ATBASE User's Guide (Version 2)

Ping Wang

October 1996

FPA-96-8

FUSION POWER ASSOCIATES

2 Professional Drive, Suite 248
Gaithersburg, Maryland 20879
(301) 258-0545

1500 Engineering Drive
Madison, Wisconsin 53706
(608) 263-2308

ATBASE User's Guide (Version 2)

Ping Wang

Fusion Power Associates
402 Gammon Place
Suite 280
Madison, WI 53719

October 1996

FPA-96-8

Contents

1. Introduction

ATBASE is a suite of FORTRAN 77 computer codes used for generating large-scale high quality basic atomic data. It includes the following nine codes:

(1) NATBASE: This is the main segment of the package. It is a configuration-interaction code with Hartree-Fock wavefunctions. It computes atomic radial wavefunctions, atomic energy levels, transition oscillator strengths, photoionization cross sections, collision strengths and autoionization rates.

(2) NSTATE: This code sets up the input table of configurations for the NATBASE calculation.

(3) ATMODEL: This code manipulates raw atomic data to form different atomic models with different level structures.

(4) ATTABLE: This code does data organization for the basic data outputted from NATBASE and/or DWBORN, computes all related rate coefficients, and creates an atomic model (i.e., detailed formatted atomic data tables) for applications.

(5) DWBORN: This code computes collision strengths for electron-impact excitation by using a distorted wave Born method.

(6) MICPSSR: This code computes ion impact ionization cross sections (for both single and multiple ionization processes).

(7) BEAMTAB: This code generates a formatted data table for ion impact ionization cross sections.

(8) CKFYED: This code does large scale calculations for fluorescence yields.

(9) CKFTAB: This code generates a formatted data table for fluorescence yields.

During the past two years, we have made significant improvements for this atomic physics calculation package. We have automated the processes for calculating electron impact excitation cross sections and autoionization rates. We have also developed a convenient, user-friendly procedure to run ATBASE for generating large scale atomic data tables. In particular, we have developed a new code, ATMODEL, to interface the large scale raw

atomic data and set up atomic models for applications. The current version of ATBASE is more compact, for most applications only the following four codes are needed:

NSTATE.f/NATBASE.f/ATMODEL.f/ATTABLE.f.

In this report we provide detailed descriptions of how to manage the package and how to perform large scale calculations under UNIX. Detailed discussions of the relevant theoretical models will not be presented here but can be found elsewhere [1,2]. Since this report is intended to be a "user's guide" for the ATBASE package, we will present step by step demonstrations for all revalent cases.

2. Flow Chart for ATBASE Package

Generally speaking, to create an atomic model for a specific application (e.g., opacity calculation, spectroscopy analysis for a plasma), we need to go through the following four steps:

(1) Specify atomic system of interest. This includes providing information for atomic number (Z), number of bound electrons in the system, and a list of electronic configurations.

(2) Compute basic atomic data (energy levels, oscillator strengths, photoionization cross sections, electron collision strengths, etc.) for all ions of the atomic system.

(3) Manipulate raw atomic data to construct an atomic model having the structure suitable for the applications (e.g. fine structure, term structure, configuration average, bundle-n average).

(4) Put all related data together to form an atomic model in the tabulated format so that other application codes can conveniently access the data.

The ATBASE package is designed in such a way that each of the above steps is handled separately by a specific module. We run NSTATE in step one, run NATBASE in step two, run ATMODEL in step three, and run ATTABLE in the last step. The logical flowchart for running the ATBASE package is given in Figure 1.

