

## A program for computing autoionization properties

Charlotte Froese Fischer and Tomas Brage

*Department of Computer Science, Vanderbilt University, Nashville, TN 37235, USA*

Received 27 March 1992

A program for computing atomic autoionization widths, decay rates and lifetimes is presented as a new part of the MCHF atomic structure package. The method used is based on the “configuration interaction in the continuum” theory by Fano, in which the continuum is computed independently of the discrete part and the width is given by the “Golden Rule” formula. The program allows for correlation in the discrete part of the atomic state function and in the target, through multiconfiguration expansions, but is restricted to only one continuum function.

### PROGRAM SUMMARY

*Title of program:* MCHF\_AUTO

*Catalogue number:* ACLD

*Program obtainable from:* CPC Program Library, Queen’s University of Belfast, N. Ireland (see application form in this issue)

*Licensing provision:* none

*Computer for which the program is designed and others on which it has been tested:*

*Computer:* VAX 11/780, SUN SPARC-station 330; *Installation:* Vanderbilt University

*Operating system under which the program has been tested:* VMS, Sun UNIX

*Programming languages used:* FORTRAN77

*Memory required to execute with typical data:* 514K words

*No. of bits in a word:* 32

*Peripherals used:* terminal, disk

*No. of lines in distributed program, including test data, etc.:* 3600

---

Correspondence to: C. Froese Fischer, Department of Computer Science, Vanderbilt University, Nashville, TN 37235, USA.

*CPC Program Library subprogram used:* Catalogue number: ABZU; *Title:* MCHF\_LIBRARIES; *Ref. in CPC:* 64 (1991) 399

*Keywords:* atomic structure, autoionization, line widths, dielectronic recombination, electron emission, resonances, complex atoms, wave functions, continuum processes

#### *Nature of physical problem*

This program is an extension of the MCHF atomic structure package [1] to calculations of autoionization rates and line widths.

#### *Method of solution*

Given a configuration list, radial functions and energies for one or more discrete states, in an *LS* or *LSJ* scheme, the program solves the Hartree–Fock equation for a continuum orbital. This is used to calculate the interaction element between the discrete state and the open channel. From this a number of different autoionization properties can be obtained by using the “Golden Rule” [2].

#### *Restrictions on the complexity of the problem*

The dimensions of the atomic structure package restrict the number of configuration states to 100, but this can readily be changed. Only one continuum orbital is allowed.

#### *Unusual features of the program*

Some nonorthogonalities are allowed [1]. Exchange interaction within the core can be included.

*Typical running time*

The CPU time required for the test run is 1.1 seconds for the first case and 29.1 seconds for the second on a SUN SPARC-station 330.

*References*

- [1] C. Froese Fischer, *Comput. Phys. Commun.* 64 (1991) 369.  
 [2] U. Fano, *Phys. Rev.* 124 (1961) 1866.

## LONG WRITE-UP

### 1. Introduction

This program is an extension of the MCHF atomic structure package [1]. Its role is to calculate autoionization rates, line widths and lifetimes from a given MCHF wave function, by solving the Hartree–Fock equation for the continuum and obtaining the interaction with bound perturbers. It has been tested and used for a wide variety of problems, from autoionization in simple systems [3] to autodetachment in negative ions [4] and nonradiative decay of core excited states [5–7].

### 2. Theory and method of calculation

#### 2.1. The configuration interaction model for autoionization

The foundation for the theory of autoionization was set by the classical paper by Fano [2], where he develops his configuration-interaction (CI) theory for autoionization. This approach is also described in the textbook by Cowan [8]. Based on this we can derive a formula for the width of the autoionizing state in an  $N$ -electron system, according to the “Golden Rule”;

$$\Gamma = 2\pi |\langle \Psi_b(N; \gamma LS) | H - E_0 | \Psi_k(N; E_0 \gamma' L' S') \rangle|^2 \quad (1)$$

in the  $LS$ -coupling scheme, and

$$\Gamma = 2\pi |\langle \Psi_b(N; \gamma LSJ) | H - E_0 | \Psi_k(N; E_0 \gamma' L' S' J') \rangle|^2 \quad (2)$$

in the  $LSJ$  scheme. (A warning to the reader that uses Cowan's book is in place here. He defines  $\Gamma$  as *half* the line width at half maximum, i.e. half our value.) The  $\Psi_b$  in these equations is the discrete state and  $\Psi_k$  represents the open channel. Both can be expressed in a set of *configuration state functions* (CSF),  $\Phi(N; \alpha_i LS)$ , which for the time being we assume to be known. This set can be divided in a localized and a continuum subset. We denote the dimension of the combined set by NCFG, and of the localized subset by ID. We can then express our discrete state in a CI-expansion,

$$\Psi_b(N; \gamma LS) = \sum_{i=1}^{\text{ID}} c_i \Phi(N; \alpha_i LS) \quad (3)$$

or

$$\Psi_b(N; \gamma J) = \sum_{i=1}^{\text{ID}} c_i \Phi(N; \alpha_i L_i S_i J). \quad (4)$$

The  $\Psi_k$  state will be of the form,

$$\Psi_k(N; E_0 \gamma' L' S') = |\Psi_b(N-1; \beta \tilde{L} \tilde{S}) \cdot \phi(kl) L' S'\rangle \quad (5)$$

where  $\Psi_b(N-1; \beta \tilde{L} \tilde{S})$  is the function representing the  $N-1$  electron system, which in the future will be labeled the target state (of dielectronic recombination). In a general case, this state will be expressed as a MCHF-expansion for the  $N-1$  electron system,

$$\Psi_b(N-1; \beta \tilde{L} \tilde{S}) = \sum_{j=\text{ID}+1}^{\text{NCFG}} a_j \Phi(N-1; \beta_j \tilde{L} \tilde{S}). \quad (6)$$

The  $\phi(kl)$  is a one-particle, continuum function and the notation in eq. (5) implies coupling of the orbital and spin momenta of this function with  $\tilde{L}, \tilde{S}$  to form total  $L, S$ .

