JOURNAL OF COMPUTATIONAL PHYSICS 40, 430–439 (1981)

Calculations of Atomic Spin-Orbit Matrix Elements in the Unitary Group Approach*

R. D. KENT, M. SCHLESINGER, AND G. W. F. DRAKE

Department of Physics, University of Windsor, Windsor, Ontario N9B 3P4, Canada

Received July 29, 1980

We present detailed closed-form expressions for the evaluation and numerical calculation of atomic spin-orbit matrix elements in the unitary group approach. A minicomputer implementation of the calculations is described.

1. INTRODUCTION

The conventional Racah scheme of atomic structure calculations becomes very cumbersome for systems involving many equivalent electrons. The use of seniority labels fails to label uniquely all the states for equivalent f electrons and beyond.

A number of workers [5] have developed simplified rule using the unitary group approach, based on the properties of $U(2l+1)$ and its unitary subgroups, for the application of the Weyl tableau representation of Gelfand basis states to problems involving many equivalent electrons.

Drake and Schlesinger [3] have worked out closed-form expressions for the matrix elements of one- and two-body operators, using graphical methods of angular momentum analysis. The same authors showed that spin dependent operators are simply related to purely orbital operators, thereby removing a long-standing difficulty with the tableaux or unitary group formalism. As a direct consequence, spin-orbit type matrix elements are directly expressible in terms of one- and two-body operators alone.

The purpose of the present paper is to report that we have programmed the evaluation of the spin-orbit matrix element in the unitary groups approach, on a PDP-11/03 minicomputer.[1] We assume that the reader is familiar with the relevant literature and, in particular, with the paper of Drake and Schlesinger [3].

We shall give in the next section a detailed account of the closed-form expressions involved in the evaluation and numerical calculation of the spin-own-orbit operator matrix elements. These expressions are a direct outgrowth of previous work [3].

* Research supported by the Natural Sciences and Engineering Council of Canada.

[1] Our operating system is DEC-RT11 and the hardware includes an RKO5 hard disk drive and a line printer.

0021-9991/81/040430–10$02.00/0

In the third section we shall give an account of the minicomputer implementation of the work. In the fourth section, a sample comparison of our results with those listed by Karwowski *et al.* [4] will be given.

2. Spin-Own-Orbit Matrix Elements

In the present section we shall consider the matrix elements of the spin-own-orbit operator, which is defined by

$$V_{SO} = \sum_{\gamma=-1}^{1} \sum_{i=1}^{N} l_{i,-\gamma} S_{i,\gamma} (-1)^{-\gamma}. \tag{2.1}$$

The matrix elements of (2.1) in the representation corresponding to the $|(\alpha); l^N S_0 M_L M_S\rangle$ vectors is of the form (cf. [3, Eqs. 13, 42–44 and text])

$$\begin{aligned}
&\langle(\alpha'); l^N S_0' M_L' M_S'| V_{SO} |(\alpha); l^N S_0 M_L M_S\rangle \qquad (2.2)\\
&= \sum_{\gamma=-1}^{1} \sum_{p=1}^{N} (-1)^{l-m_{l,p}'+\gamma} \sqrt{l(l+1)(2l+1)} \begin{pmatrix} l & 1 & l \\ -m_{l,p}' & -\gamma & m_{l,p} \end{pmatrix}\\
&\times (-1)^{S_0'-M_S'} \begin{pmatrix} S_0' & 1 & S_0 \\ -M_S' & \gamma & M_S \end{pmatrix} \frac{(-1)^{S_{-1}+S_0+1/2}}{\sqrt{6}} \begin{Bmatrix} S_0' & S_0 & 1 \\ \frac{1}{2} & \frac{1}{2} & S_{-1} \end{Bmatrix}^{-1}\\
&\times \langle \alpha_p' M_L' | E(\alpha_p', \alpha_p) | \alpha_p M_L \rangle\\
&\times \langle (\bar{\alpha}'); S_0' M_L' | E(\alpha_p', \alpha_0) E(\alpha_0', \alpha_p) + \tfrac{1}{2} E(\alpha_p', \alpha_p) | (\bar{\alpha}); S_0 M_L \rangle_{N+1},
\end{aligned}$$