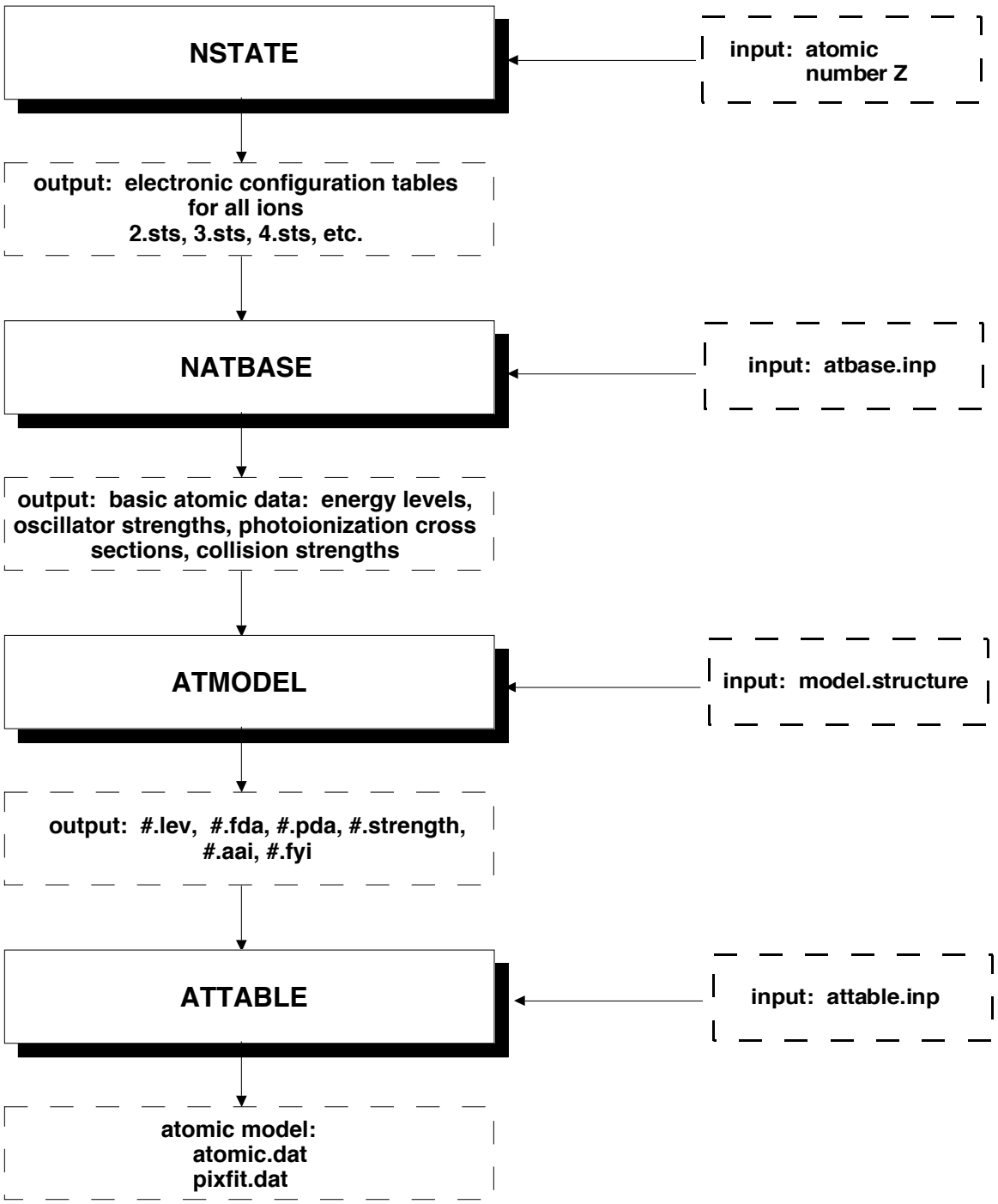

Figure 1. Flow diagram for ATBASE package.

3. Managing ATBASE under UNIX

UNIX is a popular programming enviroment because it includes many tools for program development and management. Two tools are of particular interest in managing the ATBASE package: **make** and **sh**. The **make** utility helps organize and mechanize the creation and updating of an execution module, while **sh** helps us to automate the whole process in large scale calculations. We have attached a makefile for each module if it is necessary. In order to handle the large scale calculations conveniently, each module of ATBASE should have its own directory. A **tree directory structure** for ATBASE directories is presented in Figure 2. Users can follow this **tree** to set up their own directory structure. Under such a **tree** structure, we can always work in a clean working directory with the use of the UNIX utility **sh** as long as we specify the correct PATH in the job control command files. Two sample job control files are provided in Table 6 and Table 8. Following these samples, users should be able to create their own job control files.

4. NSTATE

4.1. Program Outline

NSTATE consists of two parts:

(1) Source code: NSTATE.f

(2) Input table: CONFIGURATION.TAB

This is a code for generating input configuration tables for NATBASE calculations. Strictly speaking, NSTATE only functions as a convertor. It takes one input parameter (atomic number Z) interactively, and then goes into a pre-supplied electronic configuration table, CONFIGURATION.TAB to dig out all revelant configurations and outputs them to the files '#.sts' in the format accepted by NATBASE. Here '#' represents the number of bound electrons of the atomic system. For example, if we want to generate electronic configuratons for the carbon atom and all carbon ions, we input 6. NSTATE will generate five output files: 6.sts, 5.sts, 4.sts, 3.sts, and 2.sts. Each .sts file contains electronic configurations for a specific ion, e.g., 6.sts for neutral carbon, 5.sts for B-like carbon, 2.sts for He-like carbon, etc. To run NSTATE, both executable file and CONFIGURATION.TAB must be put in the same directory.

