Computer Physics Communications 64 (1991) 369–398
North-Holland

The MCHF atomic-structure package

Charlotte Froese Fischer

Department of Computer Science, Box 6035 B, Vanderbilt University, Nashville, TN 37235, USA

Received 28 August 1990

The multi-configuration Hartree–Fock code (MCHF) determines radial functions that define the orbitals of configuration states in the expansion of a non-relativistic wave function of a many-electron system. This code has been combined with others that perform angular and/or radial integrations to form an atomic-structure package for the prediction of atomic properties, called the MCHF atomic-structure package. This article describes the scope of the package, its design philosophy, organization, data input, and common file formats. Through examples it also outlines the series of steps that typically are required in the prediction of an atomic property. The present package is restricted to energy levels and transition probabilities, either in the non-relativistic *LS* framework or the Breit–Pauli, relativistic *LSJ* framework.

1. Introduction

Atomic-structure calculations have played an important role in atomic spectroscopy, both in the prediction of atomic properties and in the interpretation of experimental data. Astrophysics, fusion research, collision phenomena, all rely extensively on atomic-structure calculations. In addition, since atoms are the building blocks of molecules, liquids, and solids, the results of atomic calculations are of fundamental importance to these areas as well.

The MCHF method [1] has been shown to be effective for the study of correlation (where correlation is used to designate the error in the independent particle, Hartree–Fock model) and for problems where relativistic effects are small. The systems where it can be used most effectively are many-electron systems that can be modeled as two-electron systems [2] outside a closed core. Magnesium is one such example [3]. But it has been applied to some difficult cases where a perturber is embedded in a Rydberg series and mixing depends critically on correlation and possibly also on the *LS*-dependent relativistic shift effects [4]. More recently, it has been used to study the electron affinities of Ca [5] and other alkaline earths. Relativistic corrections may be included through the Breit–Pauli approximation. This approach has been used in a study of the boron sequence [6] where the fine-structure splitting obtained from multi-configuration Dirac–Fock calculations was not in good agreement with observation and a correction process was needed to obtain the correct non-relativistic limit [7]. It has also been used to study forbidden transitions in the carbon [8], nitrogen [9], and oxygen [10] sequence. Thus the MCHF (or MCHF + BP) method is a versatile method for the study of atomic structure for a large class of problems. The present software package is based on this method.

2. Design considerations

Several atomic-structure programs are already available – SUPERSTRUCTURE [11] for which the radial functions can be either statistical model-potential radial functions or user-supplied functions, CIV3

[12] where the radial functions are expanded in terms of Slater orbitals, and Cowan's program [13] designed largely for highly ionized systems. All these are large, general programs often used on "supercomputers". The present system differs appreciably in its design in several ways:

(i) Modularity

Instead of one program capable of performing many different calculations, the present package is organized as a series of programs and utilities, sharing several libraries.

(ii) File orientation

Information is communicated between programs through files and large arrays or lists are stored as files. The main requirement is an adequate amount of disk space for the problem at hand, facilitated by a large, virtual memory system.

(iii) Ease of use

Human error can be reduced through the development of an easy-to-use system. Several approaches have been taken to minimize blunders. First, a spectroscopic notation has been introduced for specifying configurations. The configurations and the coupling of their subshells is always checked for validity. Second, input may be given in answer to prompts and default values may be selected. Unless programs are used frequently, it is easy to forget the format of the input data. By having the program prompt for information, the user is freed from remembering this detail, provided the programs can be executed interactively. Also, programs are initialized for a certain size, a constraint which can readily be changed but needs to be remembered. All programs check that these basic constraints are not exceeded, thus avoiding erroneous calculations.

(iv) Portability

Computer systems are changing rapidly. In such an environment it is vital that the software be relatively portable, that is, capable of being transferred from one system to another with minimal modification.

The most portable language to date has been FORTRAN. Though programs written in FORTRAN are often badly structured, FORTRAN77 goes a long way towards meeting this goal, and is available on most computers on the market. This language supports the character variable needed to develop a friendly user interface. It also allows for free-format input.

Many earlier FORTRAN program segments have been incorporated into the present system without revision, but all new routines have been developed using FORTRAN77.

(v) Efficiency

When extensive calculations are undertaken, it often is desirable to study some atomic property along an iso-electronic sequence. The angular integrations to be described in a later section, are the most CPU-time-consuming task (at least in large MCHF + BP calculations on the Cray where the angular codes do not vectorize). At the same time, the angular integrations are independent of the nuclear charge Z, and need be performed only once per sequence, provided the configuration expansion of the wave function remains unchanged. For this reason, angular integrations are always separated from tasks requiring radial functions which, of course, are Z-dependent.

Efficiency has been a factor influencing the design at all stages. In this connection, a revised program MCHF88 [14] has been developed in which the basic data structure describing the energy expression is integral driven and exploits the sparsity of many of the arrays appearing in the problem. Also, it is designed with large calculations in mind and deals more efficiently than earlier versions with a typical situation, one where only a few orbitals are varied at a time. The dimensions of the present version have *not* been set for large cases (as may be required by state-of-the-art calculations), but may readily be increased by increasing parameter values.

3. A conceptual view of atomic-property evaluation

Every atomic property is the expectation value of some operator, i.e.

$$\langle \text{property} \rangle = \langle \psi_{\text{f}} | \text{OP} | \psi_{\text{i}} \rangle, \tag{1}$$

where OP is an appropriate operator and ψ_{i}, ψ_{f} are wave functions for the initial and final state, respectively. In some cases, such as the energy, the property is for a given atomic state in which case $\psi_{\text{f}} \equiv \psi_{\text{i}}$.

In the configuration model, the wave function is described in terms of configuration-state functions (CSF). Unlike a Hartree–Fock calculation, where the label of the state designates the configuration and the only decision might be the coupling of the open subshells, a multi-configuration calculation requires the selection of a set of configuration states that define the wave-function expansion. In the MCHF method, the total wave function, $\Psi(\gamma LS)$, which is an atomic-state function (ASF) describing a many-electron system labeled γ (usually the dominant configuration in the expansion) and total angular and spin momenta LS, is approximated by

$$\psi = \sum_{i=1}^{m} c_i \Phi(\gamma_i LS), \tag{2}$$

where $\Phi(\gamma LS)$ is a configuration-state function with a specific coupling scheme (sometimes also referred to as a coupled configuration). In particular, for an N-electron system,

$$\Phi(\gamma_i LS) = A \left\{ \left\{ \prod_{j=1}^{N} R(n_j l_j;\, r_j) \right\} |(l_1, l_2, \ldots, l_N)\gamma_i LS\rangle \right\}$$

is a normalized antisymmetric product of a radial factor and a spin-angular factor. The radial factor consists of a product of radial functions. All the programs in this package use Racah algebra for the manipulation (recoupling) and integration of angular factors.

Substituting eq. (2) into eq. (1) we then get the result

$$\langle \text{property} \rangle = \sum_{i,i'} c_i c_{i'} \langle \gamma_{i'} LS | \text{OP} | \gamma_i LS \rangle. \tag{3}$$

The operator matrix element may be expanded further as

$$\langle \gamma_{i'} LS | \text{OP} | \gamma_i LS \rangle = \sum_{j,j',k} \text{Coeff}(k,\, a_{j'},\, a_j) \text{RI}^k(a_{j'},\, a_j), \tag{4}$$

where a_j and $a_{j'}$ refer to sets of one or more electrons (or radial functions) of the initial and final state, respectively, RI is a "reduced integral", and k refers to any additional parameters that may be present. Substituting eq. (4) into eq. (3) we get

$$\langle \text{property} \rangle = \sum_{i,i'} c_i c_{i'} \sum_{j,j',k} \text{Coeff}(k,\, a_{j'},\, a_j) \text{RI}^k(a_{j'},\, a_j). \tag{5}$$

Suppose for the time being that the radial functions and the wave-function expansions are known. The task of evaluating an atomic property then breaks down into two subtasks – the angular integration of eq.

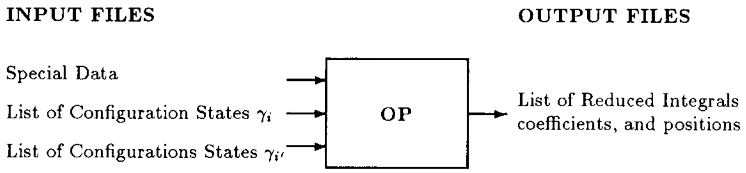

Fig. 1. Schematic for an angular integration of a general atomic property.

(4) producing the list of coefficients, Coeff(k, $a_{j'}$, a_j) along with the list of unevaluated integrals $\mathrm{RI}^k(a_{j'}, a_j)$, and the evaluation of eq. (5) where now Coeff(k, $a_{j'}$, a_j) are known and reduced integrals must be evaluated. The former is a Z-independent task, the latter Z-dependent. The result of the angular integrations is a list of reduced integrals, their coefficients, and positions i', i in the expansions of the bra and ket atomic-state functions, respectively. For brevity, we shall refer to this list as the "list of reduced integrals" for a specified operator.

Figure 1 depicts a schematic diagram for the flow of data in an angular integration. Notice that such a calculation requires a small amount of SPECIAL data that may be entered from the terminal (or a special file, if the user wishes to prepare the data in advance) and two lists of configurations. In some instances, if the initial and final states are the same, only one configuration list is required.