We will assume that  $H$  in eq. (1) is the nonrelativistic, Coulomb Hamiltonian. Therefore, from eq. (3) (or (4)), we can write the interaction element as,

$$\begin{aligned} V_{E_0} &= \langle \Psi_b(N; LS) | H - E_0 | \Psi_k(N; E_0 L' S') \rangle \\ &= \sum_{i=1}^{\text{ID}} \sum_{j=\text{ID}+1}^{\text{NCFG}} c_i a_j \langle \Phi(N; \alpha_i LS) | (H - E_0) | \Phi(N-1; \beta_j \tilde{L}_j \tilde{S}_j) \cdot \phi(kl) L' S' \rangle \\ &= \delta_{LL'} \delta_{SS'} \sum_{i=1}^{\text{ID}} \sum_{j=\text{ID}+1}^{\text{NCFG}} c_i a_j \langle \Phi(N; \alpha_i LS) | (H - E_0) | \Phi(N-1; \beta_j \tilde{L}_j \tilde{S}_j) \cdot \phi(kl) LS \rangle \end{aligned} \quad (7)$$

or

$$\begin{aligned} V_{E_0} &= \langle \Psi_b(N; \gamma J) | H - E_0 | \Psi_k(N; E_0 \gamma' J') \rangle \\ &= \delta_{JJ'} \sum_{i=1}^{\text{ID}} \sum_{j=\text{ID}+1}^{\text{NCFG}} c_i a_j \delta_{L_i L'_i} \delta_{S_i S'_i} \langle \Phi(N; \alpha_i L_i S_i) | (H - E_0) | \Phi(N-1; \beta_j \tilde{L}_j \tilde{S}_j) \cdot \phi(kl) L'_i S'_i \rangle. \end{aligned} \quad (8)$$

The matrix element on the right hand side of eqs. (7) and (8) will be denoted  $H_{ij}$ .

From the rest of the MCHF atomic structure package, we can readily obtain the CSF's, both for the discrete state,  $\Phi(N; \alpha_i L_i S_i)$ , and for the target state,  $\Phi(N-1; \beta_j L'_j S'_j)$ . It will also supply us with the CI-coefficients,  $c_i$  and  $a_j$ . We can also do the angular integration, with the MCHF\_NONH program [9], for the  $H_{ij}$ -element.

The only unknown part of eqs. (7) and (8) is therefore the one-electron continuum function,  $\phi(kl)$ , or rather its radial part, since we assume it can be written;

$$\phi(kl) = \frac{1}{r} P_{kl}(r) |ls\rangle, \quad (9)$$

where the  $|ls\rangle$  is the known spin-angular part.

## 2.2. The one-electron continuum function

The radial part of the continuum function will be a solution to the differential equation

$$\left( \frac{d^2}{dr^2} + \frac{2}{r} (Z - Y_{kl}(r)) - \frac{l(l+1)}{r^2} + k^2 \right) P_{kl}(r) = \frac{2}{r} X_{kl}(r) + \sum_n \epsilon_{kl, nl} P_{nl}(r), \quad (10)$$

where we use the earlier defined notation [10], according to which  $Y_{kl}$  is the *direct* function and  $X_{kl}$  is the *exchange* function. In deriving this equation it has been assumed that the interaction,  $H_{jj'}$ ,  $j, j' > \text{id}$ ,

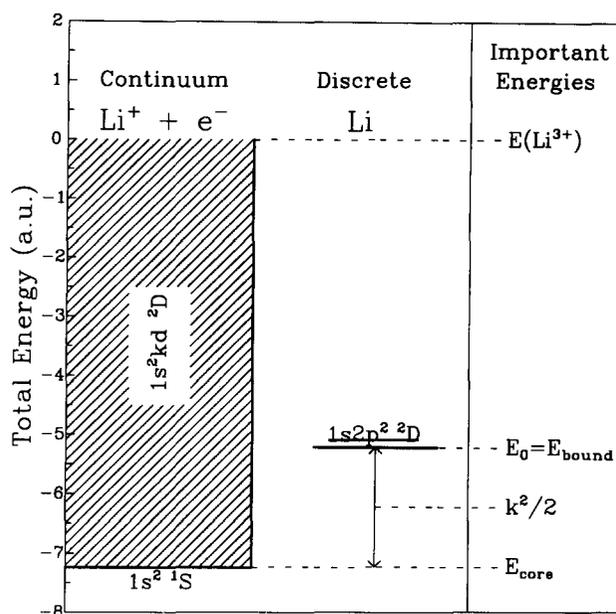


Fig. 1. Term diagram showing the autoionizing level  $1s2p^2 2D$  of lithium, and some important energy properties discussed in the text.

involves only orbitals of the target. This will be the case, for example, when the target CSF's differ by two electrons. Then

$$E_0 = E_{\text{core}} + \frac{1}{2}k^2, \quad (11)$$

where  $E_{\text{core}}$  is the MCHF energy of the target (or core) and  $\frac{1}{2}k^2$  is the energy of the free electron. To illustrate these different properties we use as an example the  $1s2p^2 2D^e$  state of neutral lithium. In fig. 1 we give the most important properties of this system. The core, "inside" the continuum function, is  $1s^2 1S$ . In this case the  $E_{\text{core}}$  is represented by the total energy of this state, while  $E_0$  is the total energy of the discrete  $1s2p^2 2D$  state.

For simplicity we will refer to the whole right hand side of eq. (10) as the exchange function, and define,

$$X_{kl}^{\text{CH}}(r) = \frac{2}{r} X_{kl}(r) + \sum_n \epsilon_{kl,nl} P_{nl}(r). \quad (12)$$

We are looking for positive energy solutions ( $k^2 \geq 0$ ) to eq. (10) and will use the boundary conditions,

$$P_{kl}(r) \rightarrow \sqrt{\frac{2}{k\pi}} [F_{kl}(r) \cos \delta + G_{kl}(r) \sin \delta], \quad (13)$$

where  $F_{kl}(r)$ ,  $G_{kl}(r)$  and  $\delta$  are the regular and irregular Coulomb functions and the short-range phase shift [11], respectively. This choice will give us a function, normalized per unit energy.

To impose the boundary condition in eq. (13) we use the fact that the value of the unnormalized continuum function,  $\bar{P}_{kl}$ , in a point far from the nucleus, is given by

$$\bar{P}_{kl}(r) = C [F_{kl}(r) \cos \delta + G_{kl}(r) \sin \delta]. \quad (14)$$

If we use two points in this region,  $r_{MJ}$  and  $r_{MJ-1}$ , we can derive the following expressions for the phase shift:

$$\tan \delta = \frac{\bar{P}_{kl}(r_{MJ})F_{kl}(r_{MJ-1}) - \bar{P}_{kl}(r_{MJ-1})F_{kl}(r_{MJ})}{\bar{P}_{kl}(r_{MJ-1})G_{kl}(r_{MJ}) - \bar{P}_{kl}(r_{MJ})G_{kl}(r_{MJ-1})} \quad (15)$$

and the normalization constant

$$C = \frac{\bar{P}_{kl}(r_{MJ})}{F_{kl}(r_{MJ}) \cos \delta + G_{kl}(r_{MJ}) \sin \delta}. \quad (16)$$

The normalized continuum function is then:

$$P_{kl}(r) = \sqrt{\frac{2}{k\pi}} \bar{P}_{kl}(r)/C. \quad (17)$$

Before discussing how to choose the values of  $r_{MJ}$  and  $r_{MJ-1}$ , big enough for eq. (15) to be valid, let us first describe the different interesting properties that can be computed after the continuum function has been obtained.

