On the Hyperfine Structure of Iodine:

1.

How to Calculate

Hyperfine Transition Energies

by

Susanne Fredin-Picard
Bureau International des Poids et Mesures

On the Hyperfine Structure of Iodine:

1.

How to Calculate

Hyperfine Transition Energies

by Susanne Fredin-Picard

Abstract

A detailed description how to calculate hyperfine line positions in iodine, using known constants and already developed algebraic expressions for the interaction terms is given. A computer program in Turbo-Pascal has been written for this purpose. Its accuracy has been compared to earlier calculations found in the literature.
# Table of Contents

1 Introduction ................................................................. 5

2 Molecular Iodine .......................................................... 8

3 Origin of Saturated-Absorption Hyperfine Spectra .................. 12  
  3.1 Iso-spin in Iodine and Hyperfine Structure ..................... 12  
  3.2 Saturated Absorption Main Lines ............................... 14  
  3.3 Cross-over Lines .................................................... 14

4 Interaction Terms of Hyperfine Splitting in Homonuclear Diatomic Molecules .... 18

5 An Example of Hyperfine Spectra: the R(98) 58-1 B-X Transition .......... 20  
  5.1 How to Find the Hyperfine States ............................. 20  
  5.2 How to Create the Hamiltonian Matrix and to Find the Energy Eigenvalues .................................................. 21  
  5.3 Numerical Procedure ............................................. 24  
  5.4 Comparison between BIPM Programme and Earlier Calculations .......... 25

6 Conclusion ..................................................................... 26

7 References ...................................................................... 27

Appendix I-IV
1 Introduction

The observation of hyperfine structure at high resolution, hidden by the Doppler effect, was considerably improved by the use of laser, a powerful light source which offers coherence, polarization and a good definition in wavelength. Numerous studies on the hyperfine structure in molecular iodine have been reported in the scientific literature since Ezekiel and Weiss [1] first observed hyperfine components in I$_2$ using an Ar$^+$ laser in a molecular beam experiment. The technique of saturated-absorption [3,4] is frequently used in order to resolve hyperfine spectra in atoms and molecules, but other methods of Doppler-free spectroscopy, like polarization spectroscopy [5], have also been successfully applied.

In connection with the new definition of the metre 1983 at the 17$^{th}$CGPM [6], five recommended frequencies were given to be used for its realization [7]. These correspond to precisely measured hyperfine transitions in iodine and methane that coincide with commonly used laser lines. High precision in mechanics, optics and electronics is consequently requested to stabilize the laser frequencies to the transitions. The most common means of frequency stabilization is probably the third-derivative locking technique [8]; by modulating the laser frequency, the third harmonic of the saturated-absorption signal from the reference-gas is detected. Often this sort of laser stabilization is made in one step, directly on to the saturated component. This gives a typical frequency reproducibility of the order of 1 part in $10^{11}$ [9]. Methods which stabilize a frequency modulated laser in two steps, correcting for short-term and long-term variations, were introduced in the second half of the seventies [10,11] and offer a frequency reproducibility of about 1 part of $10^{12}$. Using high finesse gas-cavities, frequency reproducibilities can nowadays be extended to 1 part in $10^{13}$ or even $10^{14}$ [12-14]. Using a fast external electro-optic modulator in the laser system, Hall et al. [15] and Bjorklund [16] obtained an improved signal-to-noise ratio by their optical phase modulation/rf sideband technique in the beginning of the eighties. This technique is becoming more common. Recently, Salomon at al. [17] obtained a relative frequency stability in the

---

1 A brief review of milestones in the research of hyperfine structure of I$_2$ can be found in Ref. [2].

2 Conférence Générale des Poids et Mesures
millihertz region by experiments in which two 633 nm He-Ne lasers were stabilized to the
same high-finesse Fabry-Perot cavity. The latest possibilities for a precise frequency source
concern the work being done on cooled and trapped atoms and ions [18,19].

Whatever the method chosen for frequency stabilization, one must also control the influence
of experimental parameters on the measured hyperfine frequencies. Investigations of
hyperfine frequency as a function of factors like the pressure in the reference gas [20], the
laser frequency modulation amplitude [21], the laser power [21], the wave front geometry
[22], direction of laser power output [23], gas-lens effects [24], and pollution of the reference
gas [25-28] have already been carried out.

Beside experimental progress in the study of hyperfine structure, a theory that describes
the dynamics and interactions involved has simultaneously been developed for molecules
[29-33]. One can nowadays calculate hyperfine spectra (line positions, line widths and line
strengths) with high accuracy. In this way one can carry forward the understanding of the
hyperfine structure and its perturbations.

In this report we concentrate on the hyperfine structure of iodine, which is used for fre­
quency stabilization at the BIPM3. Two reports on calculated hyperfine spectra, presented by
our former co-worker Michael Gläser, have already appeared in this series [34,35]. For this
purpose, Gläser used computer programmes put at his disposition by Foth at the PTB4
[36,37]. These programmes are no longer available so it seemed convenient, once and for all,
to have these programmes on call at the BIPM. Calculated data are necessary, for example,
for the identification of new hyperfine transitions. The aim of this report is to give a detailed
description how to calculate hyperfine line positions with departure from known constants
and already developed expressions for the interaction terms. This report is also a contribution
to maintain a continuity of the earlier studies performed at BIPM. A modified version of the
programme presented here is intended for the fitting of a set of hyperfine constants to a set of
hyperfine components. This will be reported later. For those who does not every day confront
problems on hyperfine structure and other subjects touching molecular spectroscopy, a brief

3 At BIPM, stabilized lasers, corresponding to the following wavelengths, are at the moment
available: 3.39 um He-Ne(CH4), 633 nm He-Ne(I2), 612 nm He-Ne(I2), 543 nm He-Ne(I2)
and 515 nm Ar*(I2).

4 Physikalisch-Technische Bundesanstalt

6
review of the most currently concepts in this field has earlier been presented in this report series [38]. Additionally, assembled data on hyperfine constants of molecular iodine are listed in Ref. [39].

When choosing programming language, Turbo-Pascal seemed to be an attractive possibility. Turbo-Pascal is well adapted to micro-computers, offers a short compilation time, and one can find a large number of useful subroutines already written for the commercial market. The programme that has been written is listed in Appendix II.

In the next section we describe aspects of the structure of molecular iodine that are important for our particular use. Section 3 reviews the origin of hyperfine structure as it relates to the saturated-absorption technique. Algebraic expressions for the principal energy terms contributing to the hyperfine splitting are listed in Sec. 4, and in Sec. 5 we apply the algebraic expressions for the terms to a transition in $^{127}\text{I}_2$. 
2 Molecular Iodine

Iodine was discovered in 1811 by Courtois, the French chemist, who also discovered the morphine. There are 23 known iodine isotopes of which the only naturally abundant isotope is $^{127}$I. Iodine is particulaerly useful in the field of medicine and is especially important for the functioning of the tyroid gland and the production of thyroxine. In metrology, in work on length standards, $^{127}$I$_2$ is used for frequency stabilization. Sometimes $^{129}$I$_2$, which has a very long half-time, is also used for this purpose.

An extensive review of the spectroscopy of I$_2$ has been given by Mulliken [40]. The most important publications and spectroscopic data up to 1977 are listed by Huber and Herzberg [41]. More recently, Gerstenkorn and Luc observed a large number of lines in the visible wavelength region [42]. They were detected by Fourier transform spectroscopy and originated from the $B^3Π_u$ $→$ $X^1Σ_g^−$ system in $^{127}$I$_2$. Because of the extensive number of data and high resolution, the B-X system could be unambigously analyzed [43] so precise rotational and vibrational molecular constants were calculated [44]. As this spectrum is very dense, there is a high probability of finding a ro-vibrational transition that is resonant with a laser line. This possibility has been realized on numerous occasions: parts of the hyperfine structure emanating from a wide range of ro-vibrational transitions that coincides with commonly used He-Ne, Ar$^+$ ion or Kr$^+$ laser lines have been observed in the B-X system in I$_2$, see for example [2,45,46].

Using well established nomenclature [47], the energy term $G(v)$ of a vibrational level can be written as

$$G(v) = ω_v ρ \left( v + \frac{1}{2} \right) - ω_x ρ^2 \left( v + \frac{1}{2} \right)^2 + ... \quad (1)$$

where the vibrational quantum number $v=0,1,2,...$ and $ω_v$ and $ω_x$ are vibrational constants. The isotopic correction factor $ρ$ is defined as the square-root of the fraction between the reduced masses [47], that is, $ρ=1$ for $^{127}$I$_2$ and $ρ$ is slightly less than 1 for $^{129}$I$_2$.

The energy term $F_v(J)$ of a rotational level can be written as
\[ F_r(J) = B_r J(J + 1) - D_r J^2(J + 1)^2 + \ldots \]  

where \( B_r \) and \( D_r \) are rotational constants. The total energy \( T_r(J) \) can be written as

\[ T_r(J) = T_e + G(v) + F_r(J) \]  

where \( T_e \) represents the vertical electronic energy. The energies \( T_e, G(v) \) and \( F_r(J) \) are represented in Fig.1-a.

**Fig.1-a.** Representation of the potential energy curves of the B and X states, where the electronic \( (T_e) \), vibrational \( (G(v)) \) and rotational \( (F_r(J)) \) energy contributions are represented. \( R \) is the internuclear distance.
The selection rule for J in the B-X system is

$$\Delta J = J^\prime - J^\parallel = \pm 1$$

(4)

where the ('') and ('') indicate upper and lower state respectively. The condition for $\Delta v$ is not restricted (except that the intensity of a $\Delta v$ transition depends on the vibrational wavefunction overlap; the Franck-Condon factors).

When saturated-absorption is obtained in an iodine-glass-cell within the laser cavity, ground state vibrational numbers $v''=5$ or $v''=6$ can be achieved due to the strong internal laser intensity. However, when the iodine-glass-cell is placed outside the laser cavity, only $v''=0$ or $v''=1$ can be excited as a consequence of the Boltzmann distribution at room-temperature. An example of some ro-vibrational transitions is schematically represented in Fig.1-b.
Fig. 1-b. Three examples of ro-vibrational transitions of the B-X system of I₂: R(98) 58-1, P(13) 43-0 and R(15) 43-0. These lines can be excited by the Ar⁺ laser line at 515 nm.
3 Origin of Saturated-Absorption Hyperfine Spectra

3.1 Iso-spin in Iodine and Hyperfine Structure

Let $I_i$ represent the intrinsic spin of each nucleus in a diatomic molecule (like iodine). In $^{127}\text{I}$ the intrinsic spin is $I_i=5/2$, and in the isotope $^{129}\text{I}$, $I_i=7/2$. Using the sum-rules for angular momenta, the total nuclear spin $I$ becomes $I = 0, 1, 2 \ldots 5$ for $^{127}\text{I}$. Due to the interaction between the total molecular angular momentum and the iso-spin, $J$ and $I$ couple and form the angular momentum $F$. If $J>I$ we get

$$F = |J-I|, |J-I+1| \ldots |J+I-1|, |J+I|$$

(5)

The levels labelled by $F$ and $I$ are generally not degenerate (see Sec.4); they constitute a set of $(2I+1)$ sublevels to $J$. The hyperfine structure is illustrated in Fig.2 for one level $J$.

Fig.2. Hyperfine structure, labelled by $F$ and $I$, originating from one rotational level, labelled $J$. 
Because of the symmetries of the involved wavefunctions, states are named ortho and para when J is odd or even. Ortho states are defined by I=1,3,5, while para states have I=0,2,4 in $^{127}$I$_2$. This is summerized for the B and the X states in Tab.1, taken from the thesis by Camy [48]. With a good approximation $\Delta I=0$

<table>
<thead>
<tr>
<th>electronic states</th>
<th>ortho-states</th>
<th>para-states</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{B}^3\Pi_{0}^+$</td>
<td>J even</td>
<td>J odd</td>
</tr>
<tr>
<td>$\text{X}^1\Sigma^+_g$</td>
<td>J odd</td>
<td>J even</td>
</tr>
</tbody>
</table>

For hyperfine transitions, that is, transitions between hyperfine sub-levels belonging to different electronic states, the strongest transitions are observed when $\Delta F=\Delta J$ if $J>>I$. These transitions are called 'main lines'. In iodine we then have $\Delta F=\Delta J=\pm 1$ (see Eq.(4)). Less probable are the $\Delta F=0$ transitions, which are about $1/2J^2$ less intense than $\Delta F=\Delta J$. One can also have the combination $\Delta F=-\Delta J$, which describes transitions more or less forbidden, with an intensity $1/10J^4$ less than that for $\Delta F=\Delta J$. Using saturated-absorption, so called 'cross-over lines' can also be observed in which the intensity is about $1/2J$ of that for the $\Delta F=\Delta J$ transitions. These features are described in Sec.3.3. We expect only transitions obeying $\Delta F=\Delta J$ for high values of J.