Figure 2 depicts the property evaluation task. Now all information about the wave function for the initial and final state is needed (configuration-expansion coefficients as well as radial functions). This is combined with the reduced-integral list. Each reduced integral must be evaluated and its contribution added to the sum defining the atomic property.

Thus it is convenient to represent a wave function by two files – one called the CONFIGURATION file, which lists the configurations, their coupling and possibly also their weight, and the other the file of radial functions. The former will be referred to as a "cfile" and the latter a "wfile". The formats of these and other files are discussed in sections 7–9.

An important property is the energy of a state since in the MCHF approximation it is used to determine the radial functions. The above description of the prediction fo an atomic property has not specified whether the state is viewed as an LS (non-relativistic) or LSJ (relativistic) state. In fact, in most cases, the only difference is the wave-function expansion itself, but the energy is an exception in that a restricted amount of non-orthogonality is allowed in non-relativistic calculations. Non-orthogonality has not yet been extended to a Breit–Pauli calculation.

The MCHF + BP method assumes that the radial functions will be determined using a non-relativistic form of the Hamiltonian. Thus the first stage of a wave-function calculation is the generation of the reduced-integral list for the non-relativistic Hamiltonian, H_{NR}, which consists of the well-known Slater integrals, F^k, G^k, R^k, and the one-electron integral I. In an MCHF calculation it is convenient to use the l-dependent operator

$$L = \frac{\mathrm{d}^2}{\mathrm{d}r^2} + \frac{2Z}{r} - \frac{l(l+1)}{r^2}.$$

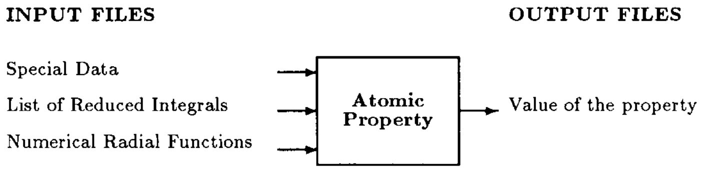

Fig. 2. Schematic for atomic property evaluation.

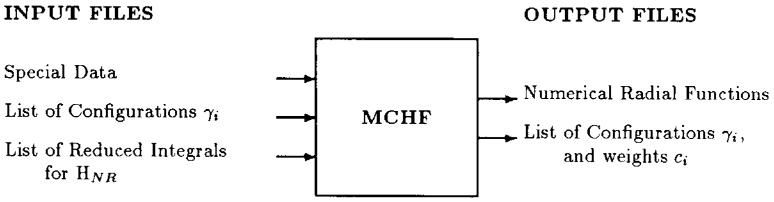

Fig. 3. Schematic for an MCHF calculation.

Then

$$I(nl,\, n'l) = -\tfrac{1}{2}\langle nl' \,|\, L \,|\, nl \rangle.$$

The latter is referred to as an "L-integral" and replaces I in the reduced list.

Figure 3 depicts the data flow for an MCHF calculation. An MCHF calculation usually is for a particular LS state, or possibly the average energy of some configuration. More generally, it is a calculation determining radial functions, and mixing coefficients or weights, for a given energy expression. The latter need not be physically meaningful. If the purpose of the calculation is to generate a balanced set of radial functions, simultaneously representing several states, say, the energy expression could be a linear combination of energy expressions for several configurations, not necessarily for the same LS. In such applications, the diagonalization process must be by-passed and the mixing coefficients kept fixed during the self-consistent field procedure.

Once radial functions have been determined relativistic effects may be added through the Breit–Pauli approximation. In the present design, it is convenient to view the Breit–Pauli Hamiltonian as a sum [15], namely

$$H_{\mathrm{BP}} = H_{\mathrm{NR}} + H_{\mathrm{R}},$$

where H_{NR} is the usual non-relativistic Hamiltonian and H_{R} represents the relativistic contributions. The latter may again be subdivided into non-fine-structure and fine-structure contributions, namely

$$H_{\mathrm{R}} = H_{\text{non-fine}} + H_{\text{fine}},$$

where

$$H_{\text{non-fine}} = H_{\text{mass}} + H_{\text{Darwin}} + H_{\mathrm{SSC}} + H_{\mathrm{OO}}$$

and

$$H_{\text{fine}} = H_{\mathrm{SO}} + H_{\mathrm{SOO}} + H_{\mathrm{SS}}.$$

The non-fine structure contributions are not J dependent and, except for H_{OO}, the coefficients for the associated reduced integrals are directly related to coefficients already determined for the non-relativistic Hamiltonian. The orbit–orbit contribution is lengthy and not deemed sufficiently important to be included in the MCHF package. CIV3 [12] also neglects this effect and SUPERSTRUCTURE [11] neglects the two-body Darwin term and the spin–spin contacts terms as well. The effect of the non-fine structure is mainly one of shifting one configuration relative to another. For this reason, the effect is sometimes referred to as the "relativistic shift effect".

The fine-structure contribution is responsible for splitting a term into several J levels. Since the Breit–Pauli Hamiltonian is not diagonal with respect to L and S, terms with different LS values may now interact. Thus MCHF + BP expansions are much longer than MCHF expansions. For these longer expansions, the Hamiltonian in the Breit–Pauli approximation must be re-evaluated to produce the

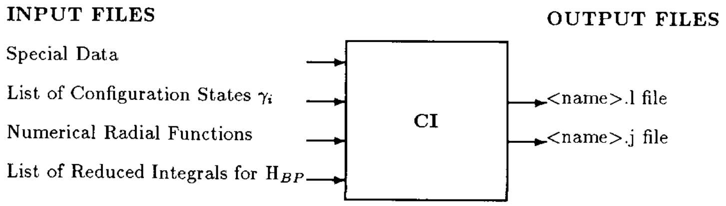

Fig. 4. Schematic for Breit–Pauli diagonalization.

BREIT list of reduced integrals – F^k, G^k, R^k, L, Z, N^k, V^k, and S^k, where the latter four represent the spin–orbit, spin–other–orbit, and spin–spin interactions.

The wave-function expansion within the MCHF + BP approximation is obtained from a CI calculation, depicted in figure 4. The ⟨name⟩.l file and ⟨name⟩.j file contain eigenvalues and eigenvectors of $H_{\mathrm{NR}} + H_{\text{non-fine}}$ and H_{BP}, respectively. The latter file may contain information for a series of J values. Once these files have been obtained, other properties such as transition probabilities can be evaluated.

4. Organization of the package

The package is organized as a set of programs, utilities, and libraries. Each program performs a specific atomic-structure calculation whereas the utilities are useful, short programs that assist in the processing and evaluation of results. The libraries contain routines that are used by more than one program. Table 1 shows the organization of the basic package. Only the short name is given in this table. When additional clarity is needed, particularly in related papers, the CI program, for example, may be referred to as MCHF_CI to emphasize that it is part of the MCHF atomic structure package. Similarly, the full name of a library may be given as MCHF_LIB_ANG, for example. Not mentioned in this table are the CPC routines for the CFP's (for p- and d-electrons) [16] and the NJGRAF recoupling package (Bar-Shalom and Klapisch [17]). A modified version of Robb's TENSOR program [18] is included in the ANG library.

With this package it is possible to perform energy-related calculations (ionization energies, electron affinities, term-energy separation, fine-structure splitting) and transition-probability data (wavelengths, line strengths, f-values) for E1, E2,... (Eλ in general) and M1, M2,... (Mλ in general) transitions from which lifetimes can be derived. Several effects may be included, namely correlation, relativistic shift effect, spin–orbit interaction, mass polarization. Programs for other atomic properties may be added in the future.

Table 1
Organization of the basic atomic structure package

	Angular	Radial	General
Programs	NONH BREIT MLTPOL	MCHF88 CI LSTR LSJTR	GENCL
Utilities		COMP LEVELS LINES	
Libraries	ANG	RAD	COM

5. Installation of the package

Though attempts have been made to make the system portable, there always are some system-related matters that should be investigated prior to compilation. The present package is a double-precision package. If single precision is required, appropriate changes will need to be made. It also assumes that file names may be specified in lower case and that a one- or three-character extension is allowed in a file name. The naming convention can readily be changed to whatever scheme is desired. The package has not been designed with vector processors in mind. Improving performance on vector machines requires the use of additional memory and degrades performance on conventional machines.

The above points are not likely to be important in many situations, but every installation should be preceded by a study of the following topics which describe system-dependent features of the package that may require change.

5.1. Unit numbers

Most files are referred to by name so that the user need not be aware of the FORTRAN unit number associated with the processing of the file. However, the following are exceptions:

UNIT = 5	Standard Input (usually a terminal or screen, 80 characters),
UNIT = 6	Standard Output (usually a terminal or screen, 80 characters),
UNIT = 0	Error or Prompt (usually a terminal or screen),
UNIT = 3	Printer Output (130 characters).

Usually, UNIT = 6 contains information important to the monitoring of a calculation in progress whereas UNIT = 3 contains information important as the completion of the calculation. UNIT = 0 is used for information that is needed even if UNIT = 6 is reassigned or redirected. (On the VAX, PRINT statements could be used for this purpose since they were not affected by a reassignment of UNIT = 6, but this feature is not supported by all operating systems). In a UNIX environment, UNIT = 0 should be the unit number of "Standard Error", but there seems to be no agreement on the unit number to be used for this purpose.

