtained the worst, pessimal, code. To such an end, we have used the inverse process, i. e., we select first the conditions with smaller entropy in the decision process. Chart 2 shows the resulting code.

Case 1.



Case 2.



Case 3.

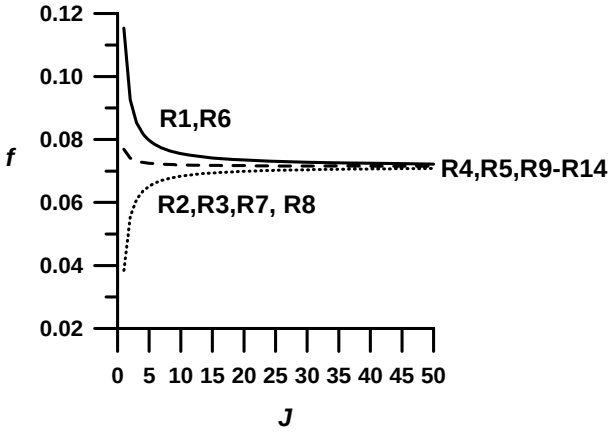


Figure 1. Variation of the decision rules frequencies as a function of the rotational *J* quantum number. The rules appearing in the same set have the same frequency.

```
if C1 then
  if C3 then
    <R1>
  else <R2>
  end_if
else
  if C2 then
    if C3 then
      <R3>
    else <R4>
    end_if
  else <R5>
  end_if
end_if
```

Chart 1. Optimal pseudocode for the DT in Table 2.

```
if C1 then
  if C3 then
    <R1>
  else <R2>
  end_if
else
  if C3 then
    <R3>
  else
    if C2 then
      <R4>
    else <R5>
    end_if
  end_if
end_if
```

Chart 2. Pessimal pseudocode for the DT in Table 2.

Case 2.

Here, the DT for the double application of the $\mathbf{J}_a$ operator is presented. This case corresponds to the pure rotational terms, the only terms appearing when using a principal axes coordinate system. As shown in Table 3, we have now four different conditions and seven different decision rules. Applying the maximization of the information entropy we obtain the code presented in Chart 3. The worst approximation obtained by the inverse process is shown in Chart 4.

```
if C1 then
  if C3 then
    <R1>
  else
    if C4 then
      <R2>
    else <R3>
    end_if
  end_if
else
  if C3 then
    if C2 then
      <R4>
    else <R7>
    end_if
  else
    if C4 then
      <R5>
    else <R6>
    end_if
  end_if
end_if
```

Chart 3. Optimal pseudocode for the DT in Table 3.

```
if C1 then
  if C4 then
    <R2>
  else
    if C3 then
      <R1>
    else <R3>
    end_if
  end_if
else
  if C4 then
    <R5>
  else
    if C3 then
      if C2 then
        <R4>
      else <R7>
      end_if
    else <R6>
    end_if
  end_if
end_if
```

Chart 4. Pessimal pseudocode for the DT in Table 3.

Case 3.

This is the most complex case. Here, we have seven different conditions and fourteen decision rules, see Table 4. Now the cases $k \leftarrow k$, $k\pm2$, and $k\pm1$ are taken into account. After maximizing the information entropy the time optimal code shown in Chart 5 is obtained. Chart 6 present the pessimal pseudocode for the decision process.

```
if C1 then
  if C4 then
    if C5 then
      <R1>
    else
      if C7 then
        <R2>
      else <R3>
      end_if
    end_if
  else
    if C6 then
      <R4>
    else <R5>
    end_if
  end_if
else
  if C2 then
    if C3 then
      if C5 then
        <R6>
      else
        if C7 then
          <R7>
        else <R8>
        end_if
      end_if
   else
     if C6 then
       <R9>
     else <R10>
     end_if
   end_if
  else
    if C4 then
      if C6 then
        <R13>
      else <R14>
    else
      if C6 then
        <R11>
      else <R12>
      end_if
    end_if
  end_if
end_if
```