The main lines are sometimes named a, b, c, d... in decreasing frequency order. It has also been suggested that the components should be named in increasing frequency order [49], a$_1$, a$_2$, a$_3$... This we have chosen to do here.
3.2 Saturated Absorption Main Lines

For a detailed description of the saturated-absorption process, Ref.[8] is recommended. Here we give a short description emphasising only its principal features.

Let a gas containing molecules (or atoms) be excited by two counter propagating laser beams originating from the same laser with frequency $f_0$. As the molecules have thermal motion, the frequency will appear as $f = f_0(1 \pm v_z/c)$ to a molecule, where $v_z$ is the velocity component of the molecule in the laser beam direction, and $c$ is the velocity of light. This means that different molecules, moving in different directions, will interact differently with the incoming and outgoing laser photon frequency. The frequency will appear the same for the incoming and outgoing laser beam if $v_z = 0$ ($f = f_0$). In this case the same molecule can interact with incoming and outgoing laser photons of the same frequency, which also means that such molecules are exposed to a stronger laser field than are molecules with $v_z \neq 0$. If the laser intensity is sufficiently strong, the lower state will be depleted (or the excited state will saturate), so the absorption coefficient is reduced. The excess photons that do not participate in the excitation process pass through the gas without being absorbed. The line profile of this transmitted light conforms with the natural linewidth of the absorbing transition in the ideal case. The decrease in absorption coefficient is sometimes called the 'Bennet hole' (for any $v_z$) or the 'Lamb dip' (for $v_z = 0$) and the increase in transmitted light is sometimes called the 'inverse Lamb-dip'. All these processes are referred to as 'hole burning'.

3.3 Cross-over Lines

Cross-over lines are lines related to ordinary saturated-absorption lines in which molecules with $v_z \neq 0$ contribute to saturated-absorption signals. This is illustrated by Fig.3-a which shows transitions with a common lower level. As the resonant frequency $f$ of the molecule appears as $f = f_0(1 \pm v_z/c)$ relative to the laser photon frequency, two hyperfine
levels can be simultaneously excited if they are separated by $2f_0 \nu/c$ and the excited molecules moves with the velocity $+v$, or $-v$: for the incoming direction one transition can be excited; for the outgoing direction the other transition in the same molecule can be excited. As the both transitions originate from the same state, this state will be depleted; the absorption coefficient is thus reduced. The excess laser light is not absorbed, and a signal can be observed at a frequency which corresponds to half the sum-frequency of the two transitions\(^5\). The strongest transitions that can contribute to cross-over lines are those for $\Delta F=\Delta J$ and $\Delta F=0$. In terms of intensity factors given in Sec. 3.1 we note that cross-over lines are expected only for rather low $J$-values. The number of possible cross-over lines is twice as great as the number of $\Delta F=0$ lines. In Fig. 3-b an energy level scheme is shown where the transitions have a common upper level.

\[ \text{Fig. 3-a. Scheme of energy levels involved to obtain a cross-over resonance: The straight arrows represent the laser photon energy in the molecule frame on its way towards/away from the laser source. The wavy arrow represent the laser photon energy in the laboratory frame. These photons passes the sample when saturation is obtained.} \]

\(^5\) The recoil effect has not been taken into account here. Bordé et al. [50] have studied this aspect in iodine excited by a 515 nm Ar\(^+$\) laser.
Fig. 3-b. Similar scheme to that of Fig. 3-a but where the common level is the upper one.
In Tab.2a-b the number of lines corresponding to ortho and para states are listed [48]. One can observe that for an ortho state transition altogether 90 lines are present!

### Table 2-a.

<table>
<thead>
<tr>
<th></th>
<th>I=1</th>
<th>I=3</th>
<th>I=5</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta F = \Delta J$</td>
<td>3</td>
<td>7</td>
<td>11</td>
<td>21</td>
</tr>
<tr>
<td>$\Delta F = 0$</td>
<td>2</td>
<td>6</td>
<td>10</td>
<td>18</td>
</tr>
<tr>
<td>$\Delta F = -\Delta J$</td>
<td>1</td>
<td>5</td>
<td>9</td>
<td>15</td>
</tr>
<tr>
<td>cross-over</td>
<td>4</td>
<td>12</td>
<td>20</td>
<td>36</td>
</tr>
</tbody>
</table>

### Table 2-b.

<table>
<thead>
<tr>
<th></th>
<th>I=0</th>
<th>I=2</th>
<th>I=4</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta F = \Delta J$</td>
<td>1</td>
<td>5</td>
<td>9</td>
<td>15</td>
</tr>
<tr>
<td>$\Delta F = 0$</td>
<td>0</td>
<td>4</td>
<td>8</td>
<td>12</td>
</tr>
<tr>
<td>$\Delta F = -\Delta J$</td>
<td>0</td>
<td>3</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>cross-over</td>
<td>0</td>
<td>8</td>
<td>16</td>
<td>24</td>
</tr>
</tbody>
</table>
4 Interaction Terms of Hyperfine Splitting in Homonuclear Diatomic Molecules

To calculate the energies of the hyperfine components we first have to derive the Hamiltonian: the Hamiltonian for hyperfine interaction, $H_{\text{HF}}$, can be written as [36]

$$H_{\text{HF}} = H_{\text{EQ}} + H_{\text{SR}} + H_{\text{SSS}} + H_{\text{TSS}}$$

where $H_{\text{EQ}}$ and $H_{\text{SR}}$ represent the electric quadrupole and spin-rotation interaction respectively. The terms $H_{\text{SSS}}$ and $H_{\text{TSS}}$ represent scalar spin-spin and tensor spin-spin interactions respectively. Additional higher order terms can be included [2], but these we neglect. The algebraic expressions for the eigenvalues of $H_{\text{EQ}}$, $H_{\text{SR}}$, $H_{\text{SSS}}$ and $H_{\text{TSS}}$ are listed below and are taken from Foth and Spieweck [36]. The big parentheses represent 3-j symbols, small and large curls brackets correspond to 6-j and 9-j symbols.

$$< F, I, J | H_{\text{EQ}} | F', I', J'> = (-1)^{F+J+2J'} \frac{1}{2} e Q(J,J') [(2J+1)(2J'+1)]^{1/2} \times$$

$$\times \begin{vmatrix} I_1 & I_1 & 2 \\ I_1 & -I_1 & 0 \end{vmatrix} \begin{vmatrix} J' & J & 2 \\ J & -J & 0 \end{vmatrix}^{-1} \begin{vmatrix} F & J & I \\ I & I' & I \end{vmatrix} \begin{vmatrix} F & J & I \\ I & I' & I \end{vmatrix}$$

$$< F, I, J | H_{\text{SR}} | F', I', J'> = \delta_{J J'} \delta_{I I'} \frac{1}{2} C [F(F+1) - I(I+1) - J(J+1)]$$

6 The labels SSS and TSS have been inversed in [36], and a missprint has been copied from Bunker and Hanes [30] in the 6-j of the $H_{\text{TSS}}$. The latter was pointed out by Broyer et al. [32] but has no significance in these calculations.
\[ <F, I, J | H_{ssl} | F, I', J'> = \delta_{i'j} \delta_{ii'} \frac{1}{2} A [I(I+1) - 2I_1(I_1+1)] \] (9)

\[ <F, I, J | H_{ssl} | F, I', J'> = \delta_{ij} (-1)^{F+i'1} D(2J+1)[I_1(I_1+1)(2I_1+1)] [30(2I+1)(2I'+1)]^{1/2} \times \]
\[ \times \begin{pmatrix} J & 2 & J \\
0 & 0 & 0
\end{pmatrix} \begin{pmatrix} F & J & I \\
I_1 & I_1 & 1
\end{pmatrix} \begin{pmatrix} I_1 & I_1 & 1 \\
I & I & 2
\end{pmatrix} \] (10)

eq Q(J', J), C, A and D represent the constants for the electric quadrupole, spin-rotation, scalar spin-spin and tensorial spin-spin interactions respectively.

It has been shown that for electric quadrupole interaction, one has to consider the interaction between states separated by \( \Delta I = \pm 2 \) \cite{45}. Sometimes it is even necessary to consider states separated by \( \Delta J = \pm 4 \) or even \( \Delta J = \pm 8 \), but we will here only calculate elements where \( \Delta J = 0 \) and \( \pm 2 \). As a consequence, one must add the rotational energy contribution, given by Eq.(2), to each diagonal element before treating the Hamiltonian matrix.
5 An Example of Hyperfine Spectra: the R(98) 58-1 B-X Transition

We exemplify the calculation of hyperfine lines by studying the R(98) 58-1 transition in the B-X system of $^{127}$I$_2$. This line coincides with the 515 nm laser line of Ar$^+$, and its hyperfine structure has been observed by several groups [11,36]. We here refer mainly to the work reported by Foth and Spieweck [36].

5.1 How to Find the Hyperfine States

For this R line we know that $J"=98$ and $J'=99$. We here only consider transitions that obey the selection rule $\Delta F=\Delta I$, that is, the strongest transitions. Let us label the transitions using the lower state quantum numbers. As both the lower and upper states can be defined as para states, we know that $I=0$, 2 and 4. Using the addition rules given by Eq.(5) we hence obtain

$$2I+1$$

| $F(I=0)$ | 98 | 1 |
| $F(I=2)$ | 96, 97, 98, 99, 100 | 5 |
| $F(I=4)$ | 94, 95, 96, 97, 98, 99, 100, 101, 102 | 9 |

We obtain 15 components since both the upper and lower state contain 15 hyperfine components, so giving 15 hyperfine lines. However, due to the electric quadrupole interaction, states which are separated with $\Delta I=\pm 2$ will interfere. The listing of components that have to be considered in our calculation is then:
5.2 How to Create the Hamiltonian Matrix and to Find the Energy Eigenvalues

We now need to calculate each matrix element of the Hamiltonian. Using Eqs (6-10) and (2) this is a straight-forward matter. The character of the 3-j, 6-j and 9-j symbols have been described by, for example, Edmonds [51] and Brink and Satchler [52]. The expressions for $H_{SR}$ have been explicitly written in the appendix of Ref. [45]. Further guidelines on the 9-j symbols are given in [2]. Note that $H_{SR}$ and $H_{SSS}$ are purely diagonal.
We hence treat two 45x45 matrices, one for the upper state and one for the lower one. By labelling the matrix by consecutive $F$, this matrix becomes block-diagonal; it contains 13 sub-matrices where each one has the dimension $N$ (see Sec.5.1) and is labelled by its specific quantum number $F$. Each matrix element within a sub-matrix is labelled by its quantum number $J$ and $I$. To calculate the diagonalized 45x45 matrix we diagonalize each sub-matrix which is, evidently, more convenient. The diagonal elements will then represent the energy of the upper (lower) state. We are only interested in the diagonal matrix elements that have $J=J'$ (or $J=J''$). Fig. 4 shows schematically a part of the hyperfine splitting and some allowed main hyperfine transitions for the R(98) 58-1 line.
Fig. 4. Schematic representation of the transitions between some of the hyperfine components of the upper and lower state respectively. Note that the energy of each level is only symbolic and does not correspond with the energy of the components in reality.
5.3 Numerical Procedure

Appendices I-III show step by step the calculation where the line R(98) 58-1 is used as an example. Below is listed the contents of each appendix.

Appendix I-a  
Dimension and labels F, J and I for each submatrix of the upper state hyperfine Hamiltonian for the R(98) 58-1 line.

Appendix I-b  
Dimension and contents of each sub-matrix.