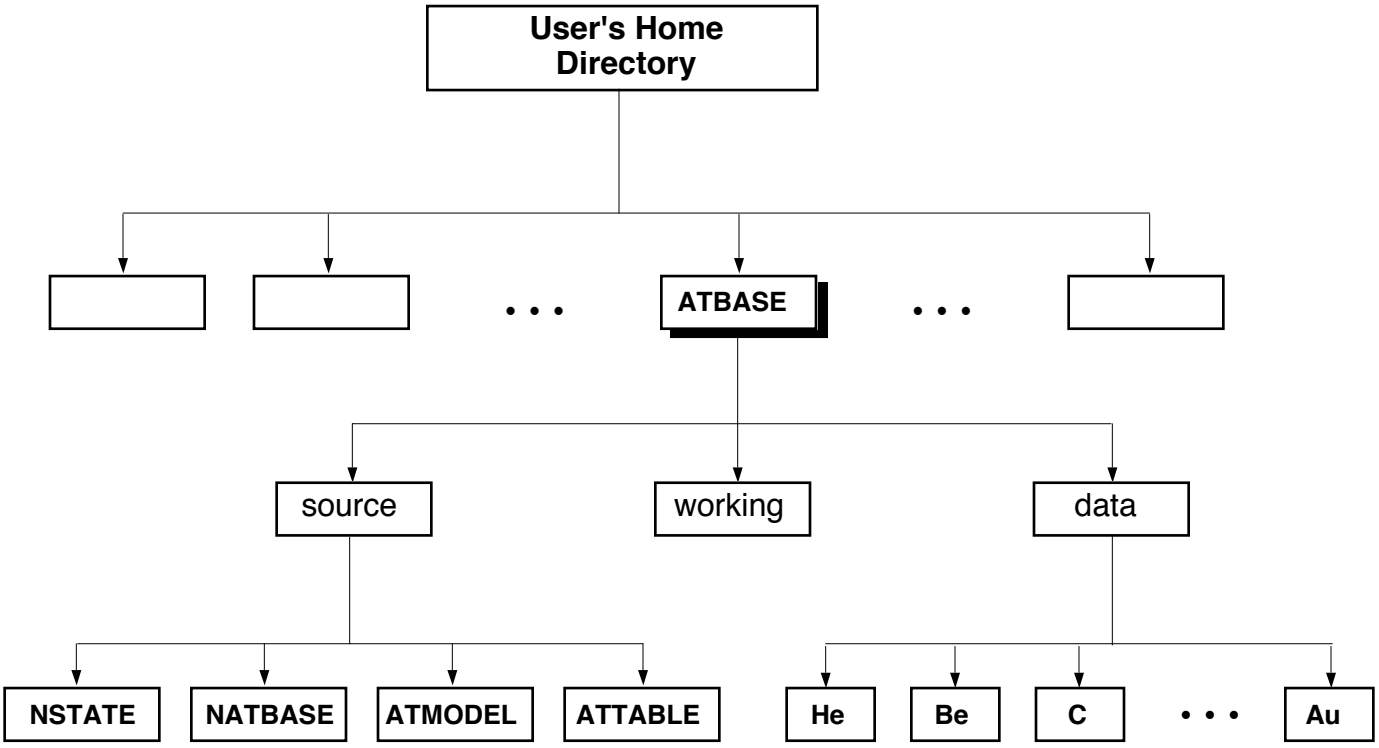

Figure 2. Tree directory structure for ATBASE.

5. NATBASE

5.1. Program Outline

This is a code for the calculation of atomic structure (radial wavefunctions $P_{nl}(r)$ and energy levels) and radiative and collisional transition properties (oscillator strengths, photoionization cross sections, electron impact excitation collision strengths, and autoionization rates). All the output data are in fine-structure (nlLSJ). If we want to have data in simpler structures such as term structure (nlLS) or even configuration average structure, we can manipulate the raw fine-structure data through ATMODEL.

This code is based on the modification of three of Cowan's atomic physics codes (RCN, RCN2, and RCG) [2,3] and Fischer's multiconfiguration Hartree-Fock program (MCHF) [4]. The fine structure levels (nlLSJ) of a many-electron atomic system are evaluated within the framework of a configuration interaction (CI) treatment.

As shown in Figure 3, NATBASE consists of four blocks: primary input, self-consistent-field calculation, CI calculation with intermediate-coupling scheme, and primary output. There are 55 data files involved in the whole calculation. These files can be classified into three groups: input files, intermediate data files, and final output files. A detailed classification for the input and output files is given in Table 1.

5.2. Primary Input

To run NATBASE, the user needs to provide the following information:

(1) specify atomic system,

(2) specify physical models,

(3) specify calculation parameters.

The basic input structure for NATBASE is shown in Figure 4. The primary input consists of four input files:

(1) atbase.inp — a namelist input file which contains all computation switches and parameters.