Chart 5. Optimal pseudocode for the DT in Table 4.

```
if C7 then
  if C1 then
    <R2>
  else <R7>
  end_if
else
  if C5 then
    if C1 then
      <R1>
    else <R6>
    end_if
  else
     if C1 then
       if C4 then
         <R3>
       else
         if C6 then
           <R4>
         else <R5>
         end_if
       end_if
     else
       if C4 then
         if C6 then
           <R3>
         else <R14>
         end_if
       else
         if C2 then
           if C3 then
             <R8>
           else
             if C6 then
               <R9>
             else <R10>
             end_if
           end_if
         else
           if C6 then
             <R11>
           else <R12>
           end_if
         end_if
       end_if
     end_if
   end_if
end_if
```

Chart 6. Pessimal pseudocode for the DT in Table 4.

3.2. Efficiency of the optimized code

It is possible to quantify the efficiency of the codes as proposed by Shwayder [11]. The idea is to determine the average information entropy per comparison.

Since the theoretical maximum is 1 bit of entropy per comparison, the average entropy directly gives how close we are to the maximum possible information entropy. This value, therefore, represents a measure of the efficiency of the code proposed. The

average entropy per comparison can be determined as follows. First we identify the conditions that appear in the code, some can appear more than once. For instance, in the pseudocode of Chart 1 we test four conditions, C1, C3, C2 and C3. For each condition we determine its frequency of evaluation (*f*) by adding the probabilities of all the decision rules than can be reached from it. In addition we compute the maximum information entropy (*H*) of each condition using equation (5). The *P(1)* and *P(0)* probabilities for each condition are obtained from the frequencies of the decision rules below it using equation (6). The weighted entropies (*WH*) are defined as the product *f*H* for each condition. The efficiency is obtained as the ratio of total *WH* to total *f*. Table 5 shows the results for the codes in Charts 1, 3 and 5 for the optimal pseudocodes. On the other hand Table 6 shows the results for the worst cases in Charts 2, 4 and 6. Table 5 shows that the three best implementations, Charts 1, 3, and 5, have efficiencies of 97%, 98%, and 97%, respectively. The worst implementations in Table 6 have efficiencies of 97%, 93%, and 81%, respectively.

The previous results were obtained for the limiting *J*=50 value. To analyze the evolution of the optimal versus the pessimal case, we have considered the efficiency as a function of the *J* quantum number. The results are shown in Figure 2. In all cases, we observe that the difference in efficiency converge to a limit as *J* increases. In addition, we find that in case 1 the difference between efficiencies is the smallest in comparison with the other cases. This is a consequence of the small number of conditions taken into account in the decision process, see Table 2. In the large *J* limit, we found an efficiency difference of 5% for the $(\mathbf{J}_a)^2$ case. The largest difference, 16%, is found in the most complex case, $\mathbf{J}_a\mathbf{J}_b$.

4 Conclusions

In this paper, we obtain the most efficient algorithms for computing all possible angular momentum matrix elements involving overall rotation. Using symmetric rotor eigenfunctions we determine all the matrix elements needed for any variational treatment of overall rotation and rovibrational coupling.

The process logic is described using three decision tables (DT's). Then, the optimal computer code corresponding to each DT is obtained by maximizing the information entropy (*H*) of the resulting algorithm.

We find that the frequency of apparition of each matrix element is almost constant for different values of the rotational quantum number *J*. Since this frequency is the factor determining the resulting computer code, the codifications proposed are optimal for any *J* quantum number.

The efficiency of each codification is quantified by comparison with the theoretical maximum value of *H*. We find efficiencies of 97%, 98% and 97% for the three considered DT's. These results show that the codifications are extremely efficient.

The pessimal code cases are also evaluated to compare the efficiencies obtained. The results are evaluated as a function of the rotational number *J*. The results show that the difference between the efficiencies of the optimal and the pessimal cases increases when the process logic is most complex, i.e., case 3.

Table 5. Efficiencies of the algorithms corresponding to the codes in Charts 1, 3 and 5.