5.2. Dimensions

The package has been dimensioned for a modestly sized calculation and as long as format statements are not affected, arrays can readily be increased or decreased as desired. PARAMETER statements have been introduced to help maintain uniformity between the different packages. The following lists PARAMETER constants that appear in different programs, their values, and their role in the package:

NWD	30	maximum number of electrons or radial functions,
NOD	220	maximum number of grid points in the representation of a radial function,
NCD	100	maximum number of configuration states,
IDIM	500	maximum number of reduced integrals,
NCDIM	1000	maximum number of coefficients.

Some parameter constants are expressed in terms of these basic constants. For example, the maximum number of radial functions for a non-relativistic transition-probability calculation between an initial state and a final state could have a maximum of NWD2 = 2∗NWD radial functions, and NCD2 = 2∗NCD configuration states. A Breit–Pauli calculation generally has a longer expansion than a non-relativistic one: rather arbitrarily, the maximum allowed number is NCD2 so that a transition-probability calculation could have as many as NCD4 = 4∗NCD configuration states in the combined initial and final states.

Table 2
Parameters used to set the dimensions in various programs

Dimension for	Parameters used in						
	NONH	BREIT	MLTPOL	MCHF88	CI	LSTR	LSJTR
radial functions/electrons	NWD	NWD	NWD2	NWD	NWD	NWD2	NWD2
configuration states	NCD	NCD2	NCD4	NCD	NCD2	NCD2	NCD4
grid points	–	–	–	NOD	NOD	NOD	NOD
integrals	–	–	–	IDIM	IDIM2	–	–
integral coefficients	NCDIM	NCDIM4	–	NCDIM	–	–	–

Table 2 shows the dimensioning scheme that has been used. These dimensions can easily be changed if larger or smaller dimensions are required.

Dimension parameters that have not been defined in terms of PARAMETER constants are incorporated into the program with parentheses. Thus, a conditional statement such as

```
IF (I.LE. (20)) THEN
```

should be interpreted as an indication that "20" is associated with a dimension. Appropriate use of an editor will then change, not only the dimension, but also constants linked to that dimension.

5.3. Options

In SUN UNIX, the FORTRAN environment supports the use of command-line arguments. This is the most convenient way of specifying the names of files for a process when the names are variable and often results in easy-to-remember syntax. Thus GENCL BE1S.c would be the command for generating a configuration-state list for a case called BE1S. In this package it was decided to use default names instead, which limits flexibility somewhat but retains ease of use. Lines of code beginning with "CSUN" support the command-line argument option. By replacing these characters by blanks, the option may readily by re-installed.

The angular codes contain options for printing-out information that could assit in the debugging of a program containing the code. It was decided here that the debugging option would not be included explicitly, but neither has it been removed from the programs. This means that it can readily be re-installed. In some instances, lines of code needed only for debugging start with "CDBG"; in other instances, the variable controlling the printout of debugging information have been assigned values in the main program that prohibits the printing.

6. Summary of input data for the basic package

Each of the programs that are part of the package will be described in separate papers. Utilities will be described in a paper on MCHF libraries and utilities. As both an overview *and* a summary, we will describe here the SPECIAL data input in response to prompts, the other input files, and the results produced for each of the programs or utilities that define the basic package. Each program produces some output that is for information only and is not needed in future calculations. In an interactive session such output would appear on the screen, whereas in a batch environment it should be routed to a file. In the programs, FORTRAN unit number 6 is used for screen output (80 characters), unit number 3 for printed output (130 characters). We shall refer to these as UNIT6 and UNIT3, respectively. All other files are referred to by name.

In this description, a file whose name is fixed (not variable) will be specified by small capitals as, for example, TR.LST. Files that are variable will be specified by the file type. The latter will be in square brackets ([ ]) if optional. The types of files are:

1. cfile	List of configurations and their couplings that may contain information about expansion coefficients.
2. wfile	File containing the radial functions.
3. ifile	File containing information specifying the interaction matrix in terms of reduced integrals of either H_{NR} or H_{BP}, coefficients, and position in the matrix.
4. jfile	File containing eigenvectors from an *LSJ* Breit–Pauli CI calculation.
5. lfile	File containing eigenvectors of an interaction matrix, excluding fine-structure splitting (similar to a jfile, but only LS dependent).
6. mfile	File containing output from a multipole calculation.

A convenient file-naming convention is to use a name to specify a case and the first letter of the type as a one letter extension to indicate the type of file. Thus hels.c, hels.w, and hels.i could be the files for a calculation for the helium ground state. Some programs refer to such collections by "name" only. Since the name is variable, it will be indicated as "⟨name⟩".

Information on the use of the programs will be provided in the order in which they would be used, first for a non-relativistic transition-probability calculation and then in a Breit–Pauli allowed or forbidden transition-probability study.

6.1. GENCL

Program to generate configurations and their couplings either from a given list, given replacements, or some general rules. This program is best used in the interactive mode. At any point, an entry of 'b' means the user wishes to backup one line and repeat the input. A rudimentary help facility is available.

(i) Special data

blank or h (for help)
header
common closed subshells
reference set:
  config (1)
  config (2)   (maximum of 15 configurations)
  ⋮
  blank
active set (blank or comma delimited list of electrons, maximum of 15)
replacement list (terminated with blank line):
  general rules: el(2) = el′.el″
    el(2) = el′(2)
    $\mathrm{el}_1 \cdot \mathrm{el}_2 = \mathrm{el}'(2)$
    $\mathrm{el}_1 \cdot \mathrm{el}_2 = \mathrm{el}' \cdot \mathrm{el}''$
  where el, el_1, and el_2 are occupied orbitals of the reference set of configurations
or
  'sd' (for single and double replacements to a virtual set)
  virtual set (comma delimited list of electrons, maximum of 15)

final terms:

$$\left.\begin{array}{l} term_1 \\ term_2 \\ \vdots \\ blank \end{array}\right\}$$ maximum of 15 terms (blank only means *all* terms are requested); with the ’sd’ option, only one term or blank is accepted

(ii) Input files: none.

(ii) Output files: UNIT6 and CFG.INP.

6.2. NONH

Program to perform the angular integrations for the non-relativistic Hamiltonian, with the possibility of non-orthogonal orbitals, with at most two non-orthogonal pairs between each configuration. The list of integrals is sorted so that the energy can be expressed in terms of a unique list of integrals. The sorted list is written to INT.LST and defines the interaction matrix.

Parameters

new – number of *new* configurations (0 = ALL)
nzero – number of configurations (from beginning) defining the zero-order set (0 = ALL)

(i) Special data

printout (y/n)
all interactions (y/n)
  if ’n’ then
    new,nzero

(ii) Input file: CFG.INP.

(iii) Output file: INT.LST.

6.3. MCHF

Program to compute radial functions and expansion coefficients for an MCHF wave-function expansion.

Parameters

atom – label (at most 6 characters)
term – label (at most 6 characters)
Z – atomic number (real, with decimal point)
electrons – the electrons whose radial functions are to be varied; this may be specified in several ways:
  – all, none, some (outside closed subshells)
  – = ii (two digit integer) for last in list
  – comma delimited list

All other parameters as described in earlier publications [19].

(i) Special data

```
atom,term,Z
electrons
default electron parameters? (y/n)
    if 'n' then
        S, IND, METH, ACC in free-format (F,I,I,F) for each electron varied
default for NO, REL, STRONG (y/n)
    if 'n' then NO, REL, STRONG in free-format (I,L,L)
default parameters for other variables? (y/n)
    if 'n' then
        default for PRINT, CFGTOL, SCFTOL? (y/n)
            if 'n' then PRINT, CFGTOL, SCFTOL in format (L,F,F)
        default for NSCF, IC? (y/n)
            if 'n' then NSCF, IC in format (I,I)
        default for ACFD, LD, TRACE? (y/n)
            if 'n' then ACFD, LD, TRACE in format (F,L,L)
if iterations have not converged do you wish to continue? (y/n)
    if 'y' then NSCF,IC in format (I,I)
if iterations have converged, do you wish to continue along the isoelectronic
    sequence? (y/n)
    if 'y' then ATOM, TERM, Z, (ACC(i),i = 1,nwf)
        in format(2A6,F6.0,(18F3.0))
```

(ii) Input files: CFG.INP, INT.LST, and [WFN.INP].

(iii) Output files: UNIT6, UNIT3, CFG.OUT and WFN.OUT.

6.4. MLTPOL

Program to perform the angular integrations for one or more transition operators.

Parameters

initial – name of initial state
final – name of final state
type – E1, E2, ..., Eλ, M1, M2, ..., Mλ, with * designating exit

(i) Special data

```
initial
final
common orbitals for both states? (y/n)
    if 'n' then list of orbitals that are common (may be blank line) in format (18(1X,A3))
type
type

*   (to designate end of case)
```

(ii) Input files: ⟨initial⟩.c and ⟨final⟩.c.

(iii) Output files: UNIT6 and MLTPOL.LST.

6.5. LSTR

Program to compute E1 or E2 length and velocity transition data in the non-relativistic formalism.

Parameters

initial – name of initial state
final – name of final state
print – (y/n) controls intermediate printing
tol – contributions are printed if the length-form contribution is greater that tol in magnitude (all are printed if tol is 0.0)

(i) Special data

initial
final
print? (y/n)
  if 'y' then tol

(ii) Input files: ⟨initial⟩.c, ⟨initial⟩.w, ⟨final⟩.c, ⟨final⟩.w, and MLTPOL.LST.

(iii) Output file: UNIT6.

6.6. BREIT

Program to perform angular integrations for selected Breit–Pauli operators. Orbitals are assumed to be orthogonal.