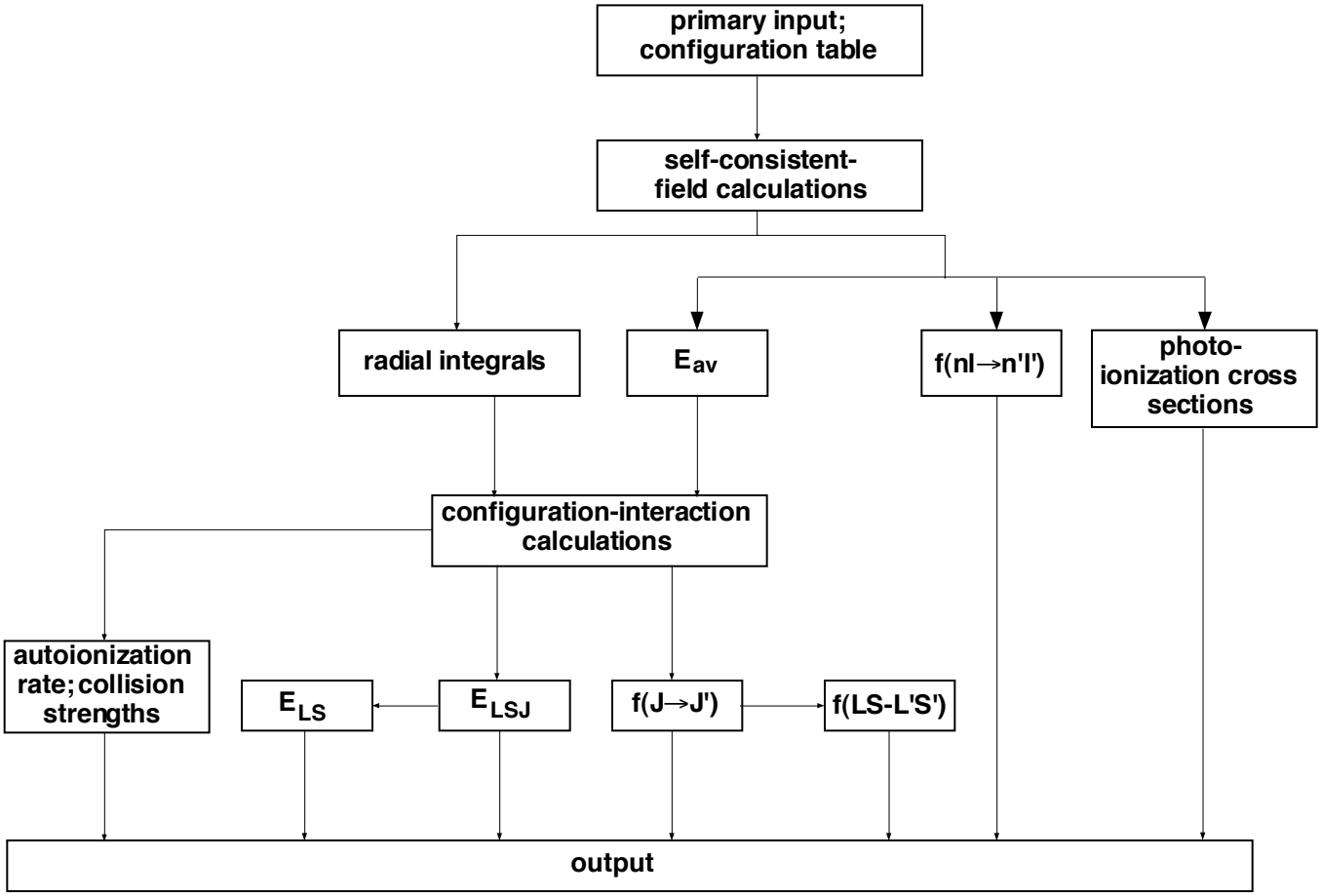

Figure 3. Flow diagram for NATBASE calculations.

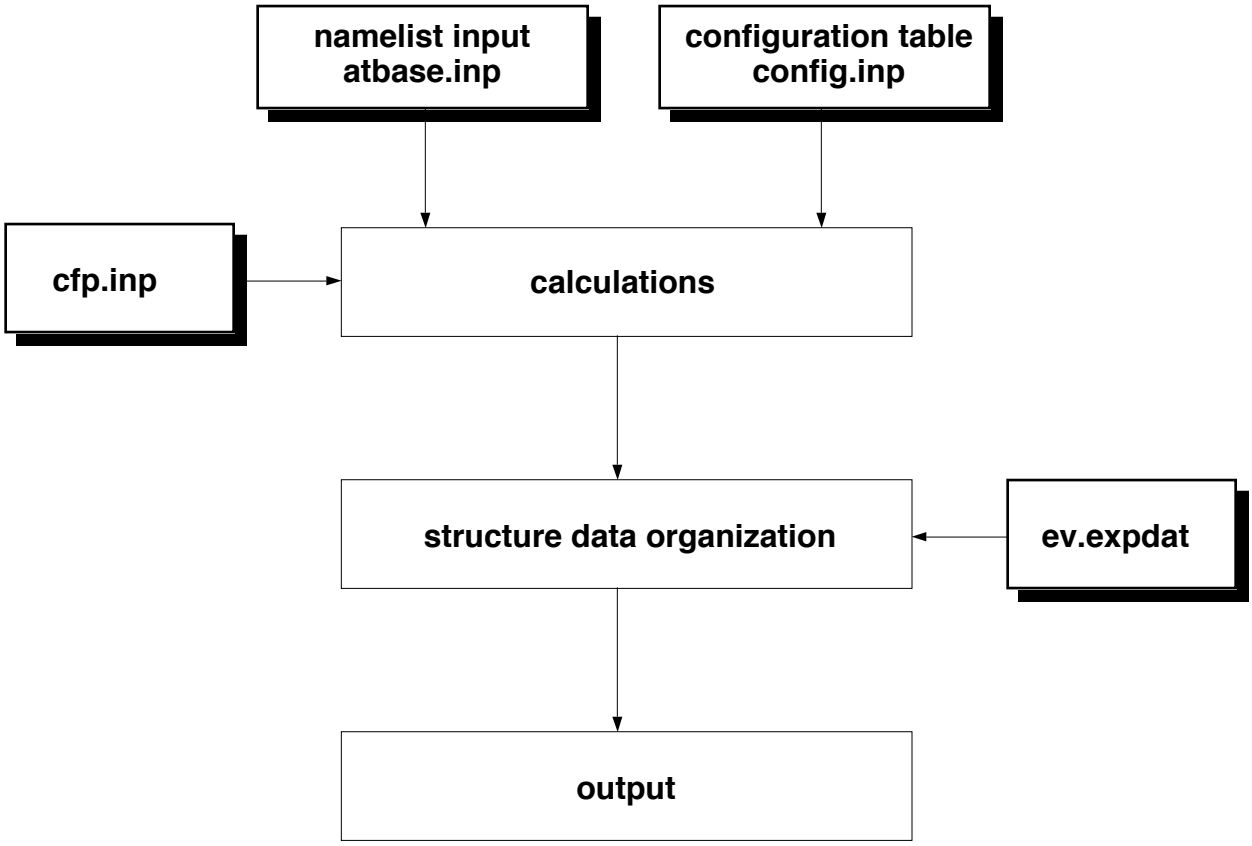

Figure 4. NATBASE input structure.

(2) config.inp — a formatted data input file which specifies the atomic system to be calculated. This file can be generated by running NSTATE.

(3) ev.expdat — a formatted data input file which contains experimental data for atomic energy levels. This is an optional input. This file should be provided only if the user wants to incorporate experimental data in the calculation.

(4) cfp.tab — a list of coefficients of fractional parentage.

Further details of these input files are given in Table 2 through Table 4. It should be noted that general users should use most of the default parameters in 'atbase.inp'; only a few parameters need to be changed for different calculations. In most cases the user only needs to change the following nine parameters:

(1) switch parameters (on=1, off=0):

As default, energy levels will always be calculated.