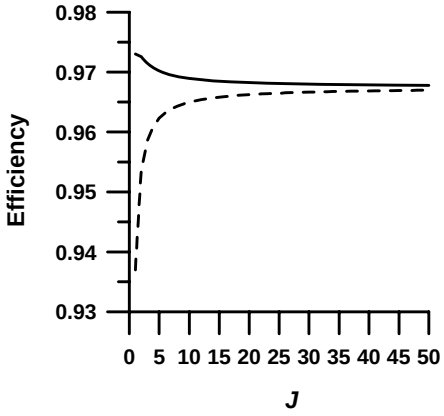
Case 1. Code in Chart 1				
Decision node	**Condition**	***H***	***f***	***WH***
1	C1	0.9704	1.0000	0.9704
2	C3	1.0000	0.3992	0.3992
3	C2	0.9204	0.6007	0.5528
4	C3	1.0000	0.3992	0.3992
Total			2.3992	2.3216
Efficiency= 0.9677=97 %				
Case 2. Code in Chart 3.				
Decision node	**Condition**	***H***	***f***	***WH***
1	C1	0.9848	1.0000	0.9848
2	C3	0.9226	0.4277	0.3946
3	C3	0.9999	0.5722	0.5722
4	C4	1.0000	0.2832	0.2832
5	C2	1.0000	0.2889	0.2889
6	C4	1.0000	0.2832	0.2832
Total			2.8555	2.8072
Efficiency=0.9831= 98 %				
Case 3. Code in Chart 5.				
Decision node	**Condition**	***H***	***f***	***WH***
1	C1	0.9401	1.0000	0.9401
2	C4	0.9714	0.3569	0.3467
3	C2	0.9912	0.6430	0.6374
4	C5	0.9226	0.2138	0.1973
5	C6	1.0000	0.1430	0.1430
6	C3	0.9700	0.3569	0.2410
7	C4	1.0000	0.2861	0.2861
8	C7	1.0000	0.1416	0.1416
9	C5	0.9226	0.2138	0.1973
10	C6	1.0000	0.1430	0.1430
11	C6	1.0000	0.1430	0.1430
12	C6	1.0000	0.1430	0.1430
13	C7	1.0000	0.1416	0.1416
Total			3.9263	3.7016
Efficiency=0.9696= 97 %				

Table 6. Efficiencies of the algorithms corresponding to the codes in Charts 2, 4 and 6.

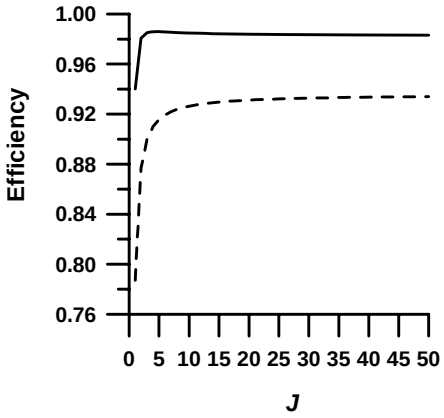
Case 1. Code in Chart 2.				
Decision node	**Condition**	***H***	***f***	***WH***
1	C1	0.9704	1.0000	0.9704
2	C3	1.0000	0.3992	0.3992
3	C2	0.9171	0.6007	0.5510
4	C3	0.9999	0.4011	0.4011
Total			2.4012	2.3219
Efficiency=0.9669= 97 %				
Case 2. Code in Chart 4.				
Decision node	**Condition**	***H***	***f***	***WH***
1	C1	0.9848	1.0000	0.9848
2	C3	0.9160	0.4277	0.3918
3	C3	0.8072	0.5722	0.4619
4	C4	0.9999	0.2861	0.2861
5	C2	0.9138	0.4306	0.3934
6	C4	1.0000	0.2889	0.2889
Total			3.0057	2.8072
Efficiency=0.9339= 93 %				
Case 3. Code in Chart 6.				
Decision node	**Condition**	***H***	***f***	***WH***
1	C7	0.5884	1.0000	0.5884
2	C1	1.0000	0.1416	0.1416
3	C5	0.6538	0.8583	0.5612
4	C1	1.0000	0.1444	0.1444
5	C1	0.8808	0.7138	0.6287
6	C4	0.9160	0.2138	0.1959
7	C4	0.8636	0.5000	0.4318
8	C6	1.0000	0.1430	0.1430
9	C6	1.0000	0.1430	0.1430
10	C2	0.9714	0.3569	0.3467
11	C3	0.9160	0.2138	0.1959
12	C6	1.0000	0.1430	0.1430
13	C6	1.0000	0.1430	0.1430
Total			4.7153	3.8073
Efficiency=0.8074=81 %				