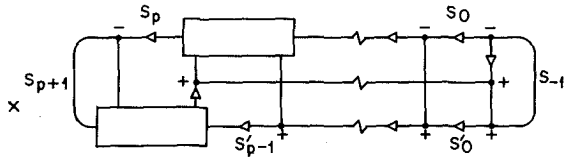
where the vectors $|(\alpha); l^N S_0 M_L M_S\rangle$ refer to the spin-adapted tableau states defined by the partition $\{a = N/2 - S_0,\ b = 2S_0,\ c = 2l + 1 - N/2 - S_0\}$ of $U(2l+1)$ and (α) refers to a particular arrangement of numbers α_i in each of the $2a + b = N$ tableau boxes. The labels N, l, S_0, M_L and M_S have the usual meaning.

The final two-body matrix element is to be evaluated between the states listed with one additional label $\alpha_0 = \alpha_0'$ and with the resulting tableau $(\bar{\alpha})$. The extra box is added in such a way as to bring the total spin of the two states into agreement, if they differ, i.e., $2S_{-1} = (S_0 + S_0') = 2S_{-1}'$ if $S_0' = S_0 \mp 1$ or $2S_{-1} = 2S_0' + 1$ if $S_0' = S_0$. The sum in (2.2) is over $p = 1 \cdots N$, rather than a sum over, say, p and q each varying independently from $1 \cdots N$ (as implied by Fig. 4 of [3]). This is so because the only non-zero matrix elements are between tableaux differing in M_L values of ± 1 or 0. Between any two such tableaux the matrix element will vanish unless $\alpha_i' = \alpha_i$ for all i except the value p, for which $\alpha_p' = \alpha_p + \gamma$. Note also that once the particular values of M_L, M_L' and M_S, M_S' have been specified only one value of γ will be operative.

The two-body matrix element in (2.2) can be expressed in terms of a graph, for $M'_L \geqslant M_L$,

$$3X_p = \langle (\bar{\alpha})\, \alpha'_0; S_{-1} M'_L | \, E(\alpha'_p, \alpha_0)\, E(\alpha'_0, \alpha_p) + \tfrac{1}{2} E(\alpha'_p, \alpha_p) \, | (\alpha)\, \alpha_0; S_{-1} M_L \rangle$$

$$= 3 \prod_{\substack{i=0 \\ \text{NP}}}^{p} [(2S_i + 1)(2S'_i + 1)]^{1/2} \times \prod_{\substack{i=p+1 \\ \text{NP}}}^{N} \delta(S_i, S'_i)$$



(2.3)

where the notation "NP" stands for no pairs, meaning that all pairs $\alpha_i = \alpha_{i+1}$ mutual to both tableaux are to be excluded. After breaking the graph and reassociating terms, we find

$$X_p = \tilde{E}_p \prod_{i=0}^{p-3} \tilde{M}(i) \prod_{i=p+1}^{N} \delta(S_i, S'_i)(-1)^{S_0 + S_{-1} + 1/2} \begin{Bmatrix} S_0 & S_0 & 1 \\ \frac{1}{2} & \frac{1}{2} & S_{-1} \end{Bmatrix}, \tag{2.4}$$

where

$$\tilde{M}(i) = \sqrt{(2S_i + 1)(2S'_i + 1)} \begin{Bmatrix} 1 & S'_i & S_i \\ \frac{1}{2} & S_{i+1} & S'_{i+1} \end{Bmatrix} (-1)^{S'_{i+1} + S_i - 1/2} \tag{2.5}$$

and $\tilde{M}(-1) = 1$, by definition. For $\tilde{E}_p$ we find four possible forms, namely,

$$\tilde{E}_p = \delta(S_p S'_p) \sqrt{(2S_{p-1} + 1)(2S'_{p-1} + 1)}(-1)^{S_p + S'_{p-1} - 1/2}$$
$$\times \begin{Bmatrix} S'_{p-1} & S_{p-1} & 1 \\ \frac{1}{2} & \frac{1}{2} & S_p \end{Bmatrix} \tilde{M}(p-2) \qquad \text{if} \quad \alpha_p \neq \alpha_{p-1} \alpha'_p \neq \alpha'_{p+1}$$