ISWICH(1) = 1, compute f-values. If you do not want f-value data, turn the switch off by setting ISWICH(1) = 0.

ISWICH(2) = 1, compute photoionization cross sections. If there is no interest in obtaining photoionization cross section data, turn off this switch by setting the parameter to 0.

ISWICH(3) = 0, compute autoionization rates for auotionizing levels.

ISWICH(4) = 0, compute electron impact excitation cross sections.

ISWICH(5) = 0, compute UTA parameters.

ISWICH(6) = 0, incorporate NIST experimental data if necessary.

(2) physical model selections:

imodl1(1) = 0; this parameter decides whether relativistic effects should be included or not. For intermediate-Z and high-Z systems, the user should change this parameter to 1.

imodl1(2) = 4; this parameter selects models of atomic potential. The user has 5 different choices: Hartree model (H), Hartree-Slater model (HS), Hartree-Fock-Slater model (HFS), Hartree-plus-Statistical-Exchange model (HX), and Hartree-Fock (HF) model. Generally speaking, HF model provides the best results, while other models can converge faster.

imodl1(3) = 10; this parameter determines how the electron correlation effects be accounted. For inexperienced users, we recommend a default value of 10.

5.3. Computational Procedure

First of all, NATBASE reads in all electronic configurations from 'config.inp', and calls subroutine DOIT to calculate the single-configuration wavefunctions $P_{nl}(r)$ for each configuration. Also calculated for each configuration are various radial integrals ($\langle \mathrm{r^m} \rangle$, $\mathrm{F^k}$, $\mathrm{G^k}$, ζ), and the configuration-average energy ($\mathrm{E_{av}}$) including approximate relativistic and correlation energy corrections. If IREL is set equal to 1, relativistic terms will be included in the potential function of the radial differential equation to give approximate relativistic corrections to the radial wavefunctions, as well as improved relativistic energy corrections in heavy atoms. The single-configuration structure data calculated in DOIT are stored in a binary data file WFUNT_DAT. With these basic single-configuration structure data, NATBASE can then perform various atomic physics calculations depending on the options selected by the user.

5.3.1. Calculating Atomic Energy Levels and Spectra of an Atomic System

NATBASE calls subroutine RINTG1 to calculate various multiple-configuration radial integrals: overlap integrals $\langle P_{nl} | P_{n'l'} \rangle$, configuration-interaction Coulomb integrals R^k and spin-orbit integrals $\zeta_{nl,n'l'}$, and radial electric-dipole and electric-quadrupole integrals. These data along with the single-configuration average energy and radial integrals will be transferred to subroutine CILSJ via a data file CILSJ_INP. Subroutine CILSJ calculates all relevant angular factors of various matrix elements in the theory of atomic structure and spectra [2] by employing Racah-algebra techniques. These angular factors include

(a) the trivial (unit-matrix) coefficient of $\mathrm{E_{av}}$, the center-of-gravity energy of each configuration;

(b) the coefficients f_k, g_k, and d of the single-configuration direct and exchange Coulomb-interaction (F^k and G^k) and spin-orbit-interaction (ζ) radial integrals, and the coefficients r_d^k and r_e^k of the direct and exchange configuration-interaction Coulomb radial integrals R^k, which are involved in the calculation of the Hamiltonian (energy-level) matrix elements;

(c) the magnetic-dipole matrix elements, and the angular coefficients of the electric-dipole and electric-quadrupole reduced matrix elements

$$P_{l,l'}^{(t)} = \langle l \| r^t \| l' \rangle. \tag{5.1}$$

Combining these angular coefficients with the radial integrals in the data file CILSJ_INP, energy levels and intermediate-coupling eigenvectors are computed. Finally, the energy levels and eigenvectors are used for computation of spectrum-line wavelengths and the associated oscillator strengths and radiative transition probabilities.

5.3.2. Calculating Photoionization Cross Sections

The calculation of photoionization cross sections is performed in subroutine BFDATA. BFDATA first calls subroutine INPUT3 to read in all bound configuration structure data from data file WFUNT_DAT and to determine the electronic configurations of a residual ion after ionization. Then BFDATA calls subroutine EKMESH to construct a universal kinetic energy mesh for an ejected electron in threshold units. Photoionization cross sections for each subshell are calculated in 50 kinetic energy points ranging from threshold to 500 times the threshold. In order to obtain a detailed shape of the cross sections (e.g., Cooper minimums, etc.), the whole kinetic energy region is divided into 5 subregions as shown in Table 5. The mesh points are distributed over each subregion in a logarithmic manner. In the calculation of photoionization cross sections, continuum wavefunctions are calculated by using a frozen core approximation, i.e., all bound wavefunctions are assumed to be unchanged before and after ionization. Configuration-interaction and autoionization resonance structure are not included.

5.4. Automated Procedure for Large Scale NATBASE Calculation

Since detailed atomic physics calculations are usually quite time consuming, it is therefore necessary to automate the calculation process so that we only need to submit the job once

to compute the results for all ions of a specific atomic system. To accomplish this, we take advantage of UNIX's **sh** utility. We have created a shell command file, RUN_ATBASE, which contains a list of commands for handling the movement of I/O files. For different problems, users can modify this file by giving a different PATH for the movement of I/O files. The steps for doing large scale NATBASE calculations are as follows:

(1) copy the executable file, XATDATA.EXE, and input files ATBASE.INP and CFP.TAB to the working directory.

(2) edit RUN_ATBASE to specify the PATH for accessing #.sts and storing output data.