### 2.3. Autoionization properties

The interaction element,  $V_{E_0}$ , of eq. (7) for the *LS*-case and eq. (8) for the *LSJ*-case, can now be computed. From this element a number of observable properties can be derived, following the Fano approach. First, the width of the resonance at half maximum is given by

$$\Gamma = 2\pi |V_{E_0}|^2. \quad (18)$$

To go further, we have to introduce physical constants, which introduces uncertainties outside the actual calculations. The user that is interested in highly accurate calculations is therefore recommended to use the value of the interaction element supplied by the code, and derive other properties with the current best values available for the physical constants.

The decay rate of autoionization is given by

$$A = \frac{\Gamma}{\hbar} = \frac{2\pi V_{E_0}^2}{\hbar} \approx 2.5976 \times 10^{17} V_{E_0}^2 \left( \frac{\text{s}^{-1}}{\text{au}} \right). \quad (19)$$

Finally the lifetime, if we only consider this autoionization branch, is

$$\tau = \frac{1}{A}. \quad (20)$$

All these properties are supplied by the code, as we will see when looking at some specific examples. Before that, though, we will discuss some details of the numerical calculations and describe the structure of the program.

### 2.4. The asymptotic region

One of the most important problems in the numerical solution of eq. (10), is to find a good approximation for the onset of the asymptotic region, where we can impose the boundary condition in

eq. (13) and use eqs. (15)–(17). There are a number of conditions that could be used. The  $F_{kl}$  and  $G_{kl}$  are solutions to the Coulomb case, for which the differential equation can be written

$$\left( \frac{d^2}{dr^2} + \frac{2Z_{\text{eff}}}{r} - \frac{l(l+1)}{r^2} + k^2 \right) P_{kl}(r) = 0 \quad (21)$$

and comparing with eq. (10), we find the following conditions:

- The exchange function,  $X_{kl}^{\text{CH}}(r)$ , of eq. (12) should be negligible.
- The direct function should be constant, so that we can define an effective  $Z$ ,  $Z_{\text{eff}} (= Z - Y_{kl}(r))$  when  $r \rightarrow \infty$ .

These conditions place an inner limit on our “asymptotic” region. The numerical method, or more precisely, the choice of grid, place an outer limit on the region. We use in this program, as in the whole MCHF\_ASP package, a logarithmic grid [1], which is very efficient for discrete orbitals, but is a severe constraint for the continuum problem. To improve our accuracy we reduce by half (in the logarithmic variable) the step-size for the region outside the classical turning point,  $r_{\text{NJ}}$ , where

$$k^2 \approx -\frac{2}{r_{\text{NJ}}} [Z - Y_{kl}(r_{\text{NJ}})] + \frac{l(l+1)}{r_{\text{NJ}}^2}. \quad (22)$$

As an outer cut-off, we use the point,  $r_{\text{MX}}$  where for all discrete, radial orbitals,  $P_{nl}(r)$ , we have

$$|P_{nl}(r)|^2 < \epsilon_{\text{MX}} \quad \text{if } r > r_{\text{MX}}. \quad (23)$$

The choice of  $\epsilon_{\text{MX}}$  is somewhat arbitrary, but can readily be changed. We also find the point, outside which the direct function is “constant”:

$$\frac{|\Delta Y_{kl}(r)|}{Z} < \epsilon_{\text{MP}} \quad \text{if } r > r_{\text{MP}}, \quad (24)$$

where  $\Delta Y_{kl}(r)$  is the difference in the value of  $Y_{kl}(r)$  in two consecutive grid points. The choice of outer cut-off might be too optimistic, since the step-size grows rapidly for large  $r$ . We will therefore reduce the outer region, to a new value  $r_{\text{M}}$ :

$$k \Delta r < \epsilon_{\text{M}} \quad \text{if } r > r_{\text{M}}, \quad (25)$$

where  $\Delta r$  is the distance between two grid points. Finally, we will find a point, where to impose the boundary condition (eq. (13)) and find the normalization for our solution. This point,  $r_{\text{MJ}}$  will be defined as the point where the exchange function is small:

$$\left| \frac{X_{kl}^{\text{CH}}(r)}{f_{kl}(r)} \right| < \epsilon_{\text{MJ}} \quad \text{if } r > r_{\text{MJ}}, \quad (26)$$

where  $f_{kl}(r)$  is defined as

$$f_{kl}(r) = -\frac{2}{r} [Z - Y_{kl}(r)] + \frac{l(l+1)}{r^2} + k^2. \quad (27)$$

Table 1  
 $\epsilon$ 's, limits and subroutines used for evaluating "asymptotic" points.

$X$	Limits of $r_X$	$\epsilon_X$	Subroutine
MX	$r_1 < r_{MX} < r_{NOD}$	$10^{-5}$	CALCAUTO
NJ	$r_{90} < r_{NJ} < r_{MX}$		COULOM
MP	$r_{NJ} < r_{MP} < r_{MX}$	$10^{-6}$	COULOM
M	$r_{NJ} < r_M < r_{MX}$	2	CSOLVE
MJ	$r_{MP} < r_{MJ} < r_M$	0.0025	CSOLVE

The range, in which we are looking for these points, the values of the "right-hand-sides" and the subroutine where the evaluations are done, are given in table 1. The values used for evaluation can easily be changed by the user.

### 3. Input data

The autoionisation calculation might be based on different representations of the discrete state, as outlined in the first section. In the program this will be referred to as different options and values of the variable ICASE, as summarized in table 2. The program requires at least three types of files [1],  $\langle \text{name} \rangle.c$ ,  $\langle \text{name} \rangle.w$  and INT.LST. For the second and third option it also need a  $\langle \text{name} \rangle.l$  and  $\langle \text{name} \rangle.j$  file, respectively.

For all three types of calculation, the list of discrete CSF's and the representation of the continuum is supplied from the  $\langle \text{name} \rangle.c$  file. The latter should be added at the end of the  $\langle \text{name} \rangle.c$  file. This includes both the list of target CSF's, with an continuum orbital added, and the  $a_j$  mixing coefficients of eq.(6). The angular coefficients of the interaction element are always read from the INT.LST and the orbital functions are always read from the  $\langle \text{name} \rangle.w$  file. Other properties are supplied from different sources, depending on the type of calculations.

The first option is to obtain the discrete state from an MCHF calculation [12]. The CFG.OUT file [1] then contains not only the list of CSF's that represent this state, but also the  $c_i$  coefficients of eq. (3) and the energy  $E_0$ . This file should then be copied to the  $\langle \text{name} \rangle.c$  file, before the continuum is added at the end. The user is in this case asked to supply the energy of the core,  $E_{\text{core}}$ , from the terminal.