$$= \delta(S_p S'_p)\, \delta(S_p S_{p-2})(-1)^{2S'_{p-1} + S_{p-2} - S'_{p-2}} \sqrt{(2S'_{p-1} + 1)(2S'_{p-2} + 1)}$$
$$\times \begin{Bmatrix} S_{p-2} & S'_{p-2} & 1 \\ \frac{1}{2} & \frac{1}{2} & S'_{p-1} \end{Bmatrix} \qquad \text{if} \quad \alpha_p = \alpha_{p-1} \alpha'_p \neq \alpha'_{p+1}$$

$$= \delta(S'_{p+1} S'_{p-1})(-1)^{2S_p + 1} \sqrt{(2S_p + 1)(2S_{p-1} + 1)}$$
$$\times \begin{Bmatrix} S'_{p-1} & S_{p-1} & 1 \\ \frac{1}{2} & \frac{1}{2} & S_p \end{Bmatrix} \tilde{M}(p-2) \qquad \text{if} \quad \alpha_p \neq \alpha_{p-1} \alpha'_p = \alpha'_{p+1}$$

$$= \delta(S_p S_{p-2})\, \delta(S'_{p+1} S'_{p-1}) \sqrt{(2S_{p-2} + 1)(2S'_{p-2} + 1)}(-1)^{S'_{p-2} + S'_{p-1} - 1/2}$$
$$\times \begin{Bmatrix} S_{p-2} & S'_{p-2} & 1 \\ \frac{1}{2} & \frac{1}{2} & S'_{p-1} \end{Bmatrix} \qquad \text{if} \quad \alpha_p = \alpha_{p-1} \alpha'_p = \alpha'_{p+1}. \tag{2.6}$$

With (2.3)–(2.6) in (2.2) we find for the reduced matrix element, for $\gamma = 0, -1$ $(M'_L = M_L,\ M'_L = M_L + 1)$

$$\langle(\alpha'); l^N S'_0 M'_L \| V^\gamma_{SO} \| (\alpha); l^N S_0 M_L\rangle$$

$$= (-1)^{S'_0 - M'_S} \begin{pmatrix} S' & 1 & S \\ -M'_S & \gamma & M_S \end{pmatrix}^{-1} \langle(\alpha)' I; l^N S'_0 M'_L M'_S | V^\gamma_{SO} | (\alpha) I; l^N S_0 M_L M_S\rangle$$

$$= \sum_{p=1}^{N} (-1)^{l - m'_{l,p} + \gamma} \sqrt{l(l+1)(2l+1)} \begin{pmatrix} l & 1 & l \\ -m'_{l,p} & -\gamma & m_{l,p} \end{pmatrix} \sqrt{\frac{3}{2}}\, \tilde{E}_p \prod_{i=0}^{p-3} \tilde{M}(i)$$

$$\times \delta(\alpha'_p, \alpha_p + \gamma) \prod_{i=1}^{p-1} \delta(\alpha'_i, \alpha_i) \prod_{i=p+1}^{N} \delta(\alpha_i, \alpha'_i) \prod_{i=p+1}^{N} \delta(S_i, S'_i). \tag{2.7}$$

It is obvious that the case $\gamma = 1$ $(M'_L = M_L - 1)$ need not be considered separately as for that case one merely changes all primed quantities to unprimed, and vice versa, and then take $\gamma = -1$.

We now consider each of the two cases $\gamma = 0$, $\gamma = -1$ separately.

Case A. $\gamma = 0$ $(M'_L = M_L)$.