(3) run the job by typing: sh RUN_ATBASE

A sample RUN_ATBASE file is given in Table 6. In this example we assume that all the #.sts files generated from NSTATE are stored in the directory $HOME/ATBASE/DATA/Al, and all the output of NATBASE is stored in the same directory.

5.5. Output

All the output results from NATBASE are stored in 18 data files. There are three different types of output data files which serve different application purposes:

(1) Atomic data tables in conventional expressions of spectral symbols. This type of file can be used for table lookup or data checking. There are eight files in this category:

ev_nl.tab	— configuration-average atomic energy levels;
ev_LS.tab	— LS term energy levels;
ev_LSJ.tab	— fine-structure energy levels;
fv_nl.tab	— the array oscillator strengths (configuration-average);
fv_LS.tab	— oscillator strengths for transitions between terms;
fv_LSJ.tab	— oscillator strengths between levels;
aai_LS.tab	— term-average autoionization rates.
aai_LSJ.tab	— level autoionization rates.

(2) Atomic data files in numerical format. This type of file is used as formatted data input files for the ATMODEL code or other applications. There are seven files in this category:

ev_nl.raw	— configuration-average atomic energy levels;
ev.raw	— fine-structure atomic energy levels;
fv.raw	— oscillator strengths between levels;
aai_LSJ.dat	— level autoionization rates;
ebbx.raw	— electron impact collision strengths;
pbf_x.dat	— subshell photoionization cross sections.
wfunt_dat	— radial wavefunctions (in binary format).

(3) Complete output files. This type of file contains detailed information on the whole calculation.

outpt_scf	— complete output for self-consistent-field calculations;
outpt_integl	— complete output for radial integral calculations;
outpt_lsj	— complete output for atomic data calculations (energy levels, spectra, dielectronic recombination rate coefficients).

All relevant data of dielectronic recombination rate coefficients are written in 'outpt_lsj'.

6. ATMODEL

6.1. Program Outline

This is a code for manipulating raw atomic data output from NATBASE to form an atomic model with a specific level structure. Strictly speaking, fine-structure levels should be used in all applications. However, a model with all levels in fine-structure would be too large, and too complicated to handle in pratice. It is therefore necessary to make some simplications in the level structure for different applications. For example, the analysis of atomic spectroscopy diagnostics requires a detailed specific classification of the level structure of ion (e.g., LS or LSJ resolution), whereas to cope with the very many excited states participating in plasma ionization equilibrium calculations necessitates a less detailed viewpoint (for example nl or even a 'bundle-n' average). ATMODEL takes the fine-structure resolved data as baseline data, makes summations and averages over

these baseline data with the use of the standard sum rules to construct an atomic model with the level structure specified by the user.

The input for ATMODEL includes baseline data files and a parameter input file for specifying level structure.

The input beseline data files include:

(1) ev.raw — fine-structure resolved energy levels
(2) ev_nl.raw — configuration average energy levels
(3) fv.raw — oscillator strengths for transitions between fine-structure resolved levels
(4) aai_LSJ.dat — autoionization rates of fine-structure resolved levels
(5) ebbx.raw — DW electron impact excitation collisionstrengths

If autoionization rates and collision strengths are not calculated in NATBASE, then only the first three sets of baseline data should be included.

Atomic level structure is specified in the input file MODEL.STRUCTUR. Since different approximations should be made for different ions, we should have a different MODEL.STRUCTURE for different ions. A sample MODEL.STRUCTURE is given in Table 7 along with detailed explanations for each input parameter.

The output from ATMODEL consists of several data files containing atomic data in the format accepted by ATTABLE. These output files include:

(1) EV_MODEL.dat — atomic level structure data, corresponding to #.lev of ATTABLE.
(2) FV_MODEL.dat — oscillator strengths, corresponding to #.fda of ATTABLE.
(3) AAI_MODEL.dat — autoionization rates, corresponding to #.aai of ATTABLE.
(4) FYI_MODEL.dat — fluorescence yields, corresponding to #.fyi of ATTABLE.
(5) CX_MODEL.dat — collision strengths, corresponding to #.streng of ATTABLE.

If autoionization rates and collision strengths are not calculated in NATBASE, then only the first two output files will be created.

In large scale calculations, there are a large number of data files which will be involved in this model creation process. In order to avoid corrupting data files, we have created a shell command file, RUN_ATMODEL, which contains a list of commands for handling the movement of I/O files. For different problems, users can modify this file by giving

a different PATH for the movement of I/O files. The steps for runing ATMODEL on a large scale base are as follows:

(1) edit MODEL.STRUCTURE to specify the level structure.

(2) edit RUN_ATMODEL to give the correct PATH for input data and output data.

(3) run the job by typing: sh RUN_ATMODEL.

A sample RUN_ATMODEL file is given in Table 8. In this example we assume that all the baseline data and MODEL.STRUCTURE files are in the directory $HOME/ATBASE/DATA/Al, while output data files are sent to $HOME/APPLICATION.

7. ATTABLE

7.1. Program Outline

ATTABLE serves as an interface between model atomic data and applications (e.g., CRE and/or equations of state and opacity calculations). ATTABLE first reads in all the model atomic data from atomic data files for each ion, and then calculates rate coefficients for all related atomic processes in the temperature and density region specified by the user, and finally creates an atomic model for applications. An atomic model is actually two formatted data tables, ATOMIC.DAT and PIXFIT.DAT, containing the information of atomic energy levels, oscillator strengths, photoionization cross sections, and rate coefficients of all related atomic processes for ions of an element.