Acknowledgments:
This work has been cofinanced by FEDER funds and the Consejería de Educación y Ciencia de la Junta de Comunidades de Castilla-La Mancha (grant # PBI08-0008). The Ministerio de Educación y Ciencia of Spain (grant# AYA2008-00446) is also acknowledged. M.E. Castro thanks the Consejo Nacional de Ciencia y Tecnología, CONACyT (Mexico) for a graduate grant (grant # 171982), and the Ministerio de Asuntos Exteriores y de Cooperación-Agencia Española de Cooperación Internacional para el Desarrollo, MAEC-AECID.

Case 1.



Case 2.



Case 3.

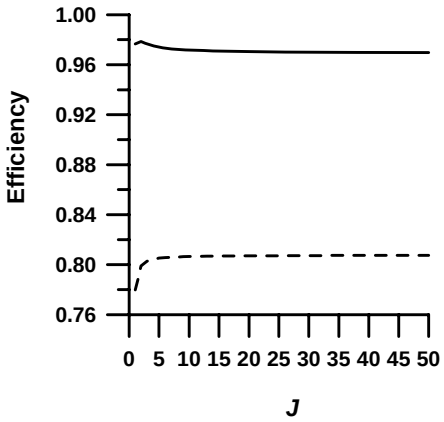


Figure 2. Variation of the efficiency as a function of the *J* rotational quantum number. The continuous line corresponds to the optimal code. The dashed line corresponds to the pessimal code.

References:

[1] Meyer H., The Molecular Hamiltonian, *Annual Reviews in Physical Chemistry*, No. 53, 2002, pp. 141-172.

[2] Meyer R., Günthard Hs.H., General Internal Motion of Molecules, Classical and Quantum-

Mechanical Hamiltonian, *Journal of Chemical Physics*, Vol. 49, No. 4, 1968, pp. 1510-1520.

[3] Watson J.K.G., Simplification of the molecular vibration-rotation Hamiltonian. *Molecular Physics.* Vol. 15, No. 5, 1968, pp. 479-490.

[4] Papousek D., Aliev M.R., *Molecular Vibrational-Rotational Spectra,* Academia, Prague, 1982.

[5] Carter S., Handy N.C., The variational method for the calculation of ro-vibrational energy levels. *Computer Physics Reports.* No. 5, 1986, pp. 115-172.

[6] Muñoz-Caro C., Niño A., Moule D.C., Theoretical determination of the torsión-wagging structure of the $S_1 \leftarrow S_0$ electronic spectrum of acetaldehyde. *Journal of Chemical Physics*, 1994**,** *186*, pp. 221-231.

[7] Kirk H.W., Use the Decision Tables in Computer Proggramming. *Communications of the ACM*, Vol. 8, No. 1, 1965, pp. 41-43.

[8] Press L.I., Conversion of Decision Tables To Computer Programs. *Communications of the ACM*, Vol. 8, No. 6, 1965, pp. 385-390.

[9] Pollack S.L., Conversion of Limited-Entry Decision Tables to Computer Programs. *Communications of the ACM*, Vol. 8, No. 11, 1965, pp. 677-682.

[10] Reinwald L.T., Soland R.M., Conversion of Limited-Entry Decision Tables to Optimal Computer Programs I: Minimum Average Processing Time. *Journal of the Association for Computing Machinery*. Vol. 13, No. 3, 1966, pp. 339-358.

[11] Shwayder K., Conversion of Limited-Entry Decision Tables to Computer Programs –A Proposed Modification to Pollack's Algorithm, *Programming Techniques, Communications of the ACM*, Vol. 14, No. 2, 1971, pp. 69-73.