In this case $\alpha_i = \alpha'_i$ $(i = 1, N)$. After removing all mutual pairs in our tableaux, we find $\alpha_i \neq \alpha_j$ $(i \neq j = 1,..., N)$ and (2.7) takes the form

$$\langle(\alpha'); l^N S'_0 M'_L \| V^0_{SO} \| (\alpha); l^N S_0 M'_L\rangle$$

$$= \prod_{i=1}^{N} \delta(\alpha_i \alpha'_i) \sum_{p=1}^{N} \prod_{i=p}^{N} \delta(S_i S'_i)(l + 1 - \alpha'_p)(-1)^{S_p + S'_{p-1} - 1/2} \sqrt{\frac{3}{2}}$$

$$\times \sqrt{(2S_{p-1} + 1)(2S'_{p-1} + 1)} \begin{Bmatrix} S'_{p-1} & S_{p-1} & 1 \\ \frac{1}{2} & \frac{1}{2} & S_p \end{Bmatrix} \prod_{i=0}^{p-2} \tilde{M}(i). \tag{2.8}$$

In the special instance where $2S_0 = N$ and $S_0 = S'_0$, i.e., single-column tableaux, or Slater determinant, we find

$$\langle(\alpha'); l^N S'_0 M'_L \| V^0_{SO} \| (\alpha); l^N S'_0 M'_L\rangle_{N=2S'_0} = (-1)^{N+1} \frac{M_L}{2S'_0} \sqrt{S'_0(S'_0 + 1)(2S'_0 + 1)} \tag{2.9}$$

using the fact that $S_i = S'_i$ and $S_{i+1} = S_i - \frac{1}{2}$.

Case B. $\gamma = -1$ $(M'_L = M_L + 1)$.

In this case $\alpha'_i = \alpha_i$ for all i except $\alpha'_p = \alpha_p - 1$ and (2.7) takes the form

$$\langle(\alpha'); l^N S'_0 M'_L \| V^{-1}_{SO} \| (\alpha); l^N S_0 M_L\rangle$$

$$= \delta(M'_L, M_L + 1) \prod_{i=1}^{p-1} \delta(\alpha'_i, \alpha_i) \prod_{i=p+1}^{N} \delta(\alpha'_i, \alpha_i) \prod_{i=p+1}^{N} \delta(S'_i S_i)$$

$$\times \sqrt{\frac{3\alpha'_p(2l + 1 - \alpha'_p)}{4}}\, \tilde{E}_p \prod_{i=0}^{p-3} \tilde{M}(i). \tag{2.10}$$

Again for the special case of $S_0' = S_0 = N/2$ we will have

$$\langle (\alpha'); l^N S_0' M_L' \| V_{SO}^{-1} \| (\alpha); l^N S_0' M_L' - 1 \rangle_{N=2S_0'}$$
$$= (-1)^{N+1} \sqrt{\frac{\alpha_p'(2l+1-\alpha_p')}{2}} \times \sqrt{\frac{(S_0'+1)(2S_0'+1)}{4S_0'}}. \tag{2.11}$$

In closing this section it should be noted that once the computations of the V_{SO} matrix elements has been completed, the transformation to states of, say, definite L is quite simple. We write

$$\langle l^N; L'(K')\, S' M_L' \tau' \| V_{SO} \| l^N; L(K)\, S M_L \tau \rangle$$
$$= \sum_{I'=1}^{Q_{M_L'}'} \sum_{I=1}^{Q_{M_L}} \Phi_{I'}'(L'(K')\, S' M_L' \tau')\, \Phi_I(L(K)\, S M_L \tau)$$
$$\times \langle (\alpha')\, I'; l^N S' M_L' \| V_{SO} \| (\alpha) I; l^N S M_L \rangle. \tag{2.12}$$

The double summation ranges independently over the Q_{M_L} and $Q_{M'_L}'$ tableaux at the M_L and M_L' levels, respectively, and the transformation coefficients $\Phi_I(L(K)\, S_0 M_L \tau)$ are those defined by Drake *et al.* [1] (to within a phase)

$$|l^N; L S_0 M_L M_S \tau\rangle = \sum_{I=1}^{Q_{M_L}} \Phi_I(L S_0 M_L \tau)\, |(\alpha) I; l^N S_0 M_L M_S\rangle, \tag{2.13}$$

where the index I refers to the Ith tableau in the list of Q_{M_L} tableaux at the M_Lth level.