Parameters

type – 0 for non-relativistic only
  1 for selected relativistic operators
  2 for both of the above
new – new configurations (from end of list)
nzero – subset of the first nzero configurations define the zero-order set

(i) Special data

type
full print-out? (y/n)
all relativistic operators? (y/n)
  if 'n' then spin–orbit? (y/n)
    spin–other–orbit? (y/n)
    spin–spin? (y/n)

```
all interactions? (y/n)
     if 'n' then new, nzero
     if type >  0 and nzero < ncfg then
                  relativistic interaction with all the zero-order block? (y/n)
                       if 'n' then define reference set (maximum is 20) in format(20I3)
                  diagonal relativistic corrections? (y/n)
     restricted 2-body interactions with zero-order set? (y/n)
```

(ii) Input file: CFG.INP.

(iii) Output file: INT.LST.

6.7. CI

Program to compute eigenvalues and eigenvectors of an interaction matrix, either in the non-relativistic or Breit–Pauli approximation. The mass-correction may also be included using either the gradient or Slater integral form.

Parameters

name – name to be used for defining files
nzero – size of zero-order matrix (0 = all)
neigv – approximate number of eigenvalues (defines an energy range)
max2j – maximum value of 2*J (0 if non-relativistic)
min2j – minimum value of 2*J (0 if non-relativistic)

(i) Special data

```
name
relativistic calculations? (y/n)
mass polarization included? (y/n)
     if 'y' gradient or Slater integral form? (g/s)
nzero (0 or blank = ALL)
neiv
max2j
min2j
matrix printed? (y/n)
```

(ii) Input files: ⟨name⟩.c, ⟨name⟩.w, and INT.LST.

(iii) Output files: UNIT6, ⟨name⟩.l, and ⟨name⟩j.

6.8. LSJTR

Program to compute length transition data in the relativistic Breit–Pauli formalism.

Parameters

initial – name of initial state
final – name of final state
tol – contributions are reported in TR.LSJ if *gf*-value is greater than tol (all are printed if tol = 0.0).
mass – mass of the nucleus for determining the Rydberg constant

(i) Special data

initial
final
intermediate printing? (y/n)
    if 'y' then tol
default Rydberg constant? (y/n)
    if 'n' then mass

(ii) Input files: MLTPOL.LST, ⟨initial⟩.c, ⟨initial⟩.w, ⟨initial⟩.j, ⟨final⟩.c, ⟨final⟩.w, and ⟨final⟩.j.

(iii) Output files: UNIT6 and TR.LSJ

6.9. COMP

Program to print wave-function expansions in increasing order of energy and decreasing magnitude of composition.

Parameters

tol – cut-off magnitude for printing contributions
name – name of files to be processed

(i) Special data

name
tol
case where
    case = 1: coefficients to be taken from cfile
    case = 2: coefficients to be taken from lfile
    case = 3: coefficients to be taken from jfile

(ii) Input files: ⟨name⟩.c, [⟨name⟩.l or ⟨name⟩.j].

(iii) Output files: UNIT6.

6.10. LEVELS

Program to sort the states in an lfile or jfile file in increasing order, report the theoretical total energies in atomic units and the energies relative to the lowest in cm^{-1}. Several such files may have been merged.

(i) Special data

name

(ii) Input file: ⟨name⟩.l or ⟨name⟩j.

(iii) Output file: UNIT6.

6.11. LINES

Program to sort and report the transition data in a number of ways.

Parameters

tol – transitions with line strength less than tol will be ignored
line list order criterion –
  1 – energy (cm^{-1})
  2 – wavelength in vacuum
  3 – wavelength in air
  4 – line strength
  5 – *gf*-value
  6 – transition probability

(i) Special data

tol
line-list-order criterion.

(ii) Input file: TR.LSJ.

(iii) Output file: UNIT6.

7. Spectroscopic notation and the cfile format

Many calculations are controlled or "driven" by the configuration list describing the wave-function expansion. This file usually starts as being CFG.INP for an MCHF calculation, the output of which is called CFG.OUT. The latter contains the same information as the former, but the total energy and the mixing coefficients have been added to the file so as to provide readable output. Spectroscopic notation is used throughout the package. Because the spectroscopic symbols, both for orbital angular momenta (l) and the total angular momenta (L), need to be converted to integers, the string of symbols is stored as data in the library routine, LVAL. From the position of the symbol in the string the $l(L)$-value can be determined. In the present implementation, the string contains the symbols for $l(L) \leq 10$. With the addition of symbols the range may be extended.

Table 3
An example of a configuration list for the $3s^23p$ ^2P state of aluminium showing the cfile format

```
  A1 2P
 1s  2s  2p
 3s( 2) 3pi( 1)
    1S0     2P1     2P0
3p2( 3)
    2P1
 3s( 1) 3p3( 1) 3d3( 1)
    2S1     2P1     2D1     1P0     2P0
 3s( 1) 3p4( 1) 3d4( 1)
    2S1     2P1     2D1     3P0     2P0
3p5( 1) 3d5( 2)
    2P1     1S0     2P0
3p5( 1) 3d5( 2)
    2P1     1D2     2P0
3p5( 1) 3d5( 2)
    2P1     3P2     2P0
```

Though a program, GENCL, has been written to generate a configuration file in the required format, it often is just as simple to enter the data using an editor. The format of this file is assumed to be as follows:

(i) A header: FORMAT(A72).
(ii) List of closed subshells, common to all configurations:
FORMAT(18(1X,A3)) and limited to a maximum of 18.
(iii) For each configuration:
  (a) the electron subshells and their occupation, followed optionally by the configuration weight: FORMAT(5(1X, A3, '(', I2, ')'), F);
  (b) the coupling of the occupied subshells: FORMAT(9(5X, A3)).

An example of a configuration list for the $3s^23p$ ^2P state of aluminium using non-orthogonal orbitals is given in table 3. The dimensions of the present implementation are such that there may be at most five subshells outside the closed subshells common to all configurations. Clearly, these subshells may also include some closed subshells, but the angular programs will execute more efficiently if common closed subshells are removed. In all cases, contributions from these closed subshells can be treated by formula. At the same time, the list of reduced integrals can be kept to a minimum. For example, the Z-integrals in the Breit–Pauli approximation contain many contributions from the core and hence would grow rapidly, as the size of the core increases, if such a scheme were not adopted. For the non-relativistic Hamiltonian, Slater introduced the notion of an "average" energy which could readily be derived from a formula so that the only data needed by a program was the deviations from the average. These deviations only involved open subshells. However, this notion was difficult to generalize to other atomic properties and so a somewhat simpler notion has been introduced, namely one where all contributions involving the core are omitted from the list of reduced integrals. The efficient organization of the reduced integral list for both Hamiltonians will be discussed in section 8.

The configuration consists of subshells, identified by their electron label, and their occupation. The electron label may contain three characters – the n symbol, the l symbol, and (optionally) a set indicator. The n symbol is the usual digit if $n = 1, \ldots, 9$, but for higher n, $n =$ CHAR(n + ICHAR('0')), in FORTRAN notation. The n character is the first non-blank character in the electron label field. The l symbol must follow immediately and is the usual spectroscopic symbol. It may be either upper case or lower case. The set indicator is used if non-orthogonal orbitals are desired and must then be the third

Table 4
A list of allowed terms (multiplicity, symmetry, and seniority) for partially filled subshells, $l(q)$, where q is either an occupation number or the number of holes. All filled subshells have a term of 1SO. For $l > 3$, the terms allowed in the package are restricted as indicated by the last two lines

CONFIGURATION	TERMS (MULTIPLICITY, SYMMETRY, AND SENIORITY)
s(1)	2S1
s(2)	1SO
p(1)	2P1
p(2)	1SO 1D2 3P2
p(3)	2P1 2D3 4S3
d(1)	2D1
d(2)	1SO 1D2 1G2 3P2 3F2
d(3)	2D1 2P3 2D3 2F3 2G3 2H3 4P3 4F3
d(4)	1SO 1D2 1G2 3P2 3F2 1S4 1D4 1F4 1G4 1I4 3P4 3D4 3F4 3G4 3H4 5D4
d(5)	2D1 2P3 2D3 2F3 2G3 2H3 4P3 4F3 2S5 2D5 2F5 2G5 2I5 4D5 4G5 6S5
f(1)	2F1
f(2)	1SO 1D2 1G2 1I2 3P2 3F2 3H2
l(1)	2L1
l(2)	1SO 1D2 1G2 1I2 3P2 3F2 3H2

character in the label field. The algebra for the evaluation of matrix elements with non-orthogonal orbitals is complex, but for the non-relativistic Hamiltonian a limited amount of non-orthogonality can be dealt with in the present codes. In such cases, orbitals with the same set indicator as well as orbitals in the same configuration are orthogonal; all orbitals are orthogonal to an orbital with the same angular orbital momentum for which the set indicator is omitted (or blank). In the non-orthogonal approximation the configuration list may also include:

(iv) An * to designate the end of the configuration list.
(v) Additional orthogonality constraints, one per line: FORMAT(2(1X, A3)).
(vi) An * to signal the end of the orthogonality list.