Appendix I-c  
Dimension and diagonal elements of each diagonalized sub-matrix, corresponding to the labels listed in App.I-a. The energies are given in MHz, and J=J' are marked with an arrow in the right margin.

Appendix II-a  
Main programme written in Turbo-Pascal 3.0.

Appendix II-b  
Subroutines for calculating the Hamiltonian matrix and deducing the energy differences.

Appendix II-c  
Subroutines for calculating the rotational contribution.

Appendix II-d  
Subroutines to diagonalize the symmetrical matrix and to calculate the energy eigenvalues.

Appendix III-a  
Format of input file.

Appendix III-b  
Appearance on screen during a run.

Appendix III-c  
Output file where ΔF=ΔJ, ΔF=0 and cross-over lines are listed in MHz. The lines where the frequency is 0.000 do not exist.
5.4 Comparison between BIPM Programme and Earlier Calculations

We have compared the calculations made by our programme with four different calculated structures where three programmes of different origin have been used. These calculations have appeared in works by

a) Foth and Spieweck [36]
b) Bordé et al. [2]
c) Razet [53].

These comparisons are listed in App.IV. In general, the differences between the BIPM calculations and the others differ by less than 1 kHz. This can be explained by round-off errors. However, one discrepancy has been noted: it seems to us that I=0 and I=2 for F=J have been reversed in Tab.1 of Foth and Spieweck [36]. These levels belong to para states. This has no influence on the calculated frequency of the main lines. Unfortunately, we believe that Gläser et al. [54] used the same programme source yet our frequencies differ from theirs for the ΔF=0 and the cross-over lines, when the levels associated to I=0 and I=2 for F=J are involved. We think this may arise from the reversal and, in confirmation, note that compatible values are found for ortho states.
6 Conclusion

A programme that calculates hyperfine lines using already known constants has been written in Turbo-Pascal. It has been tested and reproduce earlier calculations made by other groupes within an error of less than 1 kHz.

Acknowledgement

I am grateful to Dr L. Robertsson for kindly having aquainted me with Turbo-Pascal, and Mrs D. Le Coz for faithfully having assisted in the literature research. I also thank Dr. L. Sturesson (Lund, Sweden) for stimulating discussions.
References

[34] M. Gläser, Rapport BIPM -82/12.


<table>
<thead>
<tr>
<th></th>
<th>F</th>
<th>J</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>93</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>94</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>95</td>
<td>97</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>95</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>95</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>96</td>
<td>97</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>96</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>96</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
<td>97</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
<td>97</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
<td>99</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
<td>101</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>98</td>
<td>97</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>98</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>98</td>
<td>99</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>98</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>98</td>
<td>101</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>99</td>
<td>97</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>99</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>99</td>
<td>99</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>99</td>
<td>99</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>99</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>99</td>
<td>101</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>99</td>
<td>101</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>99</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>101</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>101</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>101</td>
<td>97</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>101</td>
<td>99</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>101</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>101</td>
<td>101</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>101</td>
<td>101</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>101</td>
<td>101</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>102</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>10</td>
<td>102</td>
<td>101</td>
<td>2</td>
</tr>
<tr>
<td>10</td>
<td>102</td>
<td>101</td>
<td>4</td>
</tr>
<tr>
<td>11</td>
<td>103</td>
<td>99</td>
<td>4</td>
</tr>
<tr>
<td>11</td>
<td>103</td>
<td>101</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>103</td>
<td>101</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>104</td>
<td>101</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>105</td>
<td>101</td>
<td>4</td>
</tr>
</tbody>
</table>

AI-a
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>-13.805212</td>
<td>0.003547</td>
<td>-0.009439</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.003547</td>
<td>-13.802989</td>
<td>-0.002714</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.009439</td>
<td>-0.002714</td>
<td>-0.028364</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-13.791770</td>
<td>0.005068</td>
<td>-0.006742</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.005068</td>
<td>-13.796598</td>
<td>-0.004113</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.006742</td>
<td>-0.004113</td>
<td>-0.023830</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>-13.785336</td>
<td>0.009426</td>
<td>0.000000</td>
<td>-0.011373</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>0.009426</td>
<td>-13.781805</td>
<td>0.005609</td>
<td>0.004276</td>
<td>-0.004458</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>0.005609</td>
<td>-13.789152</td>
<td>-0.001072</td>
<td>-0.020519</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>-0.004458</td>
<td>-0.004966</td>
<td>0.003548</td>
<td>-0.018300</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>-0.009438</td>
<td>-0.002716</td>
</tr>
<tr>
<td>5</td>
<td>-13.775431</td>
<td>0.005173</td>
<td>0.005291</td>
<td>-0.002600</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>0.005173</td>
<td>-13.780620</td>
<td>-0.002421</td>
<td>-0.005289</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>0.005291</td>
<td>-0.002421</td>
<td>-0.006910</td>
<td>0.005069</td>
<td>-0.006740</td>
</tr>
<tr>
<td></td>
<td>-0.002600</td>
<td>-0.005289</td>
<td>-0.001174</td>
<td>-0.004115</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
<td>-0.006740</td>
<td>-0.004115</td>
<td>13.972413</td>
</tr>
<tr>
<td>7</td>
<td>-13.772760</td>
<td>0.003695</td>
<td>-0.011258</td>
<td>0.004364</td>
<td>-0.001174</td>
</tr>
<tr>
<td></td>
<td>0.003695</td>
<td>-13.770969</td>
<td>0.000000</td>
<td>-0.004238</td>
<td>-0.005068</td>
</tr>
<tr>
<td></td>
<td>-0.011258</td>
<td>0.000000</td>
<td>-0.000312</td>
<td>0.009426</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>0.004364</td>
<td>-0.004238</td>
<td>0.009426</td>
<td>0.003219</td>
<td>0.005609</td>
</tr>
<tr>
<td></td>
<td>-0.001174</td>
<td>-0.005068</td>
<td>0.005609</td>
<td>-0.004129</td>
<td>-0.001073</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
<td>-0.011372</td>
<td>0.004277</td>
<td>-0.001073</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>-0.004456</td>
<td>-0.004967</td>
</tr>
<tr>
<td>5</td>
<td>-13.760170</td>
<td>-0.006540</td>
<td>-0.004283</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>-0.006540</td>
<td>0.009757</td>
<td>0.005172</td>
<td>0.005291</td>
<td>-0.002598</td>
</tr>
<tr>
<td></td>
<td>-0.004283</td>
<td>0.009757</td>
<td>0.005172</td>
<td>0.004572</td>
<td>-0.002423</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>0.005291</td>
<td>-0.002423</td>
<td>13.989670</td>
<td>0.005070</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>-0.002598</td>
<td>-0.005289</td>
<td>0.005070</td>
<td>13.984834</td>
</tr>
<tr>
<td>6</td>
<td>-13.748189</td>
<td>-0.009343</td>
<td>-0.002864</td>
<td>0.000000</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>-0.009343</td>
<td>0.012594</td>
<td>0.003694</td>
<td>-0.011259</td>
<td>0.004363</td>
</tr>
<tr>
<td></td>
<td>-0.002864</td>
<td>0.012594</td>
<td>0.003694</td>
<td>0.014369</td>
<td>0.000000</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>-0.011259</td>
<td>0.000000</td>
<td>13.996432</td>
<td>0.009426</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>0.004363</td>
<td>-0.004240</td>
<td>0.009426</td>
<td>13.999963</td>
</tr>
<tr>
<td></td>
<td>0.000000</td>
<td>-0.001173</td>
<td>-0.005067</td>
<td>0.000000</td>
<td>0.005609</td>
</tr>
<tr>
<td>3</td>
<td>0.025355</td>
<td>-0.006542</td>
<td>-0.004282</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.006542</td>
<td>14.006666</td>
<td>0.005171</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.004282</td>
<td>0.005171</td>
<td>14.001483</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.037499</td>
<td>-0.009344</td>
<td>-0.002863</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.009344</td>
<td>14.009669</td>
<td>0.003692</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-0.002863</td>
<td>0.003692</td>
<td>14.011468</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>14.022600</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>14.034908</td>
<td>AI-b</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
-1.38127230000000E+005
-1.38083530000000E+005
-1.38078201871944E+005
-1.38003878154855E+005
-2.83569973201285E+002
-1.37885749733650E+005
-1.37997955787220E+005
-2.38254687630513E+002
-1.3795078302640E+005
-1.37727986939517E+005
-1.37880005161877E+005
-2.31203970840877E+002
-1.56914587816620E+002
1.39677228962692E+005
-1.37722408970694E+005
-1.37838150846074E+005
-3.71301332427789E+001
-1.4939469508544E+002
1.39724174619520E+005
-1.37756751657725E+005
-1.37680676924358E+005
-1.04771131532650E+002
1.22276901108068E+002
-2.97254542089946E+001
1.39732923641354E+005
1.39807254625363E+005
-1.37601744373173E+005
1.29517872974233E+002
1.37674372475754E+001
1.39928714610798E+005
1.39816374452154E+005
-1.37481959468077E+005
9.68371323008827E+001
1.72925836129723E+002
1.39862735128497E+005
1.40089741850066E+005
1.39937759521083E+005
2.53506284968139E+002
1.40098629079713E+005
1.39982904635318E+005
3.74921573492518E+002
1.40067707216351E+005
1.40143731210157E+005
1.40226000000000E+005
1.40349080000000E+005
Program Hyper;

(* ************************************************** *)
(* Calculate frequencies for delta-F=delta-J *)
(* and delta-F=O hyperfine transitions. *)
(* Calculate cross-over lines. *)
(* Give line, J", v' and v", then *)
(* upper and lower hyperfine constants *)
(* in an external file. *)
(* ************************************************** *)

LABEL 20, 30, 40, 50;

CONST convert = 29979.2458;

IOerr : boolean = false;
TNArraySize = 8;

TYPE pp = RECORD
  F, JP, IP, JB, IB : integer;
END;

mattyp7 = array [1..15,1..8,1..8] of real;
atyp = array [1..63] of pp;
btyp = array [1..63] of real;
ctyp = array [1..15] of integer;
dtyp = array [1..15,1..8,1..8] of pp;
htyp = array [1..8] of real;

TNvector = array[1..TNArraySize] of real;
TNmatrix = array[1..TNArraySize] of TNvector;
TNIntVector = array[1..TNArraySize] of integer;

VAR M : mattyp7;
till, fran,
EQQ, CKON, AKON, DKON : real;
kl, ml, nl, jinit, jinit,
antal, antmat, aja,
Jbis, vprim, vbis, NofLines,
Dimen, xl : integer;
mille : TNmatrix;
slask, egen : atyp;
eget, egetu, egetl : btyp;
nostat : ctyp;
punkt : dtyp;
hkon : htyp;
bok, state : char;

($I a:S1.PAS )
($I a:ROTER.PAS )
($I a:ANROP.PAS )

(* ************************************************** *)
BEGIN

(* ************************************************** *)
(* Determine initial conditions. *)
(* Read hyperfine constants from file. *)
(* ************************************************** *)

AII-a/1
20: SetNumber(vprim, vbis, bok, Jbis, aja, hkon);
   IF aja=1 THEN GOTO 30;
   state:='L';

(* ****************************************** *)
(* Calculate energy differences between rotational *)
(* levels with delta-J=+-2. First lower, the upper. *)
(* ****************************************** *)

40: jinit:=Jbis-2;

   IF state='U' THEN
     BEGIN
       IF bok = 'R' THEN jinit:=jinit+1
       ELSE jinit:=jinit-1;
       EQQ:=hkon[1];  
       CKON:=hkon[3];
       AKON:=hkon[5];
       DKON:=hkon[7];
       till:=Diffup(vprim, jinit+4)  
           -Diffup(vprim, jinit+2);  
       fran:=Diffup(vprim, jinit+2)  
           -Diffup(vprim, jinit);
     END
   ELSE
     BEGIN
       EQQ:=hkon[2];
       CKON:=hkon[4];
       AKON:=hkon[6];
       DKON:=hkon[8];
       till:=Diffflow(vbis, Jbis+2)-Diffflow(vbis, Jbis);
       fran:=Diffflow(vbis, Jbis)-Diffflow(vbis, Jbis-2);
     END;

(* ****************************************** *)
(* Find eigen vectors. *)
(* F=(J-I)...(J+I) *)
(* ****************************************** *)

   iinit:=0;
   IF Odd(Jbis) THEN iinit:=1;
   FindEigVec(jinit, iinit, slask, antal);

(* ****************************************** *)
(* Arrange eigenvectors F,J,I in increasing order. *)
(* ****************************************** *)

   RangeEig(slask, egen, antal, jinit, iinit, 
            state, Noflines);

(* ****************************************** *)
(* Count number of eigen vectors *)
(* and find the number of sub matrices. *)
(* ****************************************** *)

   CalcEig(egen, nostat, antal, antmat);

(* ****************************************** *)
(* Make matrix containing the eigen states. *)
(* ************************************************* *)
PunktMat(punkt, egen, nostat, antmat);

(* ************************************************* *)
(* Calculate matrix. *)
(* ************************************************* *)
MatElement(antmat, nostat, punkt, M, till, fran, state, EQQ, CKON, AKON, DKON);

(* ************************************************* *)
(* Diagonalize and find eigenvalues. *)
(* ************************************************* *)
x1:=0;
FOR nl:=1 TO antmat DO
BEGIN
  Dimen:=nostat[nl];
  FOR ml:=1 TO Dimen DO
    FOR kl:=1 TO Dimen DO
      mille[ml, kl]:=0.0001*M[nl, ml, kl];
      diag(eget, Dimen, x1, mille);
  END;
END;

IF state<>'L' THEN egetu := eget
ELSE egetl:=eget;
WRITELN(' Program is running ! ');

IF state<>'L' THEN GOTO 50;
state:=l'U';
GOTO 40;

(* ************************************************* *)
(* Calculate frequency differences. *)
(* ************************************************* *)
50: differ(egen, egetu, egetl, nostat, jinit, antmat, NofLines, vprim, vbis, Jbis, bok, hkon);
30: END.