In our notation the $\Phi_I(L(K)\, S M_L \tau)$ are defined recursively by

$$\Phi_I(L(K)\, S M_L \tau) = -\sqrt{\frac{2}{(L+M_L+1)(L-M_L)}} \sum_{I'=1}^{Q_{M_L+1}} \left[\sqrt{\frac{\alpha_p'(2l+1-\alpha_p')}{2}}\, E_p \right]_{I,I'}$$
$$\times \Phi_{I'}(L(K)\, S M_L + 1\tau) \tag{2.14}$$

corresponding to Eq. (9) of [1] and

$$\Phi_I(L(K)\, S_0 M_L \tau) = D^{-1} \tilde{\Phi}_I(L(K)\, S_0 M_L \tau) \tag{2.15}$$
$$= D^{-1} \sum_{L(K')=M_L}^{L_{\max}} [\delta(I,K) - \Phi_I(L(K')\, S_0 M_L \tau)\, \Phi_K(L(K')\, S_0 M_L \tau)]$$

corresponding to the projection operator technique outlined in Eqs. (11) and (12) and text of [1]. The bracketed quantity in (2.14) refers to the matrix element E_p between

the I'th tableau at level $M_L + 1$ and the Ith tableau at level M_L multiplied by a factor which depends on the α'_p which is changing ($\alpha_p = \alpha'_p + 1$). In (2.15) D is simply a normalization factor given by

$$D^2 = \sum_{I=1}^{Q_{M_L}} [\tilde{\Phi}_I(L(K)\, S_0 M_L \tau)]^2. \tag{2.16}$$

Note that the $\Phi_I(L(K)\, S_0 M_L \tau)$ satisfy

$$\sum_{I=1}^{Q_{M_L}} \Phi_I(L(K)\, S_0 M_L \tau)\, \Phi_I(L'(K')\, S_0 M_L \tau') = \delta(L, L')\, \delta(K, K')\, \delta(\tau, \tau'). \tag{2.17}$$

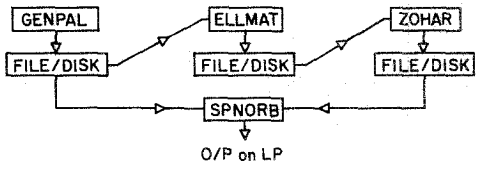
3. Programming Considerations

The entire scheme of the spin-own-orbit matrix element calculation, in the unitary group approach, has now been programmed. We have coded the programs in PDP assembler language. They constitute a string of four programs each less than 20K words (one word = 16 bits) in length so that a unit of limited size can handle them sequentially. The first of these, "GENPAL," generates and stores on disk the entire list of tableaux corresponding to a given partition. The second, "ELLMAT," generates and stores on disk the matrix elements of the lowering operator L^1_{-1} (cf. [1]) to be used in the generation of the $|l^N; LSM_L M_S \tau\rangle$ vectors. The third program, "ZOHAR," calculates and stores the numerical factors $\Phi_I(LSM_L \tau)$ (cf. Eqs. (2.13)–(2.15)) multiplying the various tableau components of these vectors.

Finally, the fourth program, "SPNORB," uses the files generated by "GENPAL" and by "ZOHAR" to calculate and eventually output the matrix elements of the reduced spin orbit operator, i.e.,

$$\langle l^N M'_S; L'(K')\, S' M'_L \tau' \| V_{SO} \| l^N M_S; L(K)\, SM_L \tau\rangle.$$

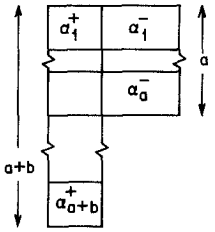
The following schematic diagram illustrates the relations among the four programs



In the first program we utilize an approach [5] whereby a Weyl tableau is represented in one "word," PALB(α), such that

$$\text{PALB}(\alpha) = \sum_{i=1}^{a+b} 2^{2\alpha_i^+ - 2} + \sum_{i=1}^{a} 2^{2\alpha_i^- - 1},$$

where the $\alpha_i^{\pm}$ represent the tableaux labels in the right and left columns



We start with the tableau corresponding to $M_L(\max)$. It is, for a given tableau shape (partition), given by

$$\mathrm{PALB}(\max) = \sum_{i=1}^{a+b} 2^{2i-2} + \sum_{i=1}^{a} 2^{2i-2},$$

that is, the choice of minimum $\alpha_i^{\pm}$ labels. Next the program proceeds to generate the remaining tableaux at successive M_L levels by operating with a "one-step" operator (cf. [1]). This is the factor E_p in Eq. (2.14).