The second and third option both assume that the discrete state has been obtained from a CI calculation [13]. Option 2 is based on an *LS*-representation of the discrete state as in eq. (3) and the  $c$  coefficients and  $E_0$  energy are obtained from an  $\langle \text{name} \rangle.l$  file. Option 3 assumes an *LSJ*-representation as in eq. (4), and the  $c_i$ 's and  $E_0$  are read in from a  $\langle \text{name} \rangle.j$  file. In both these case the  $E_{\text{core}}$  energy is supplied from the header of the  $\langle \text{name} \rangle.c$  file.

Table 2  
 Input for different type of calculations

Option	Files required	Files containing			
		$c_i$	EBOUND	$a_j$	ECORE
1	w, i, c	cfile	cfile	cfile	Terminal
2	w, i, c, l	lfile	lfile	cfile	cfile
3	w, i, c, j	jfile	jfile	cfile	cfile

We will return to the use of different files when discussing specific examples in a later section. Let us now only list, property by property, the source files for the different options. To solve eq. (10) and compute autoionization properties from eqs. (7) and (8), we need the following input:

- The set of discrete CSF and their couplings. This is supplied from the  $\langle \text{name} \rangle.c$  file in the normal format [1].
- The  $c_i$  coefficients in eq. (3) (or (4)). This is supplied from different files, depending on the type of calculation (see table 2).
- The wave functions for all “discrete” orbitals, which are supplied from the  $\langle \text{name} \rangle.w$  file.
- The list of continuum CSF and their coupling should be added at the end of the  $\langle \text{name} \rangle.c$  file. The target can be represented by a linear combination of CSF (according to eq. (6)), but only one continuum orbital is allowed.
- The  $a_j$  coefficients in eq. (6) should be added at the end of the  $\langle \text{name} \rangle.c$  file, in the normal cfile format [1].
- The energy of the discrete state (EBOUND, denoted by  $E_0$  in section 2) is read from the  $\langle \text{name} \rangle.c$ ,  $\langle \text{name} \rangle.l$  or  $\langle \text{name} \rangle.j$  file, according to table 2.
- The energy of the target (ECORE) is supplied from the terminal or the  $\langle \text{name} \rangle.c$  file according to table 2.
- Angular part of the interaction elements,  $H_{ij}$ , in eqs. (7) or (8), can be obtained from the MCHF\_NONH program [9] and stored in the INT.LST file. Only the last NCFG-ID rows of the total energy matrix is needed, which is a possible option in running the MCHF\_NONH program.

#### 4. Program organization

The structure of the MCHF\_AUTO program is depicted in figs. 2 and 3. Routines in thick line boxes and with bold style names are new, while the others have been described earlier [12,14]. Here we will describe the new routines in the order in which they are called.

##### *AUTO*

The main program initializes data arrays, handles the input/output interface and selects the type of calculation (according to table 2). It calls ADATA and CALCAUTO.

##### *ADATA*

This routine reads and interprets the  $\langle \text{name} \rangle.c$  file, determines data about the continuum electron and sets orthogonality constraints. Finally it initializes data arrays, by calling WAVEFN and ANTGRL.

##### *CALCAUTO*

This routine determines  $r_{MX}$ , according to eq. (23) and table 1. Depending on the value of ICASE it continues according to:

ICASE = 1: The header card of the  $\langle \text{name} \rangle.c$  file contains the energy of the discrete state, EBOUND. If ID  $\neq$  0 the user is prompted to give the target energy (ECORE). After that CALCAUTO calls ASCF, to perform one calculation according to eq. (7). Before that it checks the  $\delta$ -factors in this equation by calling LSTERM. If ID = 0, the value of ICASE is changed to 4 and only a continuum Hartree–Fock calculation is performed – the user is prompted to give the energy of the continuum electron (EBOUND-ECORE in other calculations), and ASCF is called.

ICASE = 2: In this case the weights,  $c_i$  are taken from the  $\langle \text{name} \rangle.l$  file and the process described above is performed for all states in this file.

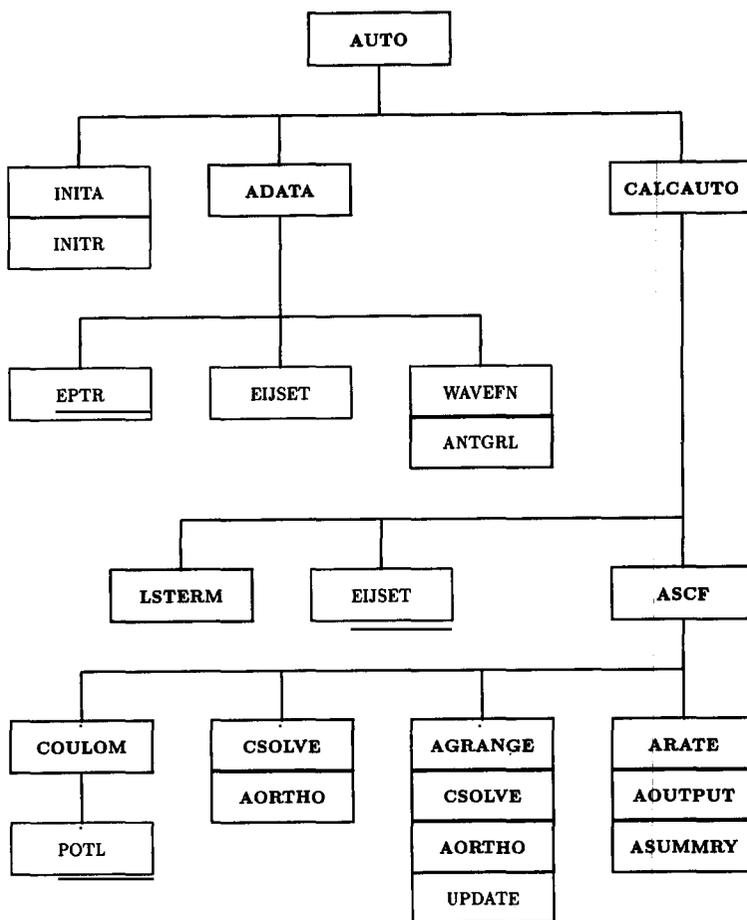


Fig. 2. Block diagram for the MCHF\_AUTO program.

ICASE = 3: In this case we solve eq. (8), and weights are read in from  $\langle \text{name} \rangle_j$ . A loop over the procedure for ICASE = 1 is done over all states for a given  $J$ -value and over  $J$ -values.

**LSTERM**

Determines the *LS* term value from the configuration data.

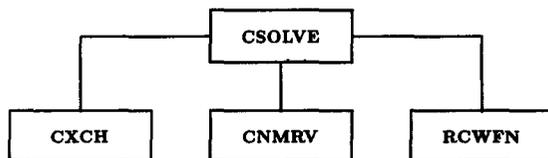


Fig. 3. Block diagram for the CSOLVE routine.