The coupling information for the subshells consists of an integer for the multiplicity, $(2S+1)$, the spectroscopic symbol for L, and the seniority. These three values define the "term" for a subshell. Table 4. lists the allowed terms for open subshells according to the number of electrons or holes in the subshell. For $l > 3$, one electron or one hole are allowed by the program; two electrons (but not two holes) can also be accommodated but the terms must then be restricted to those allowed for $l = 3$. The subshells are then coupled from left to right as required by the angular programs. Though information may be specified either in lower case or upper case, the following conventions have been found helpful:

- electron labels are lower case,
- electron labels ae right-justified,
- coupling information is in upper case,
- the coupling of subshells results in terms with seniority = 0.

8. Ifile format

The file describing the interaction matrix either for the non-relativistic Hamiltonian or the Breit–Pauli Hamiltonian, is written in "ifile" format. By default, this file is usually named INT.LST for "interaction list". Before defining the format of this file, it is helpful to consider some of the factors that influence the performance of programs using this file.

Two programs use INT.LST – MCHF88 and CI. The former applies only to the non-relativistic Hamiltonian, but non-orthogonal orbitals may be present, whereas the latter may be either the non-relativistic or the relativistic Breit–Pauli approximation. For the Breit–Pauli approximation, only an orthonormal basis can be dealt with at this time.

In the MCHF88 program, the variation of the integrals with respect to allowed perturbations define the set of integro-differential equations that need to be solved. Clearly, if the same integral occurs many times, it is most efficient if the variation is computed only once. For the MCHF program, it is also helpful to know the symmetry type of an integral. For example, F^k and G^k integrals are both special cases of R^k (see table 5), but only F^k integrals contribute to the direct potential, a function needed to solve the radial equation for an orbital [21]. In a CI calculation, the radial integrals are evaluated, multiplied by coefficients (and possibly overlap factors) and added to the appropriate matrix element. Again, if the same integral occurs many times, it would be efficient to evaluate that integral only once. In the case of a relativistic calculation, the same integral list is used to generate the different J-dependent interaction matrices. It can be shown that, in the LSJ scheme, an interaction matrix element can be factored into a J-independent part times the Racah coefficient, $W(L,S,L',S';kJ)$, where $k=1$ except for the spin–spin operator where $k=2$. Integrals arising from this operator must be separated from similar integrals arising from other operators. Thus, efficient processing of this list requires that

1. integrals be separated into types;
2. each integral occurs only once in the list.

Such an organization can be achieved by sorting the data associated with each type of integral.

The reduced radial integrals that may occur for various operators are summarized in table 5. Not all need be included in the INT.LST. Contributions that arise from the common core are included by interpreting the L-operator as an L_c-operator that includes core contributions with related coefficients. The sum on α is over the closed subshells in the common core. The coefficients $\mathrm{cb}_{l_1,l_a,k}$ represent the average exchange interaction between an electron i and an electron a in a closed subshell. Similarly, the nuclear spin–orbit integral, Z, has been redefined to include what can be interpreted as the effect of the screening of the nucleus by the common core. The coefficients $\mathrm{bwm}_{l_i,l_a,k}$, $\mathrm{bwn}_{l_i,l_a,k}$, and $\mathrm{bwv}_{l_i,l_a,k}$ are coefficients first published by Blume and Watson [22] that represent the contribution to the spin–orbit parameter from a closed subshell.

Glass and Hibbert [15] have shown that the coefficients of the relativistic shift operators are related to other integrals. These effects can be included in a calculation through the optional redefinition of the L_c and R^k (including F^k and G^k) integrals as shown in table 6.

Each integral has certain symmetry properties. To identify a minimum number of different integrals, each integral is expressed in a canonical form that can be derived from symmetry relations. The symmetry relations and the canonical form are shown in table 7. In the case of the R^k integrals, not all symmetries are used. A number of atomic properties can be derived from the same coefficient data. One such property is the mass-polarization correction that accounts for the specific mass shift in different isotopes of the same atom [20]. In this case, the tensorial operator is similar to the electrostatic operator, but the reduced integral is different. In order to maintain the proper phase shift for this operator, the symmetry relations must be restricted to interchanges of the two coordinates or an interchange of the electrons on the left with the electrons on the right. Then $R^k(ii;\ jj)$ cannot be considered the same as $R^k(ij;\ ji)$, and for this

Table 5
Radial integrals for different operators. Summations over a are over all closed subshells of a common core

(a) Non-relativistic Hamiltonian

$$F^k(i;\ j) \equiv R^k(i,\ j;\ i,\ j)$$

$$G^k(i;\ j) \equiv R^k(i,\ j;\ j,\ i)$$

$$R^k(i,\ j;\ i'j') = \int_0^\infty \int_0^\infty P_i(r_i) P_j(r_2) \frac{r_<^k}{r_>^{k+1}} P_{i'}(r_1) P_{j'}(r_2)\, \mathrm{d}r_1\, \mathrm{d}r_2$$

$$L_c(i,\ j) = \langle i \,|\, \frac{\mathrm{d}^2}{\mathrm{d}r^2} + \frac{2}{r}(Z - Y_c(r) - X_c(r)) - \frac{l(l+1)}{r^2} \,|\, j \rangle,$$

$$\text{where} \langle i \,|\, Y_c(r) \,|\, j \rangle = \sum_a 2(2l_a + 1) R^0(i,\ a;\ j,\ a)$$

$$\langle i \,|\, X_c(r) \,|\, j \rangle = -\sum_a 2(2l_a + 1) \sum_{k = |l_i - l_a|}^{l_i + l_a} \mathrm{cb}_{l_i, l_a, k} R^k(i,\ a;\ a,\ j)$$

$$O_1^k(i,\ j) = \langle i \,|\, j \rangle^k$$

$$O_2^K(i,\ i';\ j,\ j') = \langle i \,|\, i' \rangle^{k_1} \langle j \,|\, j' \rangle^{k_2}, \quad \text{where} \quad K = 64k_1 + k_2$$

(b) Breit – Pauli fine structure

$$Z(i,\ j) = \frac{\alpha^2}{4} \langle i \,|\, r^{-3} \,|\, j \rangle - \sum_a 2(2l_a + 1) N^0(i,\ a;\ j,\ a) + \sum_a \sum_{k = \mathrm{kmin}}^{l_i + l_a} \{ \mathrm{bwm}_{l_i, l_a, k} N^{k-2}(i,\ a;\ a,\ j)$$

$$+ \mathrm{bwn}_{l_i, l_a, k} N^k(a,\ i;\ j,\ a) + \mathrm{bwv}_{l_i, l_a, k} (V^{k-1}(i,\ a;\ a,\ j) - V^{k-1}(a,\ i;\ j\ a))\},$$

$$\text{where} \quad \mathrm{kmin} = \begin{cases} 2, & l_i = l_a \\ |l_i - l_a|, & l_i \neq l_a \end{cases}$$

$$N^k(i,\ j;\ i',\ j') = \frac{\alpha^2}{4} \int_0^\infty \int_0^\infty P_i(r_1) P_j(r_2) \frac{r_2^k}{r_1^{k+3}} e(r_1 - r_2) P_{i'}(r_1) P_{j'}(r_2)\, \mathrm{d}r_1\, \mathrm{d}r_2, \quad \text{where} \quad e(x) = \begin{cases} 1, & x > 0 \\ 0, & x \le 0 \end{cases}$$

$$V^k(i,\ j;\ i',\ j') = \frac{\alpha^2}{4} \int_0^\infty \int_0^\infty P_i(r_1) P_j(r_2) \frac{r_<^k}{r_>^{k+3}} r_2 \frac{\partial}{\partial r_1} P_{i'}(r_1) P_{j'}(r_2)\, \mathrm{d}r_1\, \mathrm{d}r_2$$

$$S^k(i,\ j;\ i',\ j') \equiv N^k(i,\ j:\ i',\ j')$$

(c) Breit – Pauli non-fine structure

mass correction	$C(i,\ j) = \frac{\alpha^2}{4} \int_0^\infty \left(\frac{\mathrm{d}^2}{\mathrm{d}r^2} - \frac{l_i(l_i + 1)}{r^2} \right) P_i(r) \left(\frac{\mathrm{d}^2}{\mathrm{d}r^2} - \frac{l_j(l_j + 1)}{r^2} \right) P_j(r)\, \mathrm{d}r$
one body Darwin	$D(i,\ j) = \frac{\alpha^2}{4} Z \left(\frac{p_i(r)}{r} \right)_{r=0} \left(\frac{P_j(r)}{r} \right)_{r=0}$
two-body Darwin	$X^k(i,\ j;\ i',\ j') = \frac{\alpha^2}{4} \int_0^\infty \int_0^\infty P_i(r_1) P_j(r_2) \frac{\delta(r_1 - r_2)}{r_1^2} P_{i'}(r_1) P_{j'}(r_2)\, \mathrm{d}r_1\, \mathrm{d}_2$
spin–spin contact	$X^k(i,\ j;\ i',\ j')$ same as two-body Darwin

reason, such interactions are not expressed as $G^k(i;\ j)$. Once the integrals have been expressed in a canonical form, they can be sorted and a unique list of integrals determined.