The second and third programs follow the methods described in [1]. Starting with the tableau with maximum M_L, the lowering operator connects tableaux in each M_L row with the ones in the row beneath (viz. Eq. (2.14)). This way all Q_{M_L+1} vectors $|l^N; LSM_L M_S \tau\rangle$ with $L = M_L + 1,\ldots, L_{\max}$ in the M_Lth level can be generated from the vectors in the $(M_L + 1)$th level. The remaining $Q_{M_L} - Q_{M_L+1}$ vectors $|l^N; L = M_L, SM_L M_S \tau\rangle$ are generated through a projection operator technique. As there may be more than one such vector for a given M_L value, we write our vectors as

$$|l^N; L(K)S\, M_L M_S \tau\rangle,$$

where K indicates the column of the projection operator from which the vector was extracted (viz. Eq. (2.15)). The fourth and last program is coded to utilize expressions (2.8) and (2.10) together with (2.12).

Consequently, the final output of "SPNORB" is a reduced matrix element

$$\langle l^N; L'(K')\, S' M'_L \tau' \| V_{\mathrm{SO}} \| l^N; L(K)\, SM_L \tau\rangle,$$

where all symbols stand for their usual meaning and τ stands for any additional quantum numbers needed to specify the states involved.

Regarding the use of minicomputers and assembler language two points should be made. First, since minicomputers are becoming commonplace in many laboratories, it is important to realize what complicated a job can be performed with them. Second, although the detailed expressions can be evaluated using most high-level languages, inherent in this method are a number of operations, including single bit manipulation and arbitrary precision techniques which naturally suggest the use of assembler codes. In addition, it should be noted that assembler codes greatly increase the effec-

tiveness of the programming when measured in terms of run-time and/or memory allocation.

We have compared our data for the reduced matrix elements with those tabulated by Karwowski *et al.* [4] in an $|l^N; LSJM_J\tau\rangle$ representation. We have found full agreement up to a roundoff error. Note that in our method the results are calculated and expressed as the square root of a ratio of integers and are therefore exact.

4. A Sample Calculation

Let us consider the example of the diagonal, reduced matrix elements of the two $L = 7$ states of $^4f^5$ (i.e., $l = 3$, $N = 5$ and $S = \frac{3}{2}$) with projections $M_L = L$ and $M_S = S$ (i.e., the "stretched" states). This example serves to illustrate the use of Eq. (2.8) and as a comparison with the decimal values tabulated by [4].

Recalling that the vectors are defined by $|l^N; L(\tau)\, SM_L M_S\rangle$ we have in terms of tableaux (as calculated by "ZOHAR")

$$|f^5; 7(1)\tfrac{3}{2}\, 7\tfrac{3}{2}> = \sqrt{\tfrac{33}{68}}\ \left|\begin{array}{|c|c|}\hline 1&2\\\hline 2&\tfrac{3}{2}\\\hline 3\\\cline{1-1}4\\\cline{1-1}\end{array}\right\rangle + \sqrt{\tfrac{55}{136}}\ \left|\begin{array}{|c|c|}\hline 1&2\\\hline 2&\tfrac{3}{2}\\\hline 3\\\cline{1-1}5\\\cline{1-1}\end{array}\right\rangle + \sqrt{\tfrac{45}{748}}\ \left|\begin{array}{|c|c|}\hline 1&1\\\hline 2&\tfrac{3}{2}\\\hline 4\\\cline{1-1}5\\\cline{1-1}\end{array}\right\rangle + \sqrt{\tfrac{75}{1496}}\ \left|\begin{array}{|c|c|}\hline 1&1\\\hline 2&\tfrac{3}{2}\\\hline 3\\\cline{1-1}6\\\cline{1-1}\end{array}\right\rangle \tag{4.1}$$