### ASCF

This routine controls the solution of the differential eq. (10). It consists of four main stages:

- It calls COULOM to calculate the direct function and define the asymptotic region.
- It obtains initial estimates of the continuum function by calling CSOLVE and AORTHO.
- It iterates to solve eq. (10), and during each iteration it calls AGRANGE to calculate the off-diagonal Lagrange Multipliers,  $\epsilon_{kl,nl}$ , CSOLVE to solve the equation, AORTHO, to orthogonalize the solution and UPDATE, to update the integrals.
- It calculates autoionization properties by calling ARATE and summarizes the calculation by calling AOUTPUT and ASUMMARY.

### COULOM

This routine calculates and stores the function  $f_{kl}(r)$  of eq. (27) and it determines  $r_{NJ}$  and  $r_{MP}$  according to eqs. (22) and (24). Finally it interpolates  $f_{kl}(r)$  to half the step size in the region outside  $r_{NJ}$ .

### CSOLVE

This routine is called in two different stages. To initialize the continuum function ( $IX = 0$ ) and during the actual iterative procedure ( $IX = 1$ ). It will perform the following tasks:

- Computes exchange Function, by calling CXCH.
- If  $IX = 0$  it redefines the outer region, according to eq. (25). If  $IX = 1$  it uses eq. (26).
- It interpolates  $X^{CH}$  in the outer region,  $[r_{NJ}, r_M]$ , to half the grid size.
- It calls CNMRV, to solve the differential equation.
- It finds the  $F_{kl}(r)$  and  $G_{kl}(r)$  of eq. (13), in  $r_{MJ}$  and  $r_{MJ-1}$ , by calling RCWFN.
- It calculates the phase shift from eq. (15) where  $\bar{P}_{kl}(r)$  is the unnormalized solution obtained from CNMRV. The normalisation constant is obtained from eq. (16) and the normalized solution from eq. (17).
- Finally it orthogonalize the continuum function to all discrete functions with the same  $l$ -value.

### ARATE

This routine evaluates the  $H_{ij}$  matrix elements, and calculates the interaction element, according to eqs. (7) or (8). From this the *half-line width*, *autoionization rate* and *half-life* are calculated, using eqs. (18)–(20).

### CNMRV

This routine solves the differential equation (10), with outwards integration (Numerovs method), in two different regions:  $[r_0, r_{NJ+1}]$  is the inner region with normal step size, and  $[r_{NJ+2}, r_M]$  is the outer with “half” the step size. CNMRV is a modified version of NMRV [12].

### RCWFN

This is a routine for computing regular and irregular Coulomb wave functions [11]. It returns the value of these and their derivatives for specific values of  $Z_{\text{eff}}$  and radial coordinates and for a range of  $l$ -values (see eq. (21)). The conventions of Abramowitz and Stegun [15] are used. This subroutine is an adaptation of the “Manchester” routine by Barnett et al [16]. The present version of this routine was written by Pieper [17].

ANTGRL, AORTHO, AGRANGE, AOUTPUT, ASUMMARY and CXCH are just modified versions of the corresponding routines in the MCHF program [12] (Namely INTGRL, ORTHOG, GRANGE, OUTPUT, SUMMARY and XCH).

## 5. Examples

The output in the Test Run Output contains two examples. The first illustrates *LS*-allowed autoionization from one single state, while the second illustrates calculations for a set of states, including spin-forbidden decay.

Example 1 is a simple calculation for the *Feschbach* resonance  $1s2p^2\ ^2D$  of Li I. After running MCHF, we store the radial functions for the discrete state in the li2D.w file. The li2D.c file consists of the CFG.OUT from MCHF with the  $1s^2kd_c\ ^2D$  continuum state added with weight 1.0 at the end. The header of this file contains the energy of the discrete state ( $E_0 = -5.2239685$  au). The next step is to run NONH to get the angular part of the interaction element (the li2D.c file is copied to cfg.inp). Only the last row of the interaction matrix is needed (which gives the interaction between the continuum  $1s^2kd_c\ ^2D$  and the discrete CSF's, not the interaction within the discrete state). This is accomplished by giving NEW = 1 and NZERO = 0.

After this, we are ready to start the actual autoionisation calculation. After the name of the case is specified, the program enquires about the type of calculation (according to table 2). In this example, we choose the first type, and the program prompts us for the target energy. We use the result from a Hartree-Fock calculation of the  $1s^2\ ^1S$  state of  $Li^+$  ( $E_{core} = 7.2364152$  au). Finally, we are asked to define the case, by giving the ion name, term and nuclear charge,  $Z$ .

The iterations can be monitored from the terminal since the program prints a number of important data. First it gives the energy of the continuum electron,  $\frac{1}{2}k^2$ , in atomic units. During each iteration the phase shift,  $\delta$ , of eq. (15), the coefficient  $\lim_{r \rightarrow 0} P(r)/r^{l+1}$ , the normalisation constant,  $C$ , of eq. (16) and the maximum change in the solution are displayed under the heading ED/DELTA, AZ, NORM and DPM, respectively. A well converged solution should give NORM  $\approx 1$  and a small DPM.

A summary of the result is written to the AUTO.DAT file, which is shown in the Test Run Output. The autoionisation data given are derived by using eqs. (18)–(20). Approximate transformation coefficients are used for all properties, except the interaction element,  $V$ , and the half-line width in atomic units. They are, together with the one given in eq. (19), 27.21 eV/au and 219474  $cm^{-1}$ /au. This implies that only the two first properties should be used, as supplied by the program, for calculations that requires a higher accuracy. The CPU time for this AUTO calculation was 1.1 seconds.

The second example is built on case 2 for the MCHF\_CI program [13], the lowest core excited configuration in sodium,  $2p^53s3p$ . The naquart.c file contains 90 configurations that represents the discrete states. The energies ( $E_0$ ) and composition ( $c_i$  of eq. (4)) of the nine with dominant  $2p^53s3p$  component is saved in naquart.j. The corresponding radial functions are saved in naquart.w. We now start by adding the  $2p^6kd_1\ ^2D$  continuum state, with weight 1.0, at the end of naquart.c. The target energy,  $E_{core} = -161.6769459$  au, taken from a Hartree-Fock calculation of  $2p^6\ ^1S$ , is also added to the header of naquart.c. With NONH we calculate the angular part of the interaction matrix elements,  $H_{ij}$ , before we proceed to the AUTO run. After giving the name of the case, AUTO inquires about the type of calculation, in this case the third. The calculations are performed for each discrete state in naquart.j and the final results are displayed in the form of the AUTO.DAT file at the end of the Test Run Output. The CPU time for the AUTO part of the second test case was 29.1 seconds.

## Acknowledgements

We are grateful to Steve Pieper for sharing the RCWFN subroutine with us and allowing us to include it with this program.

This research was supported by a grant from the US Department of Energy, Office of Basic Energy Sciences.