AII-a/3
Function diodj0(a,b,c:integer):real;
(* 3j: Delta-I=0 Delta-J=0 *)
CONST  Il = 2.5;
VAR   F, I, J, d, frad, g, h : real;
BEGIN
  F:=a;
  J:=b;
  I:=c;
  frad:=F*(F+1)-(I+1)-(J+1);
  d:=-3/(16*(2*J+3)*Il*(2*Il-1));
  g:=(F+I-J+1)*(F+I-J)*F*(I+J+2)*(I+J+F+1);
  g:=g*(2*Il+1)*(2*I1+1)-(I+1)*(I+1);
  h:=(F+I-J)*(F+I-J+1)*(F+J-I+1)*I*(I+J-F-1);
  h:=h*(2*Il+1)*(2*I1+1)/(4*I*I-1);
  diodj0:=d*(frad*frad+2*frad+g+h-
              16*Il*(Il+l)*J*(J+l)/3);
END; (* Function diodj0 *)

Function dim2dj0(a,b,c:integer):real;
(* 3j: Delta-I=-2 Delta-J=0 *)
CONST  Il = 2.5;
VAR   F, I, J, p, q, x, y, z : real;
BEGIN
  F:=a;
  J:=b;
  I:=c;
  x:=3*SQRT((2*Il+1)*(2*Il+1)-(I-I));
  y:=-16*(2*J+3)*(2*J-1)*Il*(2*Il-1)*SQRT((4*I*I-1)
             *(2*I-3)*(2*I+1));
  z:=SQRT((F+I-J-1)*(F+I-J)*(F+I-J+1));
  p:=SQRT((F+J-I+2)*(F+J-I)*(F+J+I+1));
  q:=SQRT((I+J-F-1)*(I+J-F));
  dim2dj0:=(x/y)*z*p*q;
END; (* Function dim2dj0 *)

Function diOdj2(a,b,c:integer):real;
(* 3j: Delta-I=0 Delta-J=2 *)
CONST  Il = 2.5;
VAR   x, y, z, p, q, F, I, J : real;
BEGIN
\( F := a; \)
\( J := b; \)
\( I := c; \)
\( x := 3(3I(I+1) - 3 - 4I^1(I^1+1)); \)
\( y := 16(2J+3)(2I+3)(2I^1 - 1)(2I^1-1)I^1 \times \sqrt{2J+1} \times (2J+5); \)
\( z := \sqrt{F+I-J-1} \times (F+I-J) \times (F+J-I+1); \)
\( p := \sqrt{F+J-I+2} \times (F+J+I+2) \times (F+I+J+3); \)
\( q := \sqrt{I+J-F+1} \times (I+J-F+2); \)
\( d\text{io}d_{j2} := (x/y) \times z \times p \times q; \)

\( \text{END; (} \star\text{ Function d\text{i}o\text{d}_{j2} \star}\) \)

\( \text{(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)} \)

\( \text{Function d\text{i}2d\text{j}2(a,b,c:integer):real; \)} \)
\( \text{(* 3j: Delta-I=2 Delta-J=2 *)} \)
\( \text{CONST } I1 = 2.5; \)
\( \text{VAR } F, I, J, x, y, p, q : \text{real; \)} \)
\( \text{BEGIN} \)
\( F := a; \)
\( J := b; \)
\( I := c; \)
\( x := 3 \times \sqrt{(2I^1+1) \times (2I^1+3) \times (2I^1-I-1) \times (2I^1-I)}; \)
\( y := 32(2J+3)(2I^1+3)(2I^1-I-1)I^1 \times \sqrt{(2J+1) \times (2J+5) \times (2I^1+1) \times (2I^1+5)}; \)
\( p := \sqrt{(F+I+J+5) \times (F+I+J+4) \times (F+I+J+2) \times (F+I+J+3)}; \)
\( q := \sqrt{(I+J-F+1) \times (I+J-F+2) \times (I+J-F+3) \times (I+J-F+4)}; \)
\( \text{d\text{i}2d\text{j}2 := (x/y) \times p \times q; \)} \)
\( \text{END; (} \star\text{ Function d\text{i}2d\text{j}2 \star}\) \)

\( \text{(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)} \)

\( \text{Function d\text{i}m\text{d}_2d\text{j}2(a,b,c:integer):real; \)} \)
\( \text{(* 3j: Delta-I=-2 Delta-J=2 *)} \)
\( \text{CONST } I1 = 2.5; \)
\( \text{VAR } x, y, z, p, q, F, I, J : \text{real; \)} \)
\( \text{BEGIN} \)
\( F := a; \)
\( J := b; \)
\( I := c; \)
\( x := 3 \times \sqrt{(2I^1-1) \times (2I^1-I) \times (2I^1-I+1) \times (2I^1-I+3)}; \)
\( y := 32(2J+3)(2I^1+3)(2I^1-I-1)I^1 \times \sqrt{(2J+1) \times (2J+5) \times (2I^1+1) \times (2I^1+5)}; \)
\( p := \sqrt{(F+I-J+5) \times (F+I-J+4) \times (F+I-J+2) \times (F+I-J+3)}; \)
\( q := \sqrt{(I+J-F+1) \times (I+J-F+2) \times (I+J-F+3) \times (I+J-F+4)}; \)
\( \text{d\text{i}m\text{d}_2d\text{j}2 := (x/y) \times p \times q; \)} \)
\( \text{END; (} \star\text{ Function d\text{i}m\text{d}_2d\text{j}2 \star}\) \)

\( \text{(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)} \)

\( A_{II-b/2} \)
Function NioJO(a:integer):real;
(* 9j: Delta-I=0 *)
CONST I1 = 2.5;
VAR I, S : real;
BEGIN
I:=a;
S:=(4*I1*(I1+1)+I*(I+1))/(I1*(I1+1)*(2*I1+1)*SQRT(120);
NioJO:=S*SQRT((I*(I+1))/((2*I-1)*(2*I+1)*(2*I+3));
END; (* Function NioJO *)

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

Function NioJ2(a:integer):real;
(* 9j: Delta-I=2 *)
CONST I1 = 2.5;
VAR I, S : real;
BEGIN
I:=a;
S:=SQRT((2*I1+1)*(2*I1+1)-(I-1)*(I-1)*(I-1)*I/5); s:=S*SQRT(((2*I1+1)*(2*I1+1)-I*I)/((2*I-3)*(2*I-1)*(2*I+1));
NioJ2:=-S/(2*I1*(2*I1+1)*(2*I1+2));
END; (* Function NioJ2 *)

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

Function Cfunk(f1,j1,i1:integer):real;
(* HF8-SR *)
VAR C, X, F, I, J : real;
BEGIN
F: =f1;
J:=j1;
I:=i1;
X:=F*(F+1)-I*(I+1)-J*(J+1);
Cfunk:=X/2;
END; (* Function Cfunk *)

(* XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

Function Afunk(a:integer):real;
(* HFS-SSS *)
CONST I1 = 2.5;
VAR X, I : real;
BEGIN

AII-b/3
I:=a;
X:=I*(I+1)-2*I1*(I1+1);
Afunk:=X/2;
END; (* Function Afunk *)

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

Function Dfunk(fl,j1,a,b:integer):real;
(* HFS-TSS *)
CONST I1 = 2.5;
VAR C, X, F, IP, IB, J : real;
fas, s : integer;
BEGIN
fas:=1;
F:=fl;
J:=j1;
IP:=a;
IB:=b;
s:=TRUNC(0.02+J+IP+F);
IF Odd(s) THEN fas:=-1;
X:=SQRT(30*(2*IB+l)*(2*IP+l) »*(2*J+l)
*11*(11+1)*(2*11+1)
*X:=X*SQRT(J*(J+l) »(2*J+3) »(2*J+1) »(2*J-1) »);
Dfunk:=X*fas;
END; (* Function Dfunk *)

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

Function sexj0(A1,Bl,Cl:integer):real;
(* 6j: a b c *)
(* 2 c b *)
LABEL 10;
VAR fas, s integer;
a, b, c, x1, y1, z1, X, se real;
BEGIN
se:=0.0;
a:=A1;
b:=Bl;
c:=Cl;
s:=TRUNC(a+b+c+0.02);
X:=b*(b+l)+c*(c+l)-a*(a+l);
fas:=1;
x1:=3*X*(X-l)-4*b*(b+l)*c*(c+l);
IF x1=0.0 THEN GOTO 10;
y1:=(2*b+3) »(2*b+2) »(2*b+1) »2*b »(2*b-1);
z1:=(2*c+3) »(2*c+2) »(2*c+1) »2*c »(2*c-1);
IF Odd(s) THEN fas:=-1;
se:=x1/SQRT(y1*z1);
10: sexj0:=2*fas*se;
END; (* Function sexj0 *)

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

AII-b/4
Function sexj2(A1,B1,C1:integer):real;
(* 6j: a b c *)
(* 2 c-2 b *)
VAR fas, sl : integer;
a, b, c, x1, y1, z1, X, s : real;
BEGIN
  a:=A1;
b:=B1;
c:=C1;
sl:=TRUNC(a+b+c+0.1);
s:=a+b+c;
fas:=1;
x1:=6*s*(s+1)*(s-2*a-1)*(s-2*a);
y1:=(s-2*b-1)*(s-2*b)/((2*b-1)
    *2*b*(2*b+1)*(2*b+2)*(2*b+3));
z1:=(s-2*c+1)*(s-2*c+2)/((2*c-3)
    *(2*c-2)*(2*c-1)*2*c*(2*c+1));
  IF Odd(sl) THEN fas:=-1;
  sexj2:=fas*SQRT(x1*y1*z1);
END; (* Function sexj2 *)

(*xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*)
Procedure SetNumber(VAR vprim, vbis:integer;
VAR bok :char;
VAR Jbis, aja :integer;
VAR hkon : htyp);
(* Read initial data from file *)
LABEL 2;
VAR FileName : string[255];
F1 : text;
nil : integer;
BEGIN
  WRITE('File name of input hyperfine constants? ');
  READLN(FileName);
  ASSIGN(F1, FileName);
  RESET(F1);
  aja:=0;
  (* ---------- Read file on form R 98 58 1 ---------- *)
  READ(F1, bok);
  READ(F1, Jbis);
  READLN(F1, vprim, vbis);
  bok:=UpCase(bok);
  IF (bok<>'R') AND (bok<>'P') AND (bok<>'Q') THEN
    BEGIN
      WRITEln('Forbidden line !! ');
      aja:=1;
      GOTO 2;
    END;
  END;
  (* xxxxxx *)
  aja:=aja+1;
  IF (aja<3) THEN
    GOTO 2;
  ELSE
    (* All data is OK *)
    END;
  END;
AI1-b/5
IF bok='Q' THEN
BEGIN
    WRITELN('Forbidden transition !!!');
    aja:=1;
    GOTO 2;
END;