$$|f^5; 7(3)\tfrac{3}{2}\, 7\tfrac{3}{2}> = \sqrt{\tfrac{5}{11}}\ \left|\begin{array}{|c|c|}\hline 1&1\\\hline 2&\tfrac{3}{2}\\\hline 4\\\cline{1-1}5\\\cline{1-1}\end{array}\right\rangle - \sqrt{\tfrac{6}{11}}\ \left|\begin{array}{|c|c|}\hline 1&1\\\hline 2&\tfrac{3}{2}\\\hline 3\\\cline{1-1}6\\\cline{1-1}\end{array}\right\rangle \tag{4.2}$$

where the $\tau = 1, 3$ label denotes that columns 1 and 3 of the projection matrix defined by [2] (cf. also Eq. (2.15) above) were used to generate the transformation coefficients shown above.

The diagonal reduced matrix element of (4.1) is therefore expressed as

$$\begin{aligned}&\langle f^5; 7(1)\tfrac{3}{2}\, 7; \tfrac{3}{2}\| V_{SO} \| f^5; 7(1)\tfrac{3}{2}\, 7; \tfrac{3}{2}\rangle\\ &= \tfrac{33}{68}\left\langle\begin{array}{|c|c|}\hline 1&3\\\hline 2\\\cline{1-1}3\\\cline{1-1}4\\\cline{1-1}\end{array}\right\| V_{SO} \left\|\begin{array}{|c|c|}\hline 1&3\\\hline 2\\\cline{1-1}3\\\cline{1-1}4\\\cline{1-1}\end{array}\right\rangle + 2\sqrt{\tfrac{33\cdot 55}{68\cdot 136}}\left\langle\begin{array}{|c|c|}\hline 1&3\\\hline 2\\\cline{1-1}3\\\cline{1-1}4\\\cline{1-1}\end{array}\right\| V_{SO} \left\|\begin{array}{|c|c|}\hline 1&2\\\hline 2\\\cline{1-1}3\\\cline{1-1}5\\\cline{1-1}\end{array}\right\rangle\\ &\quad + \cdots\cdots + \tfrac{75}{1496}\left\langle\begin{array}{|c|c|}\hline 1&1\\\hline 2\\\cline{1-1}3\\\cline{1-1}6\\\cline{1-1}\end{array}\right\| V_{SO} \left\|\begin{array}{|c|c|}\hline 1&1\\\hline 2\\\cline{1-1}3\\\cline{1-1}6\\\cline{1-1}\end{array}\right\rangle\end{aligned} \tag{4.3}$$

with a similar expression for (4.2). The cross-terms in (4.3) will all vanish, since the tableaux differ in each case by more than one α-label, leaving us with a sum of

diagonal tableau matrix elements only. Defining the sum of (4.3) and the corresponding value for the state (4.2) as W, we find

$$W = \left(-\sqrt{\frac{375}{16}}\right) + \left(-\sqrt{\frac{5}{3}}\right) = -\frac{19}{4}\sqrt{\frac{5}{3}}. \tag{4.4}$$

Now before we can compare our values with those of [4] we must perform a transformation to $J - M_J$ labelled states. Since we are concerned only with the $J = M_J$ case we will have

$$\begin{aligned} |f^5; L(\eta)\, SJJ\rangle &= \sum_{M_L, M_S} \sum_{\tau} C_{\eta\tau} \langle LSM_L M_S \,|\, JM_J\rangle\, \delta_{M_J, J} \\ &\quad \times |f^5; L(\tau)\, SM_L M_S\rangle \\ &= \sum_{\tau} C_{\eta\tau}\, |f^5; L(\tau)\, SM_L M_S\rangle\, \delta_{M_L, L}\, \delta_{M_S, S}, \end{aligned} \tag{4.5}$$

where $C_{\eta\tau}$ are the elements of a unitary matrix. (Recall that the breakdown of Racah's labelling scheme occurs precisely because of this indefiniteness in the labelling of states of the same L.)