**References**

- [1] C. Froese Fischer, *Comput. Phys. Commun.* 64 (1991) 369.
- [2] U. Fano, *Phys. Rev.* 124 (1961) 1866.
- [3] T. Brage, N. Vaeck and C. Froese Fischer, *J. Phys. B*, in press.
- [4] T. Brage and C. Froese Fischer, *Phys. Rev. A* 44 (1991) 72.
- [5] C. Froese Fischer, *Phys. Rev. A* 34 (1986) 1667.
- [6] T. Brage and J.O. Gaardsted, *Phys. Scr.* 44 (1991) 336.
- [7] J.O. Gaardsted, T. Brage, C. Froese Fischer and D. Sonnek, *Phys. Scr.* 42 (1990) 543.
- [8] R.D. Cowan, *The Theory of Atomic Structure and Spectra*, Univ. of California, Berkeley (1981).
- [9] A. Hibbert and C. Froese Fischer, *Comput. Phys. Commun.* 64 (1991) 417.
- [10] C. Froese Fischer, *The Hartree-Fock Method for Atoms* (John Wiley, New York, 1977).
- [11] U. Fano and A.R.P. Rau, *Atomic Collisions and Spectra* (Academic, New York, 1986).
- [12] C. Froese Fischer, *Comput. Phys. Commun.* 64 (1991) 431.
- [13] C. Froese Fischer, *Comput. Phys. Commun.* 64 (1991) 473.
- [14] C. Froese Fischer, *Comput. Phys. Commun.* 64 (1991) 399.
- [15] M. Abramowitz and I.A. Stegun, *Handbook of Mathematical Functions* US Govt. Print. Off., Washington, (1972).
- [16] A.R. Barnett, D.H. Feng, J.W. Steed and L.J.B. Goldfarb, *Comput. Phys. Commun.* 8 (1974) 377.
- [17] S. Pieper, private communication.

TEST RUN OUTPUT

Case 1

```

>>cat cfg.inp =====> Display configuration list (Discrete).
Li
1s( 1) 2p2( 2)
2S1 1D2 2D0
1s( 1) 2s( 1) 3d1( 1)
2S1 2S1 2D1 1S0 2D0
1s( 1) 2s( 1) 3d1( 1)
2S1 2S1 2D1 3S0 2D0
1s( 1) 3d2( 2)
2S1 1D2 2D0
>>norbh =====> Obtain Energy Expression (Discrete).
FULL PRINT-OUT ? (Y/N)
>n
ALL INTERACTIONS ? (Y/N)
>y
>>mchf =====> Compute Radial Functions (Discrete).
ATOM, TERM, Z in FORMAT(A,A,F) :
>Li,2D,3.
There are 5 orbitals as follows:
1s 2p2 2s 3d1 3d2
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
>>mv wfn.out li2D.w =====> Move wfn.out to li2D.w
... Add continuum state to cfg.inp and delete the last line in the old file.
>>cat cfg.inp =====> Display cfg.inp after continuum is added.
Li 2D -6.2239685
1s( 1) 2p2( 2) 0.9097745
2S1 1D2 2D0
1s( 1) 2s( 1) 3d1( 1) -0.1510400
2S1 2S1 2D1 1S0 2D0
1s( 1) 2s( 1) 3d1( 1) 0.3826859
2S1 2S1 2D1 3S0 2D0
1s( 1) 3d2( 2) 0.0552147
2S1 1D2 2D0
1s( 2) kdc( 1) 1.0
1S0 2D1 2D0
>>norbh =====> Compute Angular Part of Interaction.
FULL PRINT-OUT ? (Y/N)
>n
ALL INTERACTIONS ? (Y/N)
>n
NEW (O=ALL), N=ZERO (O=ALL) ?
>1,0
5
5 matrix elements
5 non-zero matrix elements
100.0000 % dense
NF= 2 NG= 1 NR= 9 NL= 3
Total number of terms = 15
The total number of integrals = 13
>>mv cfg.inp li2D.c =====> Move cfg.inp to li2D.c
>>auto =====> Obtain continuum function
and Compute Autoionization Properties.
=====
A U T O - I O N I Z A T I O N
=====
Name of State
>li2D
Select input file for discrete state(s):
1 name.c
2 name.1
3 name.j
>1
ATOM, TERM, Z in FORMAT(A,A,F) :
>Li,2D,3.
Give energy of core state (a.u.):
>-7.2364152
There are 6 orbitals as follows:
1s 2p2 2s 3d1 3d2 kdc
*** CALCULATIONS FOR k^2/2 = 2.01245 au
WEAK ORTHOGONALIZATION DURING THE SCF CYCLE= F
ACCELERATING PARAMETER FOR MCHF ITERATION = 0.00
SCF CONVERGENCE TOLERANCE (FUNCTIONS) = 1.00D-06
NUMBER OF POINTS IN THE MAXIMUM RANGE = 220
RELATIVISTIC DIAGONAL ENERGY CORRECTIONS = F
Outer region reduced by 8points in CSOLVS. Too large step size!
kdc 0.0192067 0.7869118 0.7869118 c 2.71D-01
ITERATION NUMBER 1
-----
CONVERGENCE CRITERIA = 1.0D-06
EL ED/DELTA AZ WORN DPM
kdc 0.0186544 0.8345167 1.0604959 c 8.70D-02
LEAST SELF-CONSISTENT FUNCTION IS kdc :WEIGHTED MAXIMUM CHANGE = 5.70D-02
... Iterations omitted for brevity ...

```

## Case 2

```

ITERATION NUMBER 5
-----
CONVERGENCE CRITERIA = 2.9D-06

      EL  ED/DELTA  AZ  NORM  DPH
kdc  0.0162426  0.8330101  1.0000000 c  4.21D-08

>>cat auto.dat  =====> Display Autoionization Results.

*****
AUTOIONIZATION CALCULATION
*****
Energy of Target:  -7.23641520 au

Continuum State:
=====
1s( 1) 1D2 2D0
1s( 2) kdc( 1)
1S0 2D1 2D0

Energy of continuum electron (k2/2) = 2.01245 au
=====

Discrete State, main component:
-----
1 1s( 1) 2p2( 2)
2S1 1D2 2D0

Auto-ionization Data
-----
Interaction, V = -9.2331E-03 au
Half-Line Width = 2.6782E-04 au
7.2874E-03 ev
5.8779E+01 cm-1
Auto-ionization rate = 2.2144E+13 s-1
Half-Life = 4.6158E-14 s

... See the writeup for MCHF_CI for details how ...
... the calculations were performed to this point ...
>>levels =====> Display levels in naquart.j
Enter name and type (.l or .j) of file
>naquart.j

ENERGY LEVELS

Z = 11 11 electrons
-----
Configuration Term J Total Energy (a.u.) Energy Level (cm-1)
2p(5).3s1_3p.3p 4S 1.5 -160.9119191 0.00
4D 3.5 -160.8995775 2708.61
2.5 -160.8979607 3065.84
1.5 -160.8963595 3414.85
0.5 -160.8950320 3706.21
4P 2.5 -160.8903068 4743.25
1.5 -160.8885549 5127.72
0.5 -160.8876608 5323.95

2p(5).3s1_1p.3p 2D 1.5 -160.8851575 5873.35
2.5 -160.8830958 6325.82
2P 0.5 -160.8809687 6792.67
1.5 -160.8796476 7082.60
2S 0.5 -160.8690859 9400.57

2p(5).3s1_3p.3p 2D 2.5 -160.8318630 17569.87
2S 0.5 -160.8316768 17610.73
2P 1.5 -160.8306579 17634.34
2D 1.5 -160.8254799 18970.75
2P 0.5 -160.8248117 19117.41

... The old header card is deleted in naquart.c and replaced by ...
... one containing the ECONF value. The continuum state is added ...
... at the end of the file. naquart.c is copied to cfg.inp. ...

>>cat cfg.inp =====> display cfg.inp
Na 2S -161.6769459
1s 2s
2p( 5) 3s1( 1) 3p1( 1)