Excluding overlap integrals, at most four radial functions and a k value enter into the definition of an integral. The R^k integrals with possible overlap factors represent the general cases. Figure 5 shows a data

Table 6
Redefinition of non-relativistic operators to include the relativistic shift effect

Non-relativistic	With relativistic shift
$L_c(i, j)$	$L_c(i, j) - C(i, j) - \delta_{l_i 0} d(i, j)$
$R^k(i, j; i', j')$	$R^k(i, j; i', j') + (2k+1) X^k(i, j, i', j')$

Table 7
Canonical form of integrals

Type of integral	Allowed interchanges	Canonical form
F^k, G^k, L, Z	$ij \equiv ji$	$i \le j$
R^k	$iji'j' \equiv i'j'ij \equiv jij'i'$	$i \le \min(j, i', j')$
N^k, S^k	$iji'j' \equiv jii'j' \equiv i'jij'$	$i \le i'$ and $j \le j'$
V^k	$iji'j' \equiv ij'i'j$	$j \le j'$

structure for representing the R^k-integral list. With data defining the integral is associated a "C-pointer" (CPTR) that points to the last element in the coefficient data list associated with this integral. The first element is either 1 (if the integral is the first) or the CPTR value of the preceding integral plus 1. The data associated with an integral is a set of coefficients that multiply the integral and a position in the interaction matrix. Associated with each coefficient is an "O-pointer" (OPTR) that points to an overlap

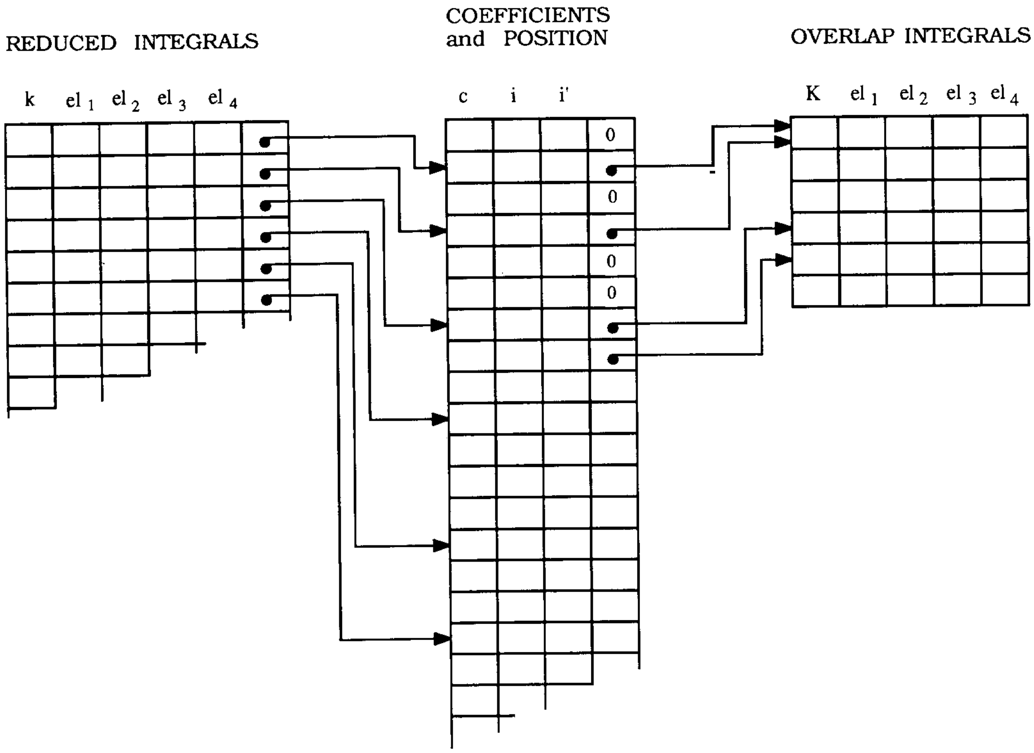

Fig. 5. Data structure showing the organization of data associated with R^k-integrals and their contribution to the interaction matrix or energy expression.

Table 8
Format of data in ifile. Sym refers to the one-character symbol designation of the type of integral; CPTR is the pointer to the coefficient data; c is a coefficient; i' and i specify the positions in the interaction matrix; OPTR is a pointer to an overlap – O_1 if OPTR > 0, O_2 if OPTR < 0, and neither if OPTR = 0 (see fig. 5)

Type	Variables	FORMAT
(a) Integrals		
F^k, G^k	Sym, k, el$_1$, el$_2$, CPTR	`1X, A1, I2,'(', A3, ',', A3, ')', I5`
L, Z	Sym, el$_1$, el$_2$, CPTR	`1X, A1, 2X,'(', A3, ',', A3, ')', I5`
O_l	'O', k$_1$, el$_1$, el$_2$	`1X, A1, I2, '<', A3, ' | ', A3, '>'`
O_2	'O', K, el$_1$, el$_2$, el$_3$, el$_4$	`1X, A1, 2(I2, '<', A3, ' | ', A3, '>')`
R^k, N^k, V^k, S^k	Sym, k, el$_1$, el$_2$, el$_3$, el$_4$	`1X, A1, I2, '(', 2A3, ',', 2A3, ')', I5`
end-of-list		`1X, '*'`
(b) Data associated with an integral		
F^k, G^k, V^k, S^k	c, Sym, i', i	`F14.8, A1, 2I3`
R^k, L	c, Sym, i', i, OPTR	`F14.8, A1, 3I3`
end-of-list		`14X, '*'`

factor that multiplies that particular term. A negative OPTR implies that the overlap is of type 2 with two factors, a positive OPTR that the overlap is of type 1 with only one factor, a zero value for OPTR that no overlap factors are present.

In the CI program, it is efficient if the O-integrals appear in the list of integrals before they are needed. The F^k and G^k integrals never have overlap integrals since they arise from diagonal interactions. The INT.LST is produced as a list containing data for the different types of integrals in the order,

$$F^k, G^k, O_1, O_2, R^k, L, Z, N^k, V^k, S^k.$$

For each type of integral, the list of integrals is given, terminated with an asterisk, and followed by the coefficients for the type of integral, also terminated with an asterisk. Overlap integrals differ in that they have no associated data.

The FORMAT of the variables that define the different types of integrals and their associated data is specified in table 8. Several facts should be noted immediately. In order that this file be readable, the coefficients have been written in a fixed format with eight significant decimal places. It may be desirable to increase this for improved accuracy. The value of k is written in fixed formats which effectively limits l to $l \leq 49$. The `I3` format for OPTR is not likely to be a limitation, but it restricts the number of type 2 overlaps to less than 100, though the number of type 1 overlaps (for which the value of OPTR is positive) may be as large as 999. The same `I3` format also limits the number of configuration states to a maximum of 999. The `I5` format for CPTR limits the number of coefficients for a given type of integral to at most 99 999. When associated parameter statements are increased beyond these values, then the format statements will also have to be changed or the file converted to unformatted, binary form.

Examples of INT.LST files will be included in the description of NONH and BREIT.

9. Lfile, jfile, and wfile formats

The lfile and jfile formats are both the same. They are text files in a fairly easy to read format that represent a summary of output from a CI calculation, namely the eigenvalues and eigenvectors for a series of interaction matrices. The lfile is diagonal in LS, the jfile is diagonal in J.

The first line is a header line, namely

```
'Atom = ', atom, 'Z = ', Z, 'NEL = ', nel, 'NCFG = ',
  ncfg in FORMAT(2X,A6,A,F5.1,A,I3,A,I3)
```

where Z is the nuclear charge, NEL the number of electrons, and NCFG the number of configurations.

Then

```
For each J value
   '2*J = ', 2*J, 'NUMBER = ', number in FORMAT(//A8,I4,2X,A8,I4)
   For i = 1, number
      JMAX, EIGVAL, CONFIG in FORMAT(I6, F16.8,2X,A)
      EIGVEC(IV,I),IV = 1,NCFG in FORMAT(7(10.7))
```

where NUMBER is the number of eigenstates for the specified *J* value, JMAX is the position in the eigenvector of the maximum component, and CONFIG is the configuration state in a packed format that includes coupling information. Finally, since sets obtained from different calculations may be combined so that level information can be determined using the LEVELS program, the data for a particular calculation is terminated with '*** ' in the first three columns.

The wfile format is written as an unformatted file, both to retain machine accuracy and reduce disk usage. It consists of information defining a set of radial functions. Consistent with the definition of a set, the order of the radial function information is not important – it may contain more radial functions than are needed but, of course, the set must contain all the radial functions that are needed for a calculation. On UNIX or VAX systems, the wfiles may be concatenated or one wfile appended to another.

The information for each radial function includes the following:

ATOM, TERM	labels for a case (type CHARACTER*6);
EL	electron label (type CHARACTER*3);
M	number of tabulated points in the range of the function (integer);
Z	atomic number (double precision);
ET	diagonal energy parameter (double precision);
EKI	kinetic energy (double precision);
AZI	AZI $\rightarrow P(r)/r^{l+1}$, $r \rightarrow 0$ (double precision);
P(j)	$P(r_j)/\sqrt{r_j}$, $j = 1$, M (double precision).

Because the data types must match exactly when reading unformatted files, the types have been specified explicitly. Note that the atomic number need not have an integer value. In this package, the nuclear charge is a double-precision variable, which can often be used to advantage when convergence problems are encountered in MCHF88.

10. Example of a transition-probability calculation

Finally, as Test Run Output, we show a transition probability calculation, performed in the interactive mode under the UNIX operating system, with output to UNIT = 6 redirected to a file. For the sake of brevity and clarity, the output to the screen (UNIT = 0) has been edited. Command lines which invoke a procedure (either a program from the MCHF package or a system command) start with the symbols " > > " designating input. Data (UNIT = 5) supplied in response to a prompt is indicated by " > ". The calculation is for the $3s^23p\ ^2P \rightarrow 3s^23d\ ^2D$ transition of aluminum and can be divided into four phases – the calculation for the initial state, the final state, the transition data, the energy-level data. Command lines are numbers to indicate the phase as well the number within a phase. In addition, each command line has had a short description added to it summarizing the calculation that is performed. Towards the end, where the output to UNIT = 6 is brief, it has been included to provide an indication of the type of results obtained by such a simple calculation. The example uses the previously published Hartree–Fock [23] program for obtaining initial estimates for the 1s, 2s, 2p, and 3s orbitals from a calculation for Al^+. This program is not part of the present package but is a very useful supplement.