(* ---------- Read 8 hyperfine constants; ---------- *)
(* ------ upper and lower for each constant. --------- *)
FOR nl:=1 TO 8 DO
    READLN(F1, hkon[nl]);
CLOSE(F1);
2:END; (* Procedure SetNumber *)

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

Procedure FindEigVec(jinit, iinit :integer;
    VAR slask :atyp;
    VAR antal: integer);

(* Find appropriate eigen-vectors *)
VAR sum, nl, JC, ml, IC, ftal : integer;
BEGIN
    sum:=0;
    FOR nl:=0 TO 2 DO
        BEGIN
            JC:=jinit+2*nl;
            FOR ml:=0 TO 2 DO
                BEGIN
                    IC:=iinit+2*ml;
                    FOR ftal:=(JC-IC) TO (JC+IC) DO
                        BEGIN
                            sum:=sum+1;
                            WITH slask[sum] DO
                                BEGIN
                                    F:=ftal;
                                    JP:=JC;
                                    IP:=IC;
                                END;
                        END;
                    END;
                END;
            END;
    antal:=sum;
END; (* Procedure FindEigVec *)

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

Procedure RangeEig(VAR slask, egen :atyp;
    antal, jinit, iinit :integer;
    state :char;
    VAR NofLines :integer);
(* Arrange eigen-vectors by increasing $F$, $I$ and $J$ *)

VAR sum, n1, m1, k1 : integer;

BEGIN
  sum:=0;
  k1:=0;
  FOR m1:=(jinit-(iinit+4)) TO (jinit+4+(iinit+4)) DO
    FOR n1:=1 TO antal DO
      IF slask[n1].F=m1 THEN
        BEGIN
          sum:=sum+11
          egen[sum]:=slask[n1];
          IF state='U' THEN
            IF egen[sum].JP=jinit+2 THEN
              k1:=k1+1;
            END;
        END;
    END;
  END;

(* ----- NofLines = number of main transitions ----- *)

NofLines:=k1;
END; (* Procedure RangeEig *)

(*xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*)

Procedure calcEig(egen VAR nostat VAR antal, antmat :integer);

(* Find number of sub-matrices and dimensions *)

LABEL 1;

VAR m1, sum, k1, n1 : integer;

BEGIN
  m1:=egen[l].F;
  sum:=0;
  k1:=1;
  FOR n1:=2 TO antal DO
    WITH egen[n1] DO
      BEGIN
        sum:=sum+1;
        IF F>m1 THEN
          BEGIN
            nostat[k1]:=sum;
            antmat:=k1;
            k1:=k1+1;
            sum:=0;
            m1:=F;
          END;
      END;
  END;
  IF n1=antal THEN
    BEGIN
      nostat[k1]:=sum+1;
      antmat:=k1;
    END;

AII-b/7
GOTO 1;
END;
END;

(* -------- antmat = number of sub-matrices -------- *)
(* ------- nostat = dimension of each matrix ------- *)

1: antmat:=kl;
END; (* Procedure CalcEig *)

(*xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*)

procedure PunktMat(VAR punkt :dtyp;
egen :atyp;
nostat :ctyp;
antmat :integer);

(* Create matrix of eigen-vectors *)
VAR n1, m1, k1, sum : integer;
BEGIN
sum:=0;
FOR n1:=1 TO antmat DO
BEGIN
FOR m1:=1 TO nostat[n1] DO
FOR k1:=1 TO nostat[n1] DO
BEGIN
punkt[n1, m1, k1].F:=egen[sum+1].F;
punkt[n1, m1, k1].JP:=egen[sum+k1].JP;
punkt[n1, m1, k1].IP:=egen[sum+k1].IP;
punkt[n1, m1, k1].JB:=egen[sum+ml].JP;
punkt[n1, m1, k1].IB:=egen[sum+ml].IP;
END;
sum:=sum+nostat[n1];
END;
END; (* Procedure PunktMat *)

(*xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx*)

Procedure MatElement(antmat :integer;
nostat :ctyp;
punkt :dtyp;
VAR M :mattyp7;
till, fran :real;
state :char;
EQQ, CKON, AKON, DKON :real);

(* Calculate raw matrix elements including constants *)
(* Calculate the hyperfine correction matrix *)
VAR n1, m1, k1, ms, ns : integer;
C, A, D, p1, p2, p3, Delta : real;
BEGIN

(* Calculate half the matrix and symmetrize later. *)
FOR n1:=1 TO antmat DO
BEGIN

ns:=l;
ms:=l;
FOR ml:=ms TO nostat[nl] DO
BEGIN
FOR kl:=ns TO nostat[nl] DO
WITH punkt[nl, ml, kl] DO
BEGIN
C:=0.0; A:=0.0; D:=0.0;
p1:=0.0; p2:=0.0; p3:=0.0;
Delta:=0.0;

IF JP-JB=0 THEN
BEGIN
p2:=0.0;
p3:=0.0;
IF IP-IB=0 THEN
BEGIN
C:=Cfunk(F, JP, IP);
A:=Afunk(IP);
p1:=di0dj0(F, JB, IB);
p2:=sexj0(F, JB, IB);
p3:=NioJ0(IP);
(* ----- Delta = rotational energy difference ------- *)
END
ELSE IF JP=jinit+2 THEN Delta:=0
ELSE IF JP=jinit+4 THEN Delta:=till
ELSE IF JP=jinit THEN Delta:=-fran;
Delta:=Delta*convert;
END
ELSE IF IP-IB=-2 THEN
BEGIN
p1:=dim2dj0(F, JB, IB);
p2:=sexj2(F, JB, IB);
p3:=NioJ2(IB);
END
ELSE IF IP-IB=2 THEN
BEGIN
p1:=dim2dj0(F, JP, IP);
p2:=sexj2(F, JP, IP);
p3:=NioJ2(IP);
END
ELSE IF ABS(IP-IB)>2.05 THEN
p1:=0;
D:=Dfunk(F, JP, IP, IB)*p2*p3;
END
ELSE IF JP-JB=2 THEN
BEGIN
IF IP-IB=0 THEN
pl:=di0dij2(F, JB, IB)
ELSE IF IP-IB=2 THEN
pl:=di2dij2(F, JB, IB)
ELSE IF IP-IB=-2 THEN
pl:=dim2dij2(F, JB, IB)
ELSE IF ABS(IP-IB)>2.05 THEN
pl:=0;
END
ELSE pl:=0;
(* pl set to 0 because matrix is symmetrized later. *)

M[nl, ml, kl]:=pl*EQQ+C*CKON+Delta
+A*AKON+D*DKON;

(* ---------------- Symmetrize ! ---------------- *)
M[nl, kl, ml]:=M[nl, ml, kl];
END;
ms:=ms+l;
ns:=ns+l;
END;
END; /* Procedure MatElement */

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX*)

procedure differ(eget, egetu, egetl, nostat, jinit, antmat, NofLines, vprim, vbis, Jbis,
bok, hkon, : atyp; : btyp; : ctyp; : integer; : char; : atyp; : btyp;)

VAR FileName : string[255];
UtFil : text;
nl, xl, yl, ml,
Fbis : integer;
diff : real;
p, b : char;
a, etikett : atyp;
linje, lu, ll,
DD, crossu, crossl : btyp;

(* lu - upper level *)
(* ll - lower level *)
(* DD line with delta-F=0 *)
(* crossu - cross-over with common upper level *)
(* crossl - cross-over with common lower level *)

BEGIN
WRITE('File name output frequency differences ? '); READLN(FileName); ASSIGN(UtFil, FileName); REWRITE(UtFil);

(* ----------------- Reset to 0. ----------------- *)
y1:=0;
x1:=0;
FOR n1:=1 TO NofLines DO BEGIN
  lu[n1]:=0;
  ll[n1]:=0;
  DD[n1]:=0;
  crossu[n1]:=0;
  crossl[n1]:=0;
END;

(* ---------- Set prime and double prime. ---------- *)
p:=chr(39);
b:=chr(34);

(* ---------- Find upper and lower levels. ---------- *)
FOR n1:=1 TO antmat DO BEGIN
  FOR m1:=1 TO nostat[n1] DO BEGIN
    x1:=x1+1;
    IF egen[x1].JP=jinit+2 THEN BEGIN
      y1:=y1+1;
      etikett[y1]:=egen[x1];
      lu[y1]:=egetu[x1];
      ll[y1]:=egetl[x1];
    END;
  END;
END;

(* --- Find Delta-F = Delta-J and Delta-0 lines. --- *)
FOR n1:=1 TO NofLines DO
  linje[n1]:=lu[n1]-ll[n1];
  IF bok='R' THEN
    FOR n1:=1 TO (NofLines-1) DO FOR m1:=0 TO 2 DO
      IF (m1+n1)<NofLines THEN
        IF etikett[n1].F = (etikett[n1+1+m1].F-1) THEN
          IF etikett[n1].IP = etikett[n1+1+m1].IP THEN
            DD[n1]:=lu[n1]-ll[n1+1+m1];
      END;
    END;
  END;

  IF bok='P' THEN
    FOR n1:=2 TO NofLines DO FOR m1:=0 TO 2 DO
      IF (n1-m1)>1 THEN
        IF etikett[n1].F = (etikett[n1-1-m1].F+1) THEN
          IF etikett[n1].IP = etikett[n1-1-m1].IP THEN
AII-b/11
DD[n1] := lu[n1] - ll[n1-1-m1];

(* -------------- Find cross-over lines. -------------- *)

FOR n1 := 1 TO NofLines DO
BEGIN
  IF DD[n1] = 0 THEN GOTO 69 ELSE
dcrossu[n1] := (linje[n1] + DD[n1]) / 2;
69: END;

IF bok = 'R' THEN
FOR n1 := 2 TO NofLines DO
FOR m1 := 0 TO 2 DO
  IF (n1 - m1) > 1 THEN
    IF etikett[n1].F = (etikett[n1-1-m1].F + 1) THEN
      IF etikett[n1].IP = etikett[n1-1-m1].IP THEN
BEGIN
  IF DD[n1-1-m1] = 0 THEN GOTO 33 ELSE
dcrossl[n1] := (linje[n1] + DD[n1-1-m1]) / 2;
33: END;

IF bok = 'P' THEN
FOR n1 := 1 TO NofLines - 1 DO
FOR m1 := 0 TO 2 DO
  IF (n1 + m1) < NofLines THEN
    IF etikett[n1].F = (etikett[n1+1+m1].F - 1) THEN
      IF etikett[n1].IP = etikett[n1+1+m1].IP THEN
BEGIN
    IF DD[n1+1+m1] = 0 THEN GOTO 44 ELSE
dcrossl[n1] := (linje[n1] + DD[n1+1+m1]) / 2;
44: END;

(* Arrange lines in increasing frequency order. *)

REPEAT
  diff := linje[1];
  FOR n1 := 1 TO NofLines - 1 DO
    IF linje[n1+1] < linje[n1] THEN
      BEGIN
      diff := linje[n1+1];
      linje[n1+1] := linje[n1];
      linje[n1] := diff;

      diff := DD[n1+1];
      DD[n1+1] := DD[n1];
      DD[n1] := diff;

      diff := crossu[n1+1];
      crossu[n1+1] := crossu[n1];
      crossu[n1] := diff;

      diff := crossl[n1+1];
      crossl[n1+1] := crossl[n1];
      crossl[n1] := diff;

      a[n1] := etikett[n1+1];
      etikett[n1+1] := etikett[n1];
      etikett[n1] := a[n1];

      END;
  UNTIL diff = linje[1];
(* --- Print data on file. --- *)