If we suppress all labels except η and τ, we can write for the diagonal matrix elements

$$\langle \eta |\, V_{\mathrm{SO}} \,| \eta \rangle = \sum_{\tau, \tau'} C_{\eta\tau'} C_{\eta\tau} \langle \tau' |\, V_{\mathrm{SO}} \,| \tau \rangle. \tag{4.6}$$

Summing over η we find

$$\begin{aligned} \sum_{\eta} \langle \eta |\, V_{\mathrm{SO}} \,| \eta \rangle &= \sum_{\tau, \tau'} \sum_{\eta} C_{\eta\tau'} C_{\eta\tau} \langle \tau' |\, V_{\mathrm{SO}} \,| \tau \rangle \\ &= \sum_{\tau} \langle \tau |\, V_{\mathrm{SO}} \,| \tau \rangle, \end{aligned} \tag{4.7}$$

where we have used the fact that the $C_{\eta\tau}$ form a unitary matrix. In terms of the reduced matrix elements we have, from (4.4)

$$\begin{aligned} \sum_{\eta} \langle \eta \|\, V_{\mathrm{SO}} \,\| \eta \rangle_{\mathrm{KSF}} &= \sqrt{\frac{(L+1)(2L+1)}{L}} \times W \\ &= \left(\sqrt{\frac{8 \cdot 15}{7}}\right)\left(-\frac{19}{4}\sqrt{\frac{5}{3}}\right) \\ &= -\frac{95}{2}\sqrt{\frac{2}{7}}, \end{aligned} \tag{4.8}$$

where the L dependent factor derives from the different definitions of the reduced matrix elements in [4] and this work.

The left-hand side of (4.8) is, from [4],

$$\langle {}^4_5K1; f^5 \| V_{SO} \| {}^4_5K1; f^5 \rangle = 11.2996545791, \tag{4.9}$$

$$\langle {}^4_5K2; f^5 \| V_{SO} \| {}^4_5K2; f^5 \rangle = 14.0901634025, \tag{4.10}$$

which yields the sum

$$\sum_{\eta} \langle \eta \| V_{SO} \| \eta \rangle_{KSF} = 25.3898179816. \tag{4.11}$$

To within a phase and roundoff error there is precise agreement between (4.11) and (4.8). Note in (4.9) and (4.10) the notation is $|\, {}^{2S+1}_{v} L(\eta); l^N \rangle$, where spectroscopic L-labelling is used and v is the seniority number which fails to differentiate between the two K-states, thus requiring the additional label η. In the definite "L" transformed unitary group scheme, the single arbitrary label τ is needed [viz. (4.1) and (4.2)] to perform the same task, since the states are defined by the particular linear combinations of tableau states, to within a unitary transformation.

5. Conclusion

The explicit expressions used in the minicomputer implementation as presented in this work help demonstrate once again [2] the relative simplicity, power and versatility of the unitary group approach to spectroscopic calculations. Despite its obvious advantages, the method has yet to gain the acceptance it deserves in atomic physics. In contrast, the use of unitary group theory by quantum chemists for solving molecular problems is becoming more common [6]. Our work at present is directed toward the treatment of systems of non-equivalent (mixed configuration) electrons and it appears that the expressions involved will also lend themselves to an easy implementation on the minicomputer.

References

1. J. Drake, G. W. F. Drake, and M. Schlesinger, *J. Phys. B.* **8** (1975), 1009–1015.
2. J. Drake, G. W. F. Drake, and M. Schlesinger, *Phys. Rev. A.* **15** (1977), 807–809.
3. G. W. F. Drake and M. Schlesinger, *Phys. Rev. A.* **15** (1977), 1990–1999.
4. J. Karwowski, K. M. S. Saxena, and S. Fraga, Technical Report TC-ME-IV-75, Dept. of Chemistry, Univ. of Alberta, unpublished.
5. J. Paldus, *Phys. Rev. A.* **14** (1976), 1620–1625 and references therein.
6. I. Shavitt and L. T. Redmon, *J. Chem. Phys.*, in press.