Several points should be noted. The files CFG.INP, WFN.INP are intermediate files that serve as initial estimates of mixing coefficients and radial functions, respectively, for MCHF. The files produced by MCHF, namely CFG.OUT, WFN.OUT, may not be final files. This is demonstrated in calculation (2.3) where the MCHF calculation did not converge to the desired accuracy. The CFG.OUT and WFN.OUT files are then recycled as CFG.INP and WFN.INP, respectively, and the MCHF iterations repeated. Also, in a transition-probability calculation, the common core should be the same in both the initial and final state since the evaluation of the transition-probability matrix elements can deal with only a limited amount of non-orthogonality and the transition program assumes that there will be at most two overlap integrals. Thus the radial functions for the ^2P calculations were used as initial estimates for the ^2D one, and only the outer functions varied. The calculations were restricted to the complex and only configurations consisting of $n = 3$ subshells were included in the expansion. In a non-relativistic calculation, a comparison of the length and velocity form of the f-value may be used as an indication of accuracy. In this case, they differ by almost a factor two. More accurate results have been obtained through the use of non-orthogonal orbitals [24] and with the inclusion of configurations containing f-electrons. Indeed, the energy associated with the configuration list of table 3, assuming the same core as in the present calculation was -241.924616 a.u., or 917.4 cm^{-1} lower in energy. This demonstrates the versatility of the non-orthogonal approximation, though at the expense of more complicated Racah algebra and the computation of more radial functions.

Transitions for the multiplet have also been determined. In this case, the mixing of different LS terms is likely to be small and so the Breit–Pauli calculation used the same configuration-state expansion as the non-relativistic calculation. This is not usually the case. The LEVELS calculations shows the fine-structure splitting for the $^2P_{1/2}$–$^2P_{3/2}$ of 106.7 cm^{-1} to be in reasonable agreement with the observed value of 112.0 cm^{-1}. The complete output from this calculation will be included in the paper describing the programs for LS and LSJ transitions [25].

Acknowledgements

Special thanks are due to many of the researchers who have collaborated with me over the years: K.M.S. Saxena who first proposed the use of spectroscopic notation, Alan Hibbert who helped interface his WEIGHTS program to MCHF, both R. Glass and A. Hibbert for help in getting the BREIT program to work, Michel Godefroid for his interest in the transition probability part of the package, and Lidia Smentek-Mielczarek for her work on the specific-mass shift. But thanks are also due to users such as Jørgen Hansen, Hari P. Saha, Tomas Brage, Claes-Göran Wahlström, and Eleanore Trefftz who, in using some of the programs, showed the scope and limitations of the package, pointing the way towards new directions and improvements.

This research has been supported by the US Department of Energy, Office of Basic Energy Sciences.

References

[1] C. Froese Fischer, The Hartree–Fock Method for Atoms: A Numerical Approach (J. Wiley, New York, 1977).
[2] C. Froese Fischer, J. Comput. Phys. 27 (1978) 221.

[3] C. Froese Fischer, Can. J. Phys. 53 (1975) 184, 338.
[4] T. Brage, C. Froese Fischer, J. Carlsson and C.-G. Wahlström, Phys. Rev. A 35 (1987) 1113.
[5] C. Froese Fischer J.B. Jolanta and S.H. Vosko, Phys. Rev. Lett. 59 (1987) 2263.
[6] C. Froese Fischer, J. Phys. B 16 (1983) 157.
[7] K.-N. Huang, Y.-K. Kim, K.T. Cheng and J.P. Desclaux, Phys. Rev. Lett. 48 (1982) 1245.
[8] C. Froese Fischer and H.P. Saha, Phys. Scr. 32 (1985) 181.
[9] M. Godefroid and C. Froese Fischer, J. Phys. B 17 (1984) 681.
[10] C. Froese Fischer and H.P. Saha, Phys. Rev. A 28 (1983) 3169.
[11] W. Eissner, M. Jones and H. Nussbaumer, Comput. Phys. Commun. 8 (1974) 270.
[12] A. Hibbert, Comput. Phys. Commun. 9 (1975) 141.
[13] R.D. Cowan, The Theory of Atomic Structure and Spectra (University of California, Berkeley, 1981).
[14] C. Froese Fischer, Comput. Phys. Commun. 64 (1991) 431, this issue.
[15] R. Glass and A. Hibbert, Comput. Phys. Commun. 16 (1978) 19.
[16] D.C.S. Allison, Comput. Phys. Commun. 1 (1970) 15.
[17] A. Bar-Shalom and M. Klapisch, Comput. Phys. Commun. 50 (1988) 375.
[18] W.D. Robb, Comput. Phys. Commun. 6 (1973) 132.
[19] C. Froese Fischer, Comput. Phys. Commun. 1 (1969) 151; 4 (1972) 107; 14 (1978) 145.
[20] C. Froese Fischer and L. Smentek-Mielczarek, J. Phys. B 16 (1983) 3479.
[21] C. Froese Fischer, Comput. Phys. Rep. 3 (1986) 273.
[22] M. Blume and R.E. Watson, Proc. Roy. Soc. London A 270 (1962) 127.
[23] C. Froese Fischer, Comput. Phys. Commun. 43 (1987) 355.
[24] C. Froese Fischer, Can. J. Phys. 54 (1976) 740.
[25] C. Froese Fischer and M.R. Godefroid, Comput. Phys. Commun. 64 (1991) 501, this issue.