WRITE(UtFil,' TRANSITION IN IODINE : ');
WRITELN(UtFil,bok:2,' (',Jbis:3,',')',' vprim:3,'-',vbis:2);
WRITELN(UtFil);
WRITELN(UtFil,' Constants: eqQ',p:1,' eqQ',b:1,' C',p:1,' C',b:1,
' A',p:1,' A',b:1,' D',p:1,' D',b:1);
WRITELN(UtFil);
FOR n1:=1 TO 8 DO
WRITELN(UtFil, n1:5, hkon[n1]:13:6);
WRITELN(UtFil);
WRITELN(UtFil,' no',',',F',',p:1,' F',',b:1,' I ',
' D-F=D-J ',',F',',p:1,'=F',',b:1,
' D-F=0 ',', F',',p:1,' cros up ',
' F',',b:1,' cros lo ');
WRITELN(UtFil);
FOR n1:=1 TO NofLines DO
WITH etikett[n1] DO
BEGIN
IF bok='R' THEN Fbis:=F-1 ELSE Fbis:=F+1,
WRITELN(UtFil, n1:3, F:4, Fbis:4, IP:3,
 linje[n1]:10:3,
 F:5, DD[n1]:10:3,
 F:5, crosu[n1]:10:3,
 Fbis:5, cros1[n1]:10:3);
END;
WRITELN(UtFil);
CLOSE(UtFil);
END; (* Procedure differ *)

(*XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX)*
Procedure Jacobi(Dimen : integer;
   Mat : TNmatrix;
   MaxIter : integer;
   Tolerance : real;
   VAR Eigenvalues : TNvector;
   VAR Eigenvectors : TNmatrix;
   VAR Iter : integer;
   VAR Error : byte);

CONST   TNNearlyZero = 1E-15;

VAR   Row, Column, Diag : integer;
   SinTheta, CosTheta : real;
   SumSquareDiag : real;
   Done : boolean;

Procedure TestData(Dimen : integer;
   VAR Mat : TNmatrix;
   VAR MaxIter : integer;
   Tolerance : real;
   VAR Error : byte);

VAR   Row, Column : integer;

BEGIN
   Error:=0;
   IF Dimen<1 THEN Error:=1;
   IF Tolerance<=TNNearlyZero THEN Error:=2;
   IF MaxIter<1 THEN Error:=3;
   IF Error=0 THEN
      FOR Row:=1 TO Dimen-1 DO
         FOR Column:=Row+1 TO Dimen DO
            IF ABS(Mat[Row, Column]-Mat[Column, Row])>
               TNNearlyZero THEN Error:=4;
      END; (* Procedure TestData *)

Procedure Initialize(Dimen : integer;
   VAR Iter : integer;
   VAR Eigenvectors : TNmatrix);

VAR   Diag : integer;

BEGIN
   Iter:=0;
   FillChar(Eigenvectors, SizeOf(Eigenvectors), 0);
   FOR Diag:=1 TO Dimen DO
      Eigenvectors[Diag, Diag]:=1;
   END; (* Procedure Initialize *)

Procedure CalculateRotation(RowRow : real;
VAR   RowCol   : real;
    ColCol   : real;
VAR   SinTheta   : real;
    CosTheta   : real;

VAR   TangentTwoTheta, TangentTheta, Dummy : real;

BEGIN   IF ABS(RowRow-ColCol)>TNNearlyZero THEN 
BEGIN
    TangentTwoTheta:=(RowRow-ColCol)/(2*RowCol);
    Dummy:=SQRT(SQR(TangentTwoTheta)+1);
    IF TangentTwoTheta<0 THEN
        TangentTheta:=-TangentTwoTheta-Dummy
    ELSE
        TangentTheta:=-TangentTwoTheta+Dummy;
    CosTheta:=1/SQRT(1+SQR(TangentTheta));
    SinTheta:=CosTheta*TangentTheta;
END
ELSE
BEGIN
    CosTheta:=SQRT(1/2);
    IF RowCol<0 THEN SinTheta:=-SQRT(1/2)
    ELSE
        SinTheta:=SQRT(1/2);
END;
END; (* Procedure CalculateRotation *)

(* XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX *)

Procedure RotateMatrix(Dimen   : integer;
                          SinTheta   : real;
                          CosTheta   : real;
                          Row   : integer;
                          Col   : integer;
                          VAR Mat   : TNmatrix);

VAR   CosSqr, SinSqr, SinCos   : real;
    MatRowRow, MatColCol, MatRowCol,
    MatRowIndex, MatColIndex   : real;
    Index   : integer;

BEGIN
    CosSqr:=Sqr(CosTheta);
    SinSqr:=Sqr(SinTheta);
    SinCos:=SinTheta*CosTheta;
    MatRowRow:=Mat[Row, Row]*CosSqr+2*Mat[Row, Col]*SinCos
        +Mat[Col, Col]*SinSqr;
    MatColCol:=Mat[Row, Row]*SinSqr-2*Mat[Row, Col]*SinCos
        +Mat[Col, Col]*CosSqr;
    MatRowCol:=(Mat[Col, Col]-Mat[Row, Row])*SinCos
        +Mat[Row, Col]*(CosSqr-SinSqr);
    FOR Index:=1 TO Dimen DO
    IF NOT(Index in [Row, Col]) THEN
BEGIN
    MatRowIndex:=Mat[Row, Index]*CosTheta
            +Mat[Col, Index]*SinTheta;
AII-d/2
MatColIndex := -Mat[Row, Index]*SinTheta + Mat[Col, Index]*CosTheta;
Mat[Row, Index] := MatRowIndex;
Mat[Col, Index] := MatColIndex;
Mat[Row, Row] := MatRowRow;
Mat[Col, Col] := MatColCol;
Mat[Row, Col] := MatRowCol;
Mat[Col, Row] := MatRowCol;

END; (* Procedure RotateMatrix *)

(* XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX *)

Procedure RotateEigenvectors(Oimen, SlnTheta, CosTheta, Row, Col, VAR Eigenvectors)
integer; : real; : real; : integer; : integer;
VAR EigenvectorsRowIndex, EigenvectorsColIndex : real;
Index : integer;
BEGIN
(* Transform eigenvector matrix *)
FOR Index := 1 TO Dimen DO
BEGIN
EigenvectorsRowIndex := CosTheta*Eigenvectors[Row, Index] + SinTheta*Eigenvectors[Col, Index];
EigenvectorsColIndex := -SinTheta*Eigenvectors[Row, Index] + CosTheta*Eigenvectors[Col, Index];
Eigenvectors[Row, Index] := EigenvectorsRowIndex;
Eigenvectors[Col, Index] := EigenvectorsColIndex;
END;
END; (* Procedure RotateEigenvectors *)

(* XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX *)

Procedure NormalizeEigenvectors(Dimen, VAR Eigenvectors)
integer; : integer;
VAR Row, Largest : integer;
VAR Eigenvector, Largest : real;
BEGIN
VAR Term : integer;

(* ------------------------------- *)

Procedeu FindLargest(Dimen, VAR Eigenvector, VAR Largest)
integer; : integer;
VAR Eigenvector, Largest : real;
BEGIN
VAR Term : integer;

Largest := Eigenvector[1];
FOR Term := 2 TO Dimen DO
IF ABS(Eigenvector[Term]) > ABS(Largest) THEN
Largest:=Eigenvector[Term];
END; (* Procedure FindLargest *)

(* -------------------------------------------------- *)

Procedure DivVecConst(Dimen : integer;
                        TNvector;
                        real);
VAR ChangingRow : TNvector;
Divisor : real);

VAR Term : integer;

BEGIN
FOR Term:=1 TO Dimen DO
  ChangingRow[Term]:=ChangingRow[Term]/Divisor;
END; (* Procedure DivVecConst *)

(* -------------------------------------------------- *)

BEGIN (* Procedure NormalizeEigenvectors *)
FOR Row:=1 TO Dimen DO
BEGIN
  FindLargest(Dimen, Eigenvectors[Row], Largest);
  DivVecConst(Dimen, Eigenvectors[Row], Largest);
END;
END; (* Procedure NormalizeEigenvectors *)

(* -------------------------------------------------- *)

BEGIN (* Procedure Jacobi *)
TestData(Dimen, Mat, MaxIter, Tolerance, Error);
IF Error=0 THEN
BEGIN
  Initialize(Dimen, Iter, Eigenvectors);
  REPEAT
    Iter:=Succ(Iter);
    SumSquareDiag:=0;
    FOR Diag:=1 TO Dimen DO
      SumSquareDiag:=SumSquareDiag+SQR(Mat[Diag, Diag]);
    Done:=TRUE;
    FOR Row:=1 TO Dimen-1 DO
      FOR Column:=Row+1 TO Dimen DO
        IF ABS(Mat[Row, Column])>Tolerance*SumSquareDiag THEN
          Done:=false;
          CalculateRotation(Mat[Row, Row], Mat[Row, Column],
                             Mat[Column, Column],
                             SinTheta, CosTheta);
          RotateMatrix(Dimen, SinTheta, CosTheta,
                        Row, Column, Mat);
          RotateEigenvectors(Dimen, SinTheta, CosTheta,
                             Row, Column, Eigenvectors);
        END;
      END;
    END;
  UNTIL Done OR (Iter>MaxIter);
FOR Diag:=1 TO Dimen DO
  Eigenvalues[Diag]:=Mat[Diag, Diag];
END;
END; (* Procedure Jacobi *)
Procedure diag(VAR eget : btyp;
Dimen : integer;
VAR kl : integer;
Mat : TNmatrix);

VAR MaxIter, Iter : integer;
Tolerance : real;
Eigenvectors : TNmatrix;
Eigenvalues : TNvector;
Error : byte;
m1 : integer;

BEGIN
Tolerance := 1E-8;
MaxIter := 200;

Jacobi(Dimen, Mat, MaxIter, Tolerance,
   Eigenvalues, Eigenvectors, Iter, Error);

FOR m1 := 1 TO Dimen DO
BEGIN
   kl := kl + 1;
   eget[kl] := 10000*Eigenvalues[m1];
END;
END; (* Procedure Diag *)
<table>
<thead>
<tr>
<th>R</th>
<th>98.58</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>-493.839</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-2374.037</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.8292</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.3568</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-0.5004</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.0015</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

File name of input hyperfine constants: BXR98.ABC

File name output frequency differences: UT.DAT

A:"/h1
TRANSITION IN IODINE:  R (98) 58-1

Constants: \( \epsilon q' \) \( \epsilon q'' \) \( C' \) \( C'' \) \( A' \) \( A'' \) \( D' \) \( D'' \)

|   | \( F' \) | \( F'' \) | \( I \) | \( D-F=D-J \) | \( F'=F'' \) | \( D-F=0 \) | \( F' \) | \( cros up \) | \( F'' \) | \( cros lo \) |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 95 | 94 | 4 | -525.145 | 95 | -351.471 | 95 | -438.308 | 94 | 0.000 |
| 3 | 96 | 95 | 4 | -306.162 | 96 | -298.737 | 96 | -302.449 | 95 | -328.816 |
| 4 | 98 | 97 | 2 | -270.771 | 98 | -630.599 | 98 | -450.685 | 97 | -367.808 |
| 5 | 97 | 96 | 4 | -217.391 | 97 | 134.968 | 97 | -41.211 | 96 | -258.064 |
| 6 | 100 | 99 | 2 | -111.641 | 100 | 431.291 | 100 | 159.825 | 99 | -115.260 |
| 7 | 97 | 96 | 2 | 60.596 | 97 | -464.846 | 97 | -202.125 | 96 | 0.000 |
| 8 | 99 | 98 | 4 | 88.847 | 99 | 271.554 | 99 | 180.200 | 98 | 29.015 |
| 9 | 101 | 100 | 4 | 114.335 | 101 | 121.943 | 101 | 118.139 | 99 | 34.757 |
| 10 | 103 | 102 | 4 | 141.590 | 103 | 0.000 | 103 | 0.000 | 102 | 80.880 |
| 11 | 98 | 97 | 4 | 142.491 | 98 | -30.818 | 98 | 55.837 | 97 | 138.729 |
| 12 | 102 | 101 | 4 | 202.523 | 102 | 20.171 | 102 | 111.347 | 101 | 162.233 |
| 13 | 100 | 99 | 4 | 315.047 | 100 | -44.822 | 100 | 135.113 | 99 | 293.300 |
| 14 | 99 | 98 | 0 | 369.941 | 99 | 0.000 | 99 | 0.000 | 98 | 0.000 |
| 15 | 101 | 100 | 2 | 398.612 | 101 | 0.000 | 101 | 0.000 | 100 | 414.951 |