There could be as many as 95 files involved in running ATTABLE. These files are listed in Table 9, along with the default logical unit numbers, names, types, and a brief description of their contents.

7.2. Input

There are two types of input for ATTABLE. One is for atomic model specification, and the other provides model atomic data.

Two files, ATTABLE.INP and LEVELS.INP, are used for atomic model specification. Further details of these two files are given in Tables 10 and 11, respectively.

Atomic data files are arranged in the sequence of ionic state. There are five different data files for each ion. Each file is designated by the number of bound electrons in the ion and an extension specifying data property. #.LEV is a data file which contains atomic structure data such as energy levels, number of shells, binding energy of each shell and the radius of each shell, etc., where # represents the ion with # bound electrons. #.FDA is a data file contains all the oscillator strengths, #.PDA is a data file containing photoionization cross sections, and #.STRENG is a data file containing collision strengths of electron impact excitations. Data file #.AAI is a data file containing autoionization rates, and #.FYI is a data file containing fluorescence yields. #.PDA is generated directly from ATBASE calculations, while others are generated from ATMODEL. If #.STRENG is not supplied, the collisional coupling between levels will not be complete; only electric-dipole-allowed couplings are considered. If #.AAI and #.FYI are not supplied, the model will not be suitable for x-ray spectroscopy analysis.

7.3. Rate Coefficient Calculations

Rate coefficients for the following atomic physics processes are calculated in ATTABLE. A Maxwellian distribution is assumed for free electrons in all cases.

(1) Spontaneous decay. Spontaneous decay rates are deduced from the corresponding transition energies and oscillator strengths.

(2) Electron impact excitation. For electric dipole allowed transitions, electron impact excitation rate coefficients are calculated by using a semiclassical impact parameter method [5], while for forbidden transitions, collision strengths must be supplied from data file #.STRENG which is generated from DWBORN calculations.

(3) Electron collisional deexcitation. Deexcitation rate coefficients are obtained from the detailed balance relationship with excitations.

(4) Electron impact ionization. Rate coefficients are calculated by using Burgess' semiempirical formula [6].

(5) Electron collisional recombination (three body recombination). Rate coefficients are obtained from the detailed balance relationship with ionizations.

(6) Radiative recombination. Rate coefficients are obtained by integrating Hartree-Fock photoionization cross sections weighted by a Maxwellian distribution.

(7) Dielectronic recombination. Rate coefficients are calculated by using Burgess-Mert semiempirical formula [7].

7.4. Output

Two output files, ATOMIC.DAT and PIXFIT.DAT, are generated from running ATTABLE.

ATOMIC.DAT contains the following information for a specific atom:

(1) atomic nuclear charge Z;

(2) total number of atomic energy levels included in the model;

(3) energy level structure for each ion;

(4) oscillator strengths and spontaneous decay rates for all transitions;

(6) autoionization rates;

(6) rate coefficients for b-b transitions at a specially designed plasma temperature grid;

(7) rate coefficients for b-f transitions at a specially designed plasma temperature and density grid.

PIXFIT.DAT contains subshell photoionization cross sections for all states in the model. Photoionization cross sections are fit to

$$\sigma(\nu) = \sigma(\nu_1)\{\beta(\frac{\nu_1}{\nu})^{\phi} + (1-\beta)(\frac{\nu_1}{\nu})^{\phi+1}\} \tag{7.2}$$

where ν_1 is the ionization threshold value. Four fitting parameters, $\sigma(\nu_1)$, ν_1, β and ϕ are listed in PIXFIT.DAT for each subshell of all atomic states.

8. DWBORN

8.1. Program Outline

DWBORN is a code for computing electron impact excitation cross sections by using a distorted wave Born method [8]. The calculations of this code may serve two purposes:

(1) as a benchmark for the calculated results of semiclassical impact parameter method;

(2) to provide large scale collision strength data for forbidden transitions. In large scale calculations DWBORN should be run in a chain with ATBASE so that all the energy levels are in the same indexing order. There are 15 I/O data files involved in DWBORN calculations. These files are listed in Table 12, along with the default logical unit numbers, names, types, and a brief description of their contents.

8.2. Input

The input of DWBORN consists of three input files:

(1) dwborn.inp — this is a namelist input file which contains information on target atom, incident electron, computational switches and parameters for a specific problem.

(2) dwborn.sts — this file contains a list of atomic states of the target atom. DWBORN will calculate electron impact excitation cross sections for transitions between these states.

(3) wfunt.dat — this is a binary data file which contains radial wavefunctions for all states listed in DWBORN.STS. It is generated from ATBASE calculations.

Further details on DWBORN.INP and DWBORN.STS are described in Tables 13 and 14. If DWBORN is not run in a chain with ATBASE, all radial wavefunctions will be calculated within DWBORN and the data file WFUNT.DAT will not be needed.

8.3. Output

DWBORN generates 4 output files:

(1) eaout.read — formatted output for cross sections and collision strengths;

(2) eaout.rate — formatted output for rate coefficients and average collision strengths;

(3) eaout.plot — plotting data file:

x - incident electron kinetic energies,

y1 - electron impact excitation cross sections,

y2 - collision strengths;

(4) streng.dat — fitting parameters for collision strengths in the format accepted by ATTABLE. If there are more than three atomic states involved in the calculation, the calculation is viewed as a 'large scale' calculation. Output files EAOUT.READ, EAOUT.PLOT, and EAOUT.RATE will not be generated in large scale calculations.