```

```

Ia 2D 11. 220 11 1 91 F T
F.0D+00.1D-05 20 00.0 90 F

*** CALCULATIONS FOR k^-2/2 = 0.77900 au

WEAK ORTHOGONALIZATION DURING THE SCF CYCLE= F
ACCELERATING PARAMETER FOR NCFE ITERATION = 0.00
SCF CONVERGENCE TOLERANCE (FUNCTIONS) = 1.00D-06
NUMBER OF POINTS IN THE MAXIMUM RANGE = 220
RELATIVISTIC DIAGONAL ENERGY CORRECTIONS = F

Outer region reduced by 4points
in CSOLVE. Too large step size!
kdi 0.0551805 1.2850599 1.2850599 c 2.22D-01

ITERATION NUMBER 1
-----
CONVERGENCE CRITERIA = 1.0D-06
EL ED/DELTA AZ NORM DPM
kdi 0.1707676 2.2988143 1.7888771 c 4.41D-01

LEAST SELF-CONSISTENT FUNCTION IS kdi :WEIGHTED MAXIMUM CHANGE = 4.41D-01
... Iterations omitted for brevity ...

ITERATION NUMBER 9
-----
CONVERGENCE CRITERIA = 8.2D-06
EL ED/DELTA AZ NORM DPM
kdi 0.1247621 1.9875022 1.0000043 c 4.29D-06

LEAST SELF-CONSISTENT FUNCTION IS kdi :WEIGHTED MAXIMUM CHANGE = 4.29D-06

ITERATION NUMBER 10
-----
CONVERGENCE CRITERIA = 1.1D-06
EL ED/DELTA AZ NORM DPM
kdi 0.1247618 1.9875004 0.9999991 c 9.45D-07

*** CALCULATIONS FOR k^-2/2 = 0.78664 au

WEAK ORTHOGONALIZATION DURING THE SCF CYCLE= F
ACCELERATING PARAMETER FOR NCFE ITERATION = 0.00
SCF CONVERGENCE TOLERANCE (FUNCTIONS) = 1.00D-06
NUMBER OF POINTS IN THE MAXIMUM RANGE = 220
RELATIVISTIC DIAGONAL ENERGY CORRECTIONS = F

```

```

2P1 2S1 2P1 1P0 2S0
2p( 5) 3s1( 1) 3p1( 1)
2P1 2S1 2P1 1P0 2F0
2p( 5) 3s1( 1) 3p1( 1)
2P1 2S1 2P1 1P0 2D0
... Rest omitted for brevity. In total, 90 Discrete CSF ...
...
2p( 5) 3p2( 1) 4d2( 1)
2P1 2P1 2D1 3D0 4F0
2p( 5) 3p2( 1) 4d2( 1)
2P1 2P1 2D1 3D0 4G0
2p( 6) kdi( 1) 1.0
1S0 2D1 2D0

... Calculate the last row of the interaction matrix. ...
>>nonh =====> Get angular part of interaction element.
FULL PRINT-OUT ? (Y/N)
>n
ALL INTERACTIONS ? (Y/N)
>n
NEW (0=ALL), N=ZERO (0=ALL) ?
>1.0
91
91 matrix elements .
21 non-zero matrix elements
23.0769 % dense
NF= 3 NG= 2 NR= 48 NL= 4
Total number of terms = 57
Total number of integrals = 30
>>mv cfg.inp naquart.c
>>auto ===== calculate autoionization data.
=====
A U T O I O N I Z A T I O N
=====

THE DIMENSIONS FOR THE CURRENT VERSION ARE:
MCD=100 MWD= 30 MD=220

Name of State
>naquart
Select input file for discrete state(s):
1 name.c
2 name.l
3 name.j
>3
ATOM, TERM, Z in FORMAT(A,A,F) :
>Na,2D,11.
There are 11 orbitals as follows:
is 2s 2p 3s1 3p1 3p2 3d2 4p1 4s1 4d2 kdi