AIII-c
TRANSITION IN IODINE: R (98) 58–1

Constants: \( e \equiv Q' \), \( e \equiv Q'' \), \( C' \), \( C'' \), \( A' \), \( A'' \), \( D' \), \( D'' \)

<table>
<thead>
<tr>
<th>no</th>
<th>( F' )</th>
<th>( F'' )</th>
<th>I</th>
<th>( D-F=D-J )</th>
<th>( F'=F'' )</th>
<th>( D-F=0 )</th>
<th>( F' ) cros up</th>
<th>( F'' ) cros lo</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>95</td>
<td>94</td>
<td>4</td>
<td>-525.145</td>
<td>95</td>
<td>-351.471</td>
<td>95</td>
<td>438.308</td>
</tr>
<tr>
<td>3</td>
<td>96</td>
<td>95</td>
<td>4</td>
<td>-306.162</td>
<td>96</td>
<td>-298.737</td>
<td>96</td>
<td>-302.449</td>
</tr>
<tr>
<td>4</td>
<td>98</td>
<td>97</td>
<td>2</td>
<td>-270.777</td>
<td>98</td>
<td>-630.599</td>
<td>98</td>
<td>-450.685</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
<td>96</td>
<td>4</td>
<td>-217.391</td>
<td>97</td>
<td>134.968</td>
<td>97</td>
<td>-41.211</td>
</tr>
<tr>
<td>6</td>
<td>100</td>
<td>99</td>
<td>2</td>
<td>-131.641</td>
<td>100</td>
<td>431.290</td>
<td>100</td>
<td>159.825</td>
</tr>
<tr>
<td>7</td>
<td>97</td>
<td>96</td>
<td>2</td>
<td>60.596</td>
<td>97</td>
<td>-464.846</td>
<td>97</td>
<td>-202.125</td>
</tr>
<tr>
<td>8</td>
<td>99</td>
<td>98</td>
<td>4</td>
<td>88.847</td>
<td>99</td>
<td>271.554</td>
<td>99</td>
<td>180.200</td>
</tr>
<tr>
<td>9</td>
<td>101</td>
<td>100</td>
<td>4</td>
<td>114.335</td>
<td>101</td>
<td>121.944</td>
<td>101</td>
<td>118.139</td>
</tr>
<tr>
<td>10</td>
<td>103</td>
<td>102</td>
<td>4</td>
<td>141.590</td>
<td>103</td>
<td>0.000</td>
<td>103</td>
<td>0.000</td>
</tr>
<tr>
<td>11</td>
<td>98</td>
<td>97</td>
<td>4</td>
<td>142.491</td>
<td>98</td>
<td>-30.818</td>
<td>98</td>
<td>55.836</td>
</tr>
<tr>
<td>12</td>
<td>102</td>
<td>101</td>
<td>4</td>
<td>202.523</td>
<td>102</td>
<td>25.171</td>
<td>102</td>
<td>111.347</td>
</tr>
<tr>
<td>13</td>
<td>100</td>
<td>99</td>
<td>4</td>
<td>315.047</td>
<td>100</td>
<td>-44.621</td>
<td>100</td>
<td>135.113</td>
</tr>
<tr>
<td>14</td>
<td>99</td>
<td>98</td>
<td>0</td>
<td>369.942</td>
<td>99</td>
<td>0.000</td>
<td>99</td>
<td>0.000</td>
</tr>
<tr>
<td>15</td>
<td>101</td>
<td>100</td>
<td>2</td>
<td>398.613</td>
<td>101</td>
<td>0.000</td>
<td>101</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Comparison between BIPM program and calculation by Foth and Spieweck [36].

<table>
<thead>
<tr>
<th>no</th>
<th>( F' )</th>
<th>( F'' )</th>
<th>I</th>
<th>( f )</th>
<th>( f + 111.631 )</th>
<th>( f [36] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>95</td>
<td>94</td>
<td>4</td>
<td>-525.145</td>
<td>-413.514</td>
<td>-413.513</td>
</tr>
<tr>
<td>2</td>
<td>99</td>
<td>98</td>
<td>2</td>
<td>-471.189</td>
<td>-359.585</td>
<td>-359.558</td>
</tr>
<tr>
<td>3</td>
<td>96</td>
<td>95</td>
<td>4</td>
<td>-306.162</td>
<td>-194.531</td>
<td>-194.530</td>
</tr>
<tr>
<td>4</td>
<td>98</td>
<td>97</td>
<td>2</td>
<td>-270.777</td>
<td>-159.140</td>
<td>-159.139</td>
</tr>
<tr>
<td>5</td>
<td>97</td>
<td>96</td>
<td>4</td>
<td>-217.391</td>
<td>-105.760</td>
<td>-105.759</td>
</tr>
<tr>
<td>6</td>
<td>100</td>
<td>99</td>
<td>2</td>
<td>-111.641</td>
<td>-0.010</td>
<td>-0.010</td>
</tr>
<tr>
<td>7</td>
<td>97</td>
<td>96</td>
<td>2</td>
<td>60.596</td>
<td>172.227</td>
<td>172.228</td>
</tr>
<tr>
<td>8</td>
<td>99</td>
<td>98</td>
<td>4</td>
<td>88.847</td>
<td>200.478</td>
<td>200.478</td>
</tr>
<tr>
<td>9</td>
<td>101</td>
<td>100</td>
<td>4</td>
<td>114.335</td>
<td>225.966</td>
<td>225.965</td>
</tr>
<tr>
<td>10</td>
<td>103</td>
<td>102</td>
<td>4</td>
<td>141.590</td>
<td>253.221</td>
<td>253.220</td>
</tr>
<tr>
<td>11</td>
<td>98</td>
<td>97</td>
<td>4</td>
<td>142.491</td>
<td>254.122</td>
<td>254.122</td>
</tr>
<tr>
<td>12</td>
<td>102</td>
<td>101</td>
<td>4</td>
<td>202.523</td>
<td>314.154</td>
<td>314.152</td>
</tr>
<tr>
<td>13</td>
<td>100</td>
<td>99</td>
<td>4</td>
<td>315.047</td>
<td>426.678</td>
<td>426.677</td>
</tr>
<tr>
<td>14</td>
<td>99</td>
<td>98</td>
<td>0</td>
<td>369.942</td>
<td>481.573</td>
<td>481.573</td>
</tr>
<tr>
<td>15</td>
<td>101</td>
<td>100</td>
<td>2</td>
<td>398.613</td>
<td>510.244</td>
<td>510.243</td>
</tr>
</tbody>
</table>
TRANSITION IN IODINE: P (13) 43-0

Constants: eqQ', eqQ" C' C" A' A" D' D"

<table>
<thead>
<tr>
<th>no</th>
<th>F'</th>
<th>F&quot;</th>
<th>I</th>
<th>D-F=D-J</th>
<th>F'=F&quot;</th>
<th>D-F=0</th>
<th>F'</th>
<th>cros up</th>
<th>F&quot;</th>
<th>cros lo</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>8</td>
<td>5</td>
<td>-543.858</td>
<td>7</td>
<td>0.000</td>
<td>7</td>
<td>0.000</td>
<td>8</td>
<td>-572.074</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>13</td>
<td>1</td>
<td>-471.995</td>
<td>12</td>
<td>570.744</td>
<td>12</td>
<td>49.375</td>
<td>13</td>
<td>-577.552</td>
</tr>
<tr>
<td>3</td>
<td>17</td>
<td>18</td>
<td>5</td>
<td>-412.087</td>
<td>17</td>
<td>46.070</td>
<td>17</td>
<td>-181.009</td>
<td>18</td>
<td>0.000</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>9</td>
<td>5</td>
<td>-336.038</td>
<td>8</td>
<td>-600.289</td>
<td>8</td>
<td>-468.164</td>
<td>9</td>
<td>-360.888</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>14</td>
<td>3</td>
<td>-208.859</td>
<td>13</td>
<td>179.136</td>
<td>13</td>
<td>-14.862</td>
<td>14</td>
<td>-232.223</td>
</tr>
<tr>
<td>6</td>
<td>11</td>
<td>12</td>
<td>3</td>
<td>-171.314</td>
<td>11</td>
<td>-42.155</td>
<td>11</td>
<td>-106.734</td>
<td>12</td>
<td>-207.853</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>10</td>
<td>3</td>
<td>-257.081</td>
<td>9</td>
<td>-385.738</td>
<td>9</td>
<td>-271.410</td>
<td>10</td>
<td>-194.835</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>11</td>
<td>3</td>
<td>-73.389</td>
<td>10</td>
<td>55.871</td>
<td>10</td>
<td>-8.759</td>
<td>11</td>
<td>-57.772</td>
</tr>
<tr>
<td>9</td>
<td>16</td>
<td>17</td>
<td>5</td>
<td>-62.373</td>
<td>16</td>
<td>286.512</td>
<td>16</td>
<td>112.070</td>
<td>17</td>
<td>-8.152</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
<td>15</td>
<td>3</td>
<td>-43.113</td>
<td>14</td>
<td>-255.586</td>
<td>14</td>
<td>-149.350</td>
<td>15</td>
<td>-45.260</td>
</tr>
<tr>
<td>11</td>
<td>15</td>
<td>16</td>
<td>3</td>
<td>-18.124</td>
<td>15</td>
<td>-47.408</td>
<td>15</td>
<td>-32.766</td>
<td>16</td>
<td>0.000</td>
</tr>
<tr>
<td>12</td>
<td>9</td>
<td>10</td>
<td>3</td>
<td>23.510</td>
<td>9</td>
<td>0.000</td>
<td>9</td>
<td>0.000</td>
<td>10</td>
<td>39.690</td>
</tr>
<tr>
<td>13</td>
<td>12</td>
<td>13</td>
<td>3</td>
<td>87.623</td>
<td>12</td>
<td>-244.391</td>
<td>12</td>
<td>-78.384</td>
<td>13</td>
<td>133.380</td>
</tr>
<tr>
<td>14</td>
<td>10</td>
<td>11</td>
<td>5</td>
<td>106.091</td>
<td>10</td>
<td>-232.589</td>
<td>10</td>
<td>-63.249</td>
<td>11</td>
<td>97.084</td>
</tr>
<tr>
<td>15</td>
<td>11</td>
<td>12</td>
<td>5</td>
<td>175.309</td>
<td>11</td>
<td>88.076</td>
<td>11</td>
<td>131.693</td>
<td>12</td>
<td>151.084</td>
</tr>
<tr>
<td>16</td>
<td>15</td>
<td>16</td>
<td>5</td>
<td>204.665</td>
<td>15</td>
<td>367.544</td>
<td>15</td>
<td>286.108</td>
<td>16</td>
<td>245.589</td>
</tr>
<tr>
<td>17</td>
<td>13</td>
<td>14</td>
<td>1</td>
<td>248.844</td>
<td>13</td>
<td>-683.110</td>
<td>13</td>
<td>-217.133</td>
<td>14</td>
<td>0.000</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
<td>15</td>
<td>3</td>
<td>328.272</td>
<td>14</td>
<td>429.164</td>
<td>14</td>
<td>378.718</td>
<td>15</td>
<td>347.911</td>
</tr>
<tr>
<td>19</td>
<td>11</td>
<td>12</td>
<td>1</td>
<td>330.056</td>
<td>11</td>
<td>0.000</td>
<td>11</td>
<td>0.000</td>
<td>12</td>
<td>450.400</td>
</tr>
<tr>
<td>20</td>
<td>12</td>
<td>13</td>
<td>5</td>
<td>345.544</td>
<td>12</td>
<td>126.859</td>
<td>12</td>
<td>236.202</td>
<td>13</td>
<td>337.720</td>
</tr>
<tr>
<td>21</td>
<td>13</td>
<td>14</td>
<td>5</td>
<td>405.248</td>
<td>13</td>
<td>329.896</td>
<td>13</td>
<td>367.572</td>
<td>14</td>
<td>417.206</td>
</tr>
</tbody>
</table>

Comparison between BIPM program and calculation by Bordé at al. [2].