[12] Cavouras J.C., On the Conversion of Programs to Decision Tables: Method and Objectives. *Communications of the ACM*, Vol. 17, No. 8, 1974, pp. 456-462.

[13] Pooch U.W., Translation of Decision Tables. *Computing Surveys.* Vol. 6, No. 2, 1974, pp. 125-151.

[14] Shwayder K., Combining Decision Rules in a Decision Table. *Communications of the ACM*, Vol. 18, No. 8, 1975, pp. 476-480.

[15] Schumacher H., Sevcik K.C., The Synthetic Approach to Decision Table Conversion. *Communications of the ACM*, Vol. 19, No. 6, 1976, pp. 343-351.

[16] Nijssen S., Fromont E., Mining Optimal Decision Trees from Itemset Lattices, *KDD'07* August 12-15, *ACM*, San Jose, California, USA, 2007, pp. 530-539.

[17] Liu H., Feng B., Wei J., An Effective Data Classification Algorithm Based on the Decision Table Grid. *Seventh IEEE/ACIS International Conference on Computer and Information Science*, 2008, pp. 306-311.

[18] Chen Y., Xu W., Sundaram H., Rikakis T., Liu S.-M., A Dynamic Decision Network Framework for Online Media Adaptation in Stroke Rehabilitation, *ACM Transactions on Multimedia Computing, Communications and Applications,* Vol. 5, No. 1, 2008, pp. 4:1-38.

[19] Smith J.A., RNA Search with Decision Trees and Partial Covariance Models, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, Vol. 6, No. 3, 2009, pp. 517-527.

[20] Srivastava P.R., Patel P., Chatrola S., Cause effect graph to decision table generation. *ACM SIGSOFT Software Engineering Notes*. Vol. 34, No. 2, 2009, pp. 1-4.

[21] Shannon C.E., A Mathematical Theory of Communication. *The Bell System Technical Journal*, Vol. 27, 1948, pp. 379-423, 623-656.

[22] Gray R.M., *Entropy and Information Theory*, Springer-Verlag, New York, 2009, on-line version http://ee.stanford.edu/~gray/it.html, last visit January, 2010.

[23] Castro M.E., Muñoz-Caro C., Niño A., Information theory-based code optimization of matrix elements for overall rotation angular momenta. *Recent Advances in Biology, Biophysics, Bioengineering and Computational Chemistry. Proceedings of the 3rd. WSEAS International Conference on Computational Chemistry (COMPUCHEM'09)*, WSEAS Press, 2009, pp. 45-50.

[24] Podolsky B., Quantum-mechanically correct form of Hamiltonian function for conservative systems, *Physical Review*, Vol. 32, 1928, pp. 812-816.

[25] Pickett H.M., Vibration-Rotation Interactions and the Choice of Rotating Axes for Polyatomic Molecules. *Journal of Chemical Physics*, Vol. 56, No. 4, 1972, pp. 1715-1723.

[26] Harthcock M.A., Laane J., Calculation of kinetic Energy Terms for the Vibrational Hamiltonian: Application to Large-Amplitude Vibrations Using One-, Two-, and Three-Dimensional Models. *Journal of Molecular Spectroscopy*, Vol. 91, 1982, pp. 300-324.

[27] Harthcock M.A., Laane J., Calculation of Two-Dimensional Vibrational Potential Energy Surfaces Utilizing Prediagonalized Basis Set and Van Vleck Perturbation Methods. *Journal of Physical Chemistry*, Vol. 89, 1985, pp. 4231-4220.

[28] Castro M.E., Niño A., Muñoz-Caro C., Gmat. A software tool for the computation of the rovibrational G matrix. *Computer Physics Communications*, Vol. 180, 2009, pp. 1183-1187.
[29] Shaffer W.H., Operational Derivation of the Wave Functions for a Symmetrycal Rigid Rotor, *Journal of Molecular Spectroscopy,* Vol. 1, 1957, pp. 69-80.
[30] Burkhard D.G., Factorization and Wave Functions for the Symmetric Rigid Rotator. *Journal of Molecular Spectroscopy,* Vol. 2, 1958, pp. 187-202.