```

```

*****
AUTOIONIZATION CALCULATION
*****
Energy of Target: -161.67694590 au

Continuum State:
=====
2p( 6) kd1( 1)
150 2D1 2D0

Energy of continuum electron (k2/2) = 0.77900 au
=====

Discrete State, main component:
-----
9 2p( 5) 3s1( 1) 3p1( 1) 2*J = 5
  2P1 2S1 2P1 3P0 4D0

Auto-ionization Data
-----
Interaction, V = -6.7471E-04 au
Half-Line Width = 1.4301E-06 au
3.8914E-05 ev
3.1368E-01 cm-1
Auto-ionization rate = 1.1825E+11 s-1
Half-Life = 8.4666E-12 s

Energy of continuum electron (k2/2) = 0.78664 au
=====

Discrete State, main component:
-----
9 2p( 5) 3s1( 1) 3p1( 1) 2*J = 5
  2P1 2S1 2P1 3P0 4P0

Auto-ionization Data
-----
Interaction, V = 1.2210E-03 au
Half-Line Width = 4.6837E-06 au
1.2744E-04 ev
1.0279E+00 cm-1
Auto-ionization rate = 3.8726E+11 s-1
Half-Life = 2.5822E-12 s

Energy of continuum electron (k2/2) = 0.79385 au
=====

Discrete State, main component:
-----
9 2p( 5) 3s1( 1) 3p1( 1) 2*J = 5
  2P1 2S1 2P1 3P0 4P0

Auto-ionization Data
-----
Interaction, V = 1.2210E-03 au
Half-Line Width = 4.6837E-06 au
1.2744E-04 ev
1.0279E+00 cm-1
Auto-ionization rate = 3.8726E+11 s-1
Half-Life = 2.5822E-12 s

*****
ITERATION NUMBER 1
-----
CONVERGENCE CRITERIA = 1.0D-06

EL ED/DELTA AZ NORM DPM
Outer region reduced by 4points in CSOLVE. Too large step size!
kd1 0.1400036 1.9990506 1.0056114 c 5.78D-03

LEAST SELF-CONSISTENT FUNCTION IS kd1 :WEIGHTED MAXIMUM CHANGE = 5.78D-03
... Iterations omitted for brevity ...

ITERATION NUMBER 7
-----
CONVERGENCE CRITERIA = 4.8D-06

EL ED/DELTA AZ NORM DPM
kd1 0.1259582 2.0021483 1.0000006 c 5.62D-07

... Calculation repeated for each state in the jfile ...
... The last one follows:
...

*** CALCULATIONS FOR k2/2 = 0.85147 au

WEAK ORTHOGONALIZATION DURING THE SCF CYCLE= F
ACCELERATING PARAMETER FOR MCF ITERATION = 0.00
SCF CONVERGENCE TOLERANCE (FUNCTIONS) = 1.00D-06
NUMBER OF POINTS IN THE MAXIMUM RANGE = 220
RELATIVISTIC DIAGONAL ENERGY CORRECTIONS = F

ITERATION NUMBER 1
-----
CONVERGENCE CRITERIA = 1.0D-06

EL ED/DELTA AZ NORM DPM
Outer region reduced by 4points in CSOLVE. Too large step size!
kd1 0.1520969 2.1114624 0.9977833 c 2.22D-03

LEAST SELF-CONSISTENT FUNCTION IS kd1 :WEIGHTED MAXIMUM CHANGE = 2.22D-03
... Iterations omitted for brevity ...

ITERATION NUMBER 7
-----
CONVERGENCE CRITERIA = 4.8D-06

EL ED/DELTA AZ NORM DPM
kd1 0.1360688 2.1260250 1.0000003 c 3.47D-07

>>>cat auto.dat =====> Display the result file.

```

3 2p( 5) 3s1( 1) 3p1( 1) 2\*J = 5  
 2P1 2S1 2P1 1P0 2D0 2\*J = 5

Auto-ionization Data

Interaction, V = 3.8127E-03 au  
 Half-Line Width = 4.5667E-05 au  
 1.2426E-03 ev  
 Auto-ionization rate = 1.0023E+01 cm-1  
 Half-Life = 3.7760E+12 s-1  
 2.6493E-13 s

Energy of continuum electron ( $k^2/2$ ) = 0.84508 au

Discrete State, main component:

6 2p( 5) 3s1( 1) 3p1( 1) 2\*J = 5  
 2P1 2S1 2P1 3P0 2D0 2\*J = 5

Auto-ionization Data

Interaction, V = -3.4107E-03 au  
 Half-Line Width = 3.6545E-05 au  
 9.9440E-04 ev  
 Auto-ionization rate = 8.0208E+00 cm-1  
 Half-Life = 3.0217E+12 s-1  
 3.3094E-13 s

Energy of continuum electron ( $k^2/2$ ) = 0.76503 au

Discrete State, main component:

7 2p( 5) 3s1( 1) 3p1( 1) 2\*J = 3  
 2P1 2S1 2P1 3P0 4S0 2\*J = 3

Auto-ionization Data

Interaction, V = 2.8087E-06 au  
 Half-Line Width = 2.4783E-11 au  
 6.7436E-10 ev  
 Auto-ionization rate = 5.4393E-06 cm-1  
 Half-Life = 2.0492E+06 s-1  
 4.8799E-07 s

Energy of continuum electron ( $k^2/2$ ) = 0.78059 au

Discrete State, main component:

9 2p( 5) 3s1( 1) 3p1( 1) 2\*J = 3  
 2P1 2S1 2P1 3P0 4D0 2\*J = 3

Auto-ionization Data

Interaction, V = -7.2213E-04 au  
 Half-Line Width = 1.6382E-06 au  
 4.4576E-05 ev  
 Auto-ionization rate = 3.5955E-01 cm-1  
 Half-Life = 1.3545E+11 s-1  
 7.3625E-12 s

Energy of continuum electron ( $k^2/2$ ) = 0.78839 au

Discrete State, main component:

8 2p( 5) 3s1( 1) 3p1( 1) 2\*J = 3  
 2P1 2S1 2P1 3P0 4P0 2\*J = 3

Auto-ionization Data

Interaction, V = 1.4907E-03 au  
 Half-Line Width = 6.9811E-06 au  
 1.8996E-04 ev  
 Auto-ionization rate = 1.5322E+00 cm-1  
 Half-Life = 5.7723E+11 s-1  
 1.7324E-12 s

Energy of continuum electron ( $k^2/2$ ) = 0.79179 au

Discrete State, main component:

3 2p( 5) 3s1( 1) 3p1( 1) 2\*J = 3  
 2P1 2S1 2P1 1P0 2D0 2\*J = 3

Auto-ionization Data

Interaction, V = 3.7375E-03 au  
 Half-Line Width = 4.3884E-05 au  
 1.1941E-03 ev  
 Auto-ionization rate = 9.6313E+00 cm-1  
 Half-Life = 3.6285E+12 s-1  
 2.7560E-13 s

Energy of continuum electron ( $k^2/2$ ) = 0.79730 au

Discrete State, main component:

```

2  2p( 6) 3s1( 1) 3p1( 1)          2*j = 3
    2P1  2S1  2P1  1P0  2P0
-----
Auto-ionization Data
-----
Interaction, V      = 4.8185E-04 au
Half-Line Width   = 7.2940E-07 au
                  = 1.9847E-05 ev
Auto-ionization rate = 1.6009E-01 cm-1
Half-Life         = 6.0310E+10 s-1
                  = 1.6581E-11 s

Energy of continuum electron (k^2/2) = 0.84629 au
=====

Discrete State, main component:
-----
5  2p( 6) 3s1( 1) 3p1( 1)          2*j = 3
    2P1  2S1  2P1  3P0  2P0
-----
Auto-ionization Data
-----
Interaction, V      = -1.7193E-03 au
Half-Line Width   = 9.2867E-06 au
                  = 2.5269E-04 ev
Auto-ionization rate = 2.0382E+00 cm-1
Half-Life         = 7.8787E+11 s-1
                  = 1.3023E-12 s

Energy of continuum electron (k^2/2) = 0.85147 au
=====

Discrete State, main component:
-----
6  2p( 6) 3s1( 1) 3p1( 1)          2*j = 3
    2P1  2S1  2P1  3P0  2D0
-----
Auto-ionization Data
-----
Interaction, V      = -2.8763E-03 au
Half-Line Width   = 2.5891E-06 au
                  = 7.0720E-04 ev
Auto-ionization rate = 5.7042E+00 cm-1
Half-Life         = 2.1490E+12 s-1
                  = 4.6533E-13 s

```