TEST RUN OUTPUT

```
>> hf                      =======>  0) Get initial estimates for core
 Enter ATOM,TERM,Z
 Examples: O,3P,8. or Oxygen,AV,8.
> Al,1S,13.
 List the CLOSED shells in the fields indicated (blank line if none)
 ... ... ... ... ... ... ... ... etc.
> 1s  2s  2p  3s
 Enter electrons outside CLOSED shells (blank line if none)
 Example: 2s(1)2p(3)
>
 There are   4 orbitals as follows:
   1s  2s  2p  3s

 Orbitals to be varied: ALL/NONE/=i (last i)/comma delimited list/H
>all
 Default electron parameters ? (Y/N/H)
>y
 Default values for remaining parameters? (Y/N/H)
>y
 Additional parameters ? (Y/N/H)
>n
 Do you wish to continue along the sequence ?
>n
mv wfn.out wfn.inp         =======>0.1) HF results are initial estimates

>> gencl                   =======>1.1) Generate 2P configuration list
>                           ===> Blank for no help
             Header  ?
>  Al 2P
      Closed Shells  ?
>  1s  2s  2p
      Reference Set  ?
>3s(2)3p(1)
                2  ?        ===> Prompt for second configuration
>                           ===> blank, terminating the reference set
         Active Set  ?
>3s,3p,3d
       Replacements  ?
>
        Final Terms  ?
>2P
                2  ?        ===> Prompt for second Term
>                           ===> blank, terminating the list of Terms

>> nonh                    =======>1.2) Perform angular integrations
  FULL PRINT-OUT ? (Y/N)
>n
  ALL INTERACTIONS ? (Y/N)
>y
>> mchf                    =======>1.3) Determine all radial functions
  ATOM, TERM, Z in FORMAT(A,A,F) :
>Al,2P,13.
  There are   6 orbitals as follows:
    1s  2s  2p  3s  3p  3d
  Enter orbitals to be varied: (ALL,NONE,SOME,NIT=,comma delimited list)
>all
  Default electron parameters ? (Y/N)
>y
```

```
  Default values (NO,REL,STRONG) ? (Y/N)
>y
  Default values for other parameters ? (Y/N)
>y
  Do you wish to continue along the sequence ?
>n
>> mv cfg.out al2p.c       =======>1.4) Save cfile for al2p
>> mv wfn.out al2p.w       =======>1.5) Save wfile for al2p

>> breit                   =======>1.6) Determine BP interaction matrix
  Indicate the type of calculation
  0 => non-relativistic Hamiltonian only;
  1 => one or more relativistic operators only;
  2 => non-relativistic operators and selected relativistic:
>2
  Is full print-out requested? (Y/N)
>n
  All relativistic operators ? (Y/N)
>y
  All Interactions? (Y/N):
>y

>> ci                      =======>1.7 Determine LSJ eigenvectors
    Name of State
>al2p
    Is this a relativistic calculation ? (Y/N) :
>y
    Is mass-polarization to be included ? (Y/N) :
>n
 The size of the matrix is     7
  Enter the approximate number of eigenvalues required
>1
    Maximum and minimum values of 2*J ?
>3,1
    Do you want the matrix printed? (Y or N)
>n
>> gencl                   =======>2.1) Generate 2D configuration list

             Header  ?
>  Al 2D
      Closed Shells  ?
>  1s  2s  2p
      Reference Set  ?
>3s(2)3d(1)
                2  ?
>
         Active Set  ?
>3s,3p,3d
       Replacements  ?
>
        Final Terms  ?
>2D
                2  ?
>

>> nonh                    =======>2.2) Perform angular integrations
  FULL PRINT-OUT ? (Y/N)
>n
> ALL INTERACTIONS ? (Y/N)
 y
```

```
>> mchf                    =======>2.3) Determine 3s,3p,3d functions
  ATOM, TERM, Z in FORMAT(A,A,F) :
>al,2D,13.
  There are   6 orbitals as follows:
    1s  2s  2p  3s  3d  3p
  Enter orbitals to be varied: (ALL,NONE,SOME,NIT=,comma delimited list)
>=3
  Default electron parameters ? (Y/N)
>y
  Default values (NO,REL,STRONG) ? (Y/N)
>y
  Default values for other parameters ? (Y/N)
>y
  SCF ITERATIONS HAVE CONVERGED TO THE ABOVE ACCURACY
    Do you wish to continue ? (Y/N)
>n

>> mv wfn.inp cfg.inp
>> mv wfn.out wfn.inp      =======>2.4) Recycle for better convergence
>> mv cfg.out cfg.inp
>> mchf                    =======>2.5) Improve radial functions
  ATOM, TERM, Z in FORMAT(A,A,F) :
>al,2D,13.
  There are   6 orbitals as follows:
    1s  2s  2p  3s  3d  3p
  Enter orbitals to be varied: (ALL,NONE,SOME,NIT=,comma delimited list)
>=3
  Default electron parameters ? (Y/N)
>y
  Default values (NO,REL,STRONG) ? (Y/N)
>y
  Default values for other parameters ? (Y/N)
>y
  Do you wish to continue along the sequence ?
>n
>> mv wfn.out al2d.w       =======>2.7) Save results for al2d
>> mv cfg.out al2d.c

>> breit                   =======>2.8) Determine BP interaction matrix
  Indicate the type of calculation
  0 => non-relativistic Hamiltonian only;
  1 => one or more relativistic operators only;
  2 => non-relativistic operators and selected relativistic:
>2
  Is full print-out requested? (Y/N)
>n
  All relativistic operators ? (Y/N)
>y
  All Interactions? (Y/N):
>y
>> ci                      =======>2.9) Determine LSJ eigenvectors
    Name of State
>al2d
    Is this a relativistic calculation ? (Y/N) :
>y
    Is mass-polarization to be included ? (Y/N) :
>n
 The size of the matrix is     8
  Enter the approximate number of eigenvalues required
>1
```

```
    Maximum and minimum values of 2*J ?
>5,3
    Do you want the matrix printed? (Y or N)
>n

>> mltpol                  =======>3.0) Compute transition matrix elements
   Name of Initial State
>al2p
   Name of Final State
>al2d
  THERE ARE  3 INITIAL STATE ORBITALS AS FOLLOWS:
    3s  3p  3d
  THERE ARE  3 FINAL STATE ORBITALS AS FOLLOWS:
    3s  3d  3p
  Initial & final state orbitals an orthonormal set ? (Y/N)
>n
  List common orbitals, terminating with a blank orbital.
  Upper and lower case characters must match.
  Fixed format (18(1X,A3)) as indicated below:
  AAA AAA AAA AAA AAA AAA AAA .... etc (up to 18/line)
>
    Type of transition ? (E1, E2, M1, M2, .. or *)
>E1
    Type of transition ? (E1, E2, M1, M2, .. or *)
>*
>> lstr                    =======>3.1) Non-relativistic gf-values
    Name of Initial State
>al2p
    Name of Final State
>al2d
    INTERMEDIATE PRINTING (Y OR N) ?
>y
     TOLERANCE FOR PRINTING ?
>0.001
  Default Rydberg constant (Y or N) ?
>y

>>type UNIT6.out           =======>3.2) Display output

                    ============================
                      LS - T R A N S I T I O N
                    ============================
   Initial State
   -------------
   Al    2P       -241.9204360

   1    0.9604447   3s(2).3p_2P
   2    0.1902863   3p(3)_2P1
   3    0.1542743   3s.3p_1P.3d_2P
   4   -0.1148907   3s.3p_3P.3d_2P
   5   -0.0545261   3p.3d(2)1S0_2P
   6   -0.0227076   3p.3d(2)1D2_2P
   7   -0.0291200   3p.3d(2)3P2_2P
```

```
  Final   State
  -------------
  Al    2D        -241.7720190

  1    0.9581457   3s(2).3d_2D
  2    0.1465009   3s.3p(2)1D2_2D
  3    0.2454186   3p(2)1S0.3d_2D
  4   -0.0035531   3p(2)1D2.3d_2D
  5    0.0149756   3p(2)3P2.3d_2D
  6   -0.0047824   3s.3d(2)1D2_2D
  7   -0.0020545   3d(3)_2D1
  8   -0.0000980   3d(3)_2D3

Default Rydberg constant (Y or N) ?

  Al       Z = 13.   Angular multiplicities =  3  5
                     Spin    multiplicities =  2  2
Orbitals
Initial:   6;  1s  2s  2p  3s  3p  3d
Final  :   6;  1s  2s  2p  3s  3d  3p

            Transition Integrals                      Length     Velocity

    1->  1:   3p-> 3d  < 3s| 3s>^ 2  <    |   >^ 0      2.8876017   0.4979971
    1->  2:   3s-> 3p  < 3s| 3s>^ 1  < 3p| 3p>^ 1     -0.4533017  -0.0780489
    2->  2:   3p-> 3s  < 3p| 3p>^ 2  <    |   >^ 0      0.0617022  -0.0128295
    2->  3:   3p-> 3d  < 3p| 3p>^ 2  <    |   >^ 0      0.1133674   0.0195514
    2->  4:   3p-> 3d  < 3p| 3p>^ 2  <    |   >^ 0      0.0010856   0.0001872
    2->  5:   3p-> 3d  < 3p| 3p>^ 2  <    |   >^ 0     -0.0089864  -0.0015498
    3->  1:   3p-> 3s  < 3s| 3s>^ 1  < 3d| 3d>^ 1     -0.2425596   0.0504345
    3->  2:   3d-> 3p  < 3s| 3s>^ 1  < 3p| 3p>^ 1      0.0076390  -0.0022564
    3->  3:   3s-> 3p  < 3p| 3p>^ 1  < 3d| 3d>^ 1      0.0349805   0.0060229
    4->  2:   3d-> 3p  < 3s| 3s>^ 1  < 3p| 3p>^ 1      0.0098534  -0.0029106
    4->  5:   3s-> 3p  < 3p| 3p>^ 1  < 3d| 3d>^ 1      0.0023844   0.0004105
    5->  3:   3d-> 3p  < 3p| 3p>^ 1  < 3d| 3d>^ 1     -0.0058007   0.0017134
    6->  3:   3d-> 3p  < 3p| 3p>^ 1  < 3d| 3d>^ 1     -0.0031957   0.0009440
    7->  3:   3d-> 3p  < 3p| 3p>^ 1  < 3d| 3d>^ 1     -0.0080486   0.0023774

          SUM OF LENGTH FORM                  =      2.39649031
          SUM OF VELOCITY FORM                =      0.48227912
       E1 FINAL gf-VALUES: LENGTH  =   0.113651D+01
                           VELOCITY=   0.208955D+01

          ENERGY DIFFERENCE OF THE STATES:       3.25731264D+04 CM-1
                                                 3.07001541D+03 ANGSTROMS(vac)
                                                 1.48417000D-01 A. U.

          Rydberg constant for conversion:          109735.16

          TRANSITION PROBABILITY: (final->initial) =    8.0432484D-01 10^8 sec-1
>> lsjtr                  =======>3.3) Multiplet Transitions
    Name of Initial State
>al2p
    Name of Final State
>al2d
    INTERMEDIATE PRINTING (Y OR N) ?
>y
     TOLERANCE FOR PRINTING ?
>0.001
    Default Rydberg constant (Y or N) ?
```

```
>y
>> lines                   =======>3.4) Line List
   Enter tolerance on line strength
>0.0
   Number of transitions =  3
  Select the line list order:
  1:  Energy (cm-1)
  2:  Wavelength (Angstroms) in Vacuum
  3:  Wavelength (Angstroms) in Air
  4:  Line Strength
  5:  gf Value
  6:  Transition Probability
  Enter your selection:
>2

  Line List for al     ( Z = 13.) with 13 electrons

 -------------------------------------------------------------------------

 Transition Array Multiplet  Line Type E(cm-1) L(air)   S     gf    Aki
 -------------------------------------------------------------------------

 3s(2).3p  3s(2).3d 2P 2D  0.5-1.5  E1 32573.6 3069.1 3.833 0.379 6.71E+07
                           1.5-2.5  E1 32468.0 3079.1 6.901 0.681 7.98E+07
                           1.5-1.5  E1 32467.0 3079.1 0.767 0.076 1.33E+07

 -------------------------------------------------------------------------

>> cat al2p.j al2d.j >al.j =======>4.0) Combine 2P and 2D data

>> levels                  =======>4.1) Print energy level data
   Enter name and type (.l or .j) of file
>al.j

  ENERGY LEVELS
  Z =  13      13 electrons
 ----------------------------------------------------------------
   Configuration         Term  J     Total  Energy  Energy Level
                                        (a.u.)          (cm-1)
 ----------------------------------------------------------------

  3s(2).3p              2P   0.5    -242.3385514        0.00
                             1.5    -242.3380659      106.57


  3s(2).3d              2D   1.5    -242.1901323    32573.60
                             2.5    -242.1901280    32574.54
 ----------------------------------------------------------------
```