<table>
<thead>
<tr>
<th>no</th>
<th>F'</th>
<th>F&quot;</th>
<th>I</th>
<th>f</th>
<th>f+543.858</th>
<th>f[2]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>8</td>
<td>5</td>
<td>-543.858</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>13</td>
<td>1</td>
<td>-471.995</td>
<td>71.863</td>
<td>71.864</td>
</tr>
<tr>
<td>3</td>
<td>17</td>
<td>18</td>
<td>5</td>
<td>-412.087</td>
<td>131.771</td>
<td>131.772</td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>9</td>
<td>5</td>
<td>-336.038</td>
<td>207.820</td>
<td>207.820</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>14</td>
<td>3</td>
<td>-208.859</td>
<td>335.000</td>
<td>335.000</td>
</tr>
<tr>
<td>6</td>
<td>11</td>
<td>12</td>
<td>3</td>
<td>-171.314</td>
<td>372.544</td>
<td>372.544</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>10</td>
<td>5</td>
<td>-157.081</td>
<td>386.777</td>
<td>386.777</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>11</td>
<td>3</td>
<td>-73.389</td>
<td>470.469</td>
<td>470.470</td>
</tr>
<tr>
<td>9</td>
<td>16</td>
<td>17</td>
<td>5</td>
<td>-62.373</td>
<td>481.485</td>
<td>481.485</td>
</tr>
<tr>
<td>10</td>
<td>14</td>
<td>15</td>
<td>3</td>
<td>-43.113</td>
<td>500.745</td>
<td>-</td>
</tr>
<tr>
<td>11</td>
<td>15</td>
<td>16</td>
<td>3</td>
<td>-18.124</td>
<td>525.734</td>
<td>525.735</td>
</tr>
<tr>
<td>12</td>
<td>9</td>
<td>10</td>
<td>3</td>
<td>23.510</td>
<td>567.368</td>
<td>567.368</td>
</tr>
<tr>
<td>13</td>
<td>12</td>
<td>13</td>
<td>3</td>
<td>87.623</td>
<td>631.481</td>
<td>631.482</td>
</tr>
<tr>
<td>14</td>
<td>10</td>
<td>11</td>
<td>5</td>
<td>106.091</td>
<td>649.949</td>
<td>-</td>
</tr>
<tr>
<td>15</td>
<td>11</td>
<td>12</td>
<td>5</td>
<td>175.309</td>
<td>719.167</td>
<td>719.167</td>
</tr>
<tr>
<td>16</td>
<td>15</td>
<td>16</td>
<td>5</td>
<td>204.665</td>
<td>748.523</td>
<td>748.524</td>
</tr>
<tr>
<td>17</td>
<td>13</td>
<td>14</td>
<td>1</td>
<td>248.844</td>
<td>792.702</td>
<td>792.703</td>
</tr>
<tr>
<td>18</td>
<td>14</td>
<td>15</td>
<td>5</td>
<td>328.272</td>
<td>872.130</td>
<td>-</td>
</tr>
<tr>
<td>19</td>
<td>11</td>
<td>12</td>
<td>1</td>
<td>330.056</td>
<td>873.914</td>
<td>-</td>
</tr>
<tr>
<td>20</td>
<td>12</td>
<td>13</td>
<td>5</td>
<td>345.544</td>
<td>889.402</td>
<td>889.403</td>
</tr>
<tr>
<td>21</td>
<td>13</td>
<td>14</td>
<td>5</td>
<td>405.248</td>
<td>949.106</td>
<td>949.106</td>
</tr>
</tbody>
</table>

AIV-b/1
The frequency spacing between a main component 'a' and a cross-over line 'c' has been calculated by Bordé et al. and correspond to the underlined data.

**The numerotation of the cross-over lines in [2] is not the same one as recommended in [49].**

<table>
<thead>
<tr>
<th>no</th>
<th>F'</th>
<th>F''</th>
<th>I</th>
<th>D-F=D-J</th>
<th>cros up</th>
<th>cros lo</th>
<th>a</th>
<th>c-a*</th>
<th>c-a[2]**</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>8</td>
<td>5</td>
<td>543.858</td>
<td>0.000</td>
<td>-572.674</td>
<td>1</td>
<td>-28.216</td>
<td>-28.215</td>
</tr>
<tr>
<td>2</td>
<td>12</td>
<td>13</td>
<td>1</td>
<td>-471.995</td>
<td>49.375</td>
<td>-577.552</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>17</td>
<td>18</td>
<td>5</td>
<td>-412.087</td>
<td>-183.009</td>
<td>0.000</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>8</td>
<td>9</td>
<td>5</td>
<td>-336.038</td>
<td>-468.164</td>
<td>-360.888</td>
<td>1</td>
<td>75.694</td>
<td>75.693</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>-24.850</td>
<td>-24.850</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>7</td>
<td>-203.807</td>
<td>-203.807</td>
</tr>
</tbody>
</table>

5  | 13 | 14  | 3 | -208.859 | -14.862 | -232.223 |   |       |         |
6  | 11 | 12  | 3 | -171.314 | -106.734 | -207.853 |   |       |         |
7  | 9  | 10  | 5 | -157.081 | -271.410 | -194.835 | 4 | 64.628 | 64.629   |
|    |    |     |   |         |         |         | 7 | -114.329 | -114.328 |

AIV-b/2
Comparison between BIPM program and calculation by Razet (53).

<table>
<thead>
<tr>
<th>no</th>
<th>F' F'' I</th>
<th>D-F=D-J F'=F''</th>
<th>D-F=0 F'</th>
<th>cros up</th>
<th>F'' cros lo</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>123 122 5</td>
<td>-510.782 123</td>
<td>-145.689 123</td>
<td>-328.235 122</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>128 127 1</td>
<td>-485.996 128</td>
<td>646.756 128</td>
<td>80.380 127</td>
<td>-613.200</td>
</tr>
<tr>
<td>3</td>
<td>133 132 5</td>
<td>-497.055 133</td>
<td>607.840 133</td>
<td>0.000 132</td>
<td>-507.166</td>
</tr>
<tr>
<td>4</td>
<td>124 123 5</td>
<td>-223.686 124</td>
<td>156.734 124</td>
<td>-33.476 123</td>
<td>-184.687</td>
</tr>
<tr>
<td>5</td>
<td>127 126 3</td>
<td>-195.703 127</td>
<td>176.750 127</td>
<td>-9.477 126</td>
<td>-218.139</td>
</tr>
<tr>
<td>6</td>
<td>129 128 3</td>
<td>-193.219 129</td>
<td>-1.685 129</td>
<td>-97.452 128</td>
<td>-237.375</td>
</tr>
<tr>
<td>7</td>
<td>132 131 5</td>
<td>-166.138 132</td>
<td>-552.107 132</td>
<td>-359.123 131</td>
<td>-209.450</td>
</tr>
<tr>
<td>8</td>
<td>125 124 5</td>
<td>-65.737 125</td>
<td>-59.044 125</td>
<td>-62.390 124</td>
<td>0.000</td>
</tr>
<tr>
<td>9</td>
<td>126 125 3</td>
<td>-56.733 126</td>
<td>-240.575 126</td>
<td>-148.654 125</td>
<td>-57.889</td>
</tr>
<tr>
<td>10</td>
<td>130 129 3</td>
<td>-40.907 130</td>
<td>-35.258 130</td>
<td>-38.083 129</td>
<td>-21.296</td>
</tr>
<tr>
<td>11</td>
<td>131 130 3</td>
<td>-32.901 131</td>
<td>0.000 131</td>
<td>0.000 130</td>
<td>-34.080</td>
</tr>
<tr>
<td>12</td>
<td>125 124 5</td>
<td>75.488 125</td>
<td>260.549 125</td>
<td>168.019 124</td>
<td>116.111</td>
</tr>
<tr>
<td>13</td>
<td>128 127 3</td>
<td>97.065 128</td>
<td>-281.531 128</td>
<td>-92.233 127</td>
<td>136.907</td>
</tr>
<tr>
<td>14</td>
<td>131 130 5</td>
<td>119.007 131</td>
<td>-252.761 131</td>
<td>-66.877 130</td>
<td>95.987</td>
</tr>
<tr>
<td>15</td>
<td>126 125 5</td>
<td>222.753 126</td>
<td>234.749 126</td>
<td>228.791 125</td>
<td>241.651</td>
</tr>
<tr>
<td>16</td>
<td>127 126 5</td>
<td>235.958 127</td>
<td>427.453 127</td>
<td>331.705 126</td>
<td>235.353</td>
</tr>
<tr>
<td>17</td>
<td>129 128 5</td>
<td>249.324 129</td>
<td>261.363 129</td>
<td>255.343 128</td>
<td>226.850</td>
</tr>
<tr>
<td>18</td>
<td>130 129 5</td>
<td>262.184 130</td>
<td>72.968 130</td>
<td>167.576 129</td>
<td>261.773</td>
</tr>
<tr>
<td>19</td>
<td>127 126 1</td>
<td>380.087 127</td>
<td>-740.404 127</td>
<td>-180.159 126</td>
<td>0.000</td>
</tr>
<tr>
<td>20</td>
<td>128 127 5</td>
<td>388.164 128</td>
<td>204.375 128</td>
<td>296.270 127</td>
<td>407.808</td>
</tr>
<tr>
<td>21</td>
<td>129 128 1</td>
<td>396.991 129</td>
<td>0.000 129</td>
<td>0.000 128</td>
<td>521.874</td>
</tr>
</tbody>
</table>

Comparison between BIPM program and calculation by Razet [53].
TRANSITION IN IODINE : \( R (47) \) \( 9-2 \)

Constants: \( \text{eqQ}' \) \( \text{eqQ}'' \) \( C' \) \( C'' \) \( A' \) \( A'' \) \( D' \) \( D'' \)

<table>
<thead>
<tr>
<th></th>
<th>( \text{eqQ}' )</th>
<th>( \text{eqQ}'' )</th>
<th>( C' )</th>
<th>( C'' )</th>
<th>( A' )</th>
<th>( A'' )</th>
<th>( D' )</th>
<th>( D'' )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-505.114000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-2453.127000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.025222</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-0.019813</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>-0.019410</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.000000</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Comparison between BIPM program and calculation by Razet [53].

<table>
<thead>
<tr>
<th>no</th>
<th>( F' )</th>
<th>( F'' )</th>
<th>I</th>
<th>( D-F=D-J )</th>
<th>( f )</th>
<th>( f+509.925 )</th>
<th>( f[53] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>43</td>
<td>42</td>
<td>5</td>
<td>-509.925</td>
<td>0.000</td>
<td>0.000</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>48</td>
<td>47</td>
<td>1</td>
<td>-486.730</td>
<td>23.195</td>
<td>23.195</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>53</td>
<td>52</td>
<td>5</td>
<td>-465.235</td>
<td>44.690</td>
<td>44.690</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>44</td>
<td>43</td>
<td>5</td>
<td>-238.928</td>
<td>270.997</td>
<td>270.998</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>49</td>
<td>48</td>
<td>3</td>
<td>-200.040</td>
<td>309.885</td>
<td>309.885</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>47</td>
<td>46</td>
<td>3</td>
<td>-189.537</td>
<td>320.388</td>
<td>320.388</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>52</td>
<td>51</td>
<td>5</td>
<td>-152.763</td>
<td>357.162</td>
<td>357.162</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>45</td>
<td>44</td>
<td>5</td>
<td>-71.313</td>
<td>438.612</td>
<td>438.612</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>46</td>
<td>45</td>
<td>5</td>
<td>-71.313</td>
<td>438.612</td>
<td>438.612</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>49</td>
<td>48</td>
<td>5</td>
<td>-45.297</td>
<td>464.628</td>
<td>464.628</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>51</td>
<td>50</td>
<td>3</td>
<td>-33.715</td>
<td>476.210</td>
<td>476.210</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>45</td>
<td>44</td>
<td>3</td>
<td>66.828</td>
<td>576.763</td>
<td>576.763</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>48</td>
<td>47</td>
<td>3</td>
<td>96.837</td>
<td>606.762</td>
<td>606.762</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>46</td>
<td>45</td>
<td>3</td>
<td>374.392</td>
<td>884.317</td>
<td>884.317</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>47</td>
<td>46</td>
<td>3</td>
<td>386.461</td>
<td>896.386</td>
<td>896.386</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>49</td>
<td>48</td>
<td>1</td>
<td>402.332</td>
<td>962.257</td>
<td>962.257</td>
<td></td>
</tr>
</tbody>
</table>

AIV-d