9. MICPSSR

9.1. Program Outline

MICPSSR is a code for calculating ion-impact ionization cross sections for both single and multiple ionization processes. This code employs a modified plane-wave Born method (MPWBA) [9] which incorporates binding effect, Coulomb-deflection effects, and relativistic corrections for target wavefunctions. This method has been shown [9,10] to produce much better results than the conventional plane-wave Born approximation. The calculation for the multiple ionization cross section is based on a formulation of the single-electron ionization probability with the binomial distribution [11]. This code is usually run in a chain with another code, BEAMTAB, to generate data tables for analyzing inner-shell emission spectra of ion beam heated plasmas.

There are 5 I/O data files involved in MICPSSR calculations. These files are listed in Table 15, along with the default logical unit numbers, names, types, and a brief description of their contents.

9.2. Input

To run MICPSSR, the user needs to provide the following information:

(1) Target ion properties — nuclear charge, atomic weight, number of bound electrons, electronic configuration.

(2) Ionization process specification — specify principal quantum (n), orbital quantum number (l) for each ionizing shell.

(3) Projectile ion properties — nuclear charge, atomic weight, net charge, beam energy.

(4) Calculation controlling parameters — most parameters have been set by default. The user only needs to specify one debugging switch and a mesh for beam energies.

All the inputs for MICPSSR calculations are contained in a namelist input file 'mi_cpssr.inp'. Further details on 'mi_cpssr.inp' are described in Table 16.

9.3. Output

MICPSSR generates 3 output files:

(1) Xsecton.tab — an output table for ion impact ionization cross sections.

(2) Xsecton.dat — data file for ionization cross sections in the format accepted by computer code BEAMTAB as input.

(3) debug.out — debugging output information.

10. BEAMTAB

10.1. Program Outline

BEAMTAB does data organization to generate a data table for analyzing inner-shell emission spectra of ion beam heated plasmas. BEAMTAB first reads all energy levels from data files #.lev which are the same as those for ATTABLE calculations. Then it searches for the upper and lower levels connected by ion impact ionizations and couples them with the proper cross sections.

10.2. Input

The inputs for BEAMTAB include atomic energy levels and ion impact ionization cross sections. Hence, the input for BEAMTAB consists of the following files:

(1) beamtab.inp — general input file to specify target atom and beam ion properties. This file includes four parameters: atomic nuclear charge Z of the target, the number of ionization states included in the model, nuclear charge of the beam ion, and the net charge of the beam ion.

(2) levels.inp — the same as that for ATTABLE;

(3) #.lev — the same as those for ATTABLE;

(4) #.iics — ion impact ionization cross sections for the target ion with # bound electrons. #.iics is in the same format as data file 'Xsecton.dat' of MICPSSR.

10.3. Output

The output file of BEAMTAB is 'beams.dat'. 'beams.dat' is in the format accepted by the CRE code. Data file 'beams.dat' along with 'atomic.dat', 'pixfit.dat', and 'ackf.table' form a complete *atomic model* for studying inner-shell emission spectra of ion beam heated plasmas.

11. CKFYED

11.1. Program Outline

CKFYED is a code for doing large-scale calculations for both configuration-averaged and term-dependent fluorescence yields. CKFYED does the calculation ion by ion. There are 14 I/O data files involved in running CKFYED. These files are listed in Table 17, along with the default logical unit numbers, names, types, and a brief description of their contents.

11.2. Input

To calculate fluorescence yield for an auotionizing state, one must specify the electronic configuration of the autoionizing state, all possible Auger and/or Coster-Kroning exit channels, and radiative decay channels. In CKFYED, all the radiationless and radiative exit channels are set up automatically. The user only needs to specify a list of autoionizing configurations and the corresponding 'hole' transitions.

The input for CKFYED consists of two files:

(1) yields.inp — a namelist input file containing calculation switches and parameters;

(2) ckf.state — a data file containing all autoionizing configurations to be considered. Further details of these two input files are described in Tables 18 and 19.

11.3. Output

The output of CKFYED consists of four data files:

(1) Auger_CK_fyield.tab — a readable data table which contains Auger rates, Coster-Kroning rates, radiative decay rates, and fluorescence yields for each LS term of all input autoionizing configurations.

(2) fyield_term.dat — a data file containing term-dependent fluorescence yields in the format accepted by computer code CKFTAB.

(3) fyield_config.dat — a data file containing configuration-averaged fluorescence yields in the format accepted by computer code CKFTAB.

(4) debug.out — debugging output file.

Since CKFYED is run ion by ion, every time one finishes running CKFYED for a specific ion, one should remember to copy data files (1), (2), and (3) to other files with specially designed names. When the calculations for all ions are done, all the 'fyield_term.dat' or 'fyield_config.dat' files should be combined together in one file 'fyield.dat' which is the input data file for code CKFTAB. As an example, a step by step guide for generating a K_α fluorescence yield data table for the carbon system is shown as follows.

STEP 1: edit 'yields.inp' by setting

Z = 6	—	atomic nuclear charge
klmno = 1	—	initial hole is in K-shell
nih = 1	—	principal quantum number of the K-shell
lih = 0	—	orbital quantum number of the K-shell
nfhole= 1	—	only consider 2p - 1s radiative transition, hence there is only one hole created by radiative transition
nfh(1)= 2	—	principal quantum number of the final hole in the 2p shell
lfh(1)= 1	—	orbital quantum number of the final hole in the 2p shell
jjcog = 0	—	only consider LS coupling.

STEP 2: figure out autoionizing configurations for each ion CI, CII, CIII ...

STEP 3: type in all the autoionizing configurations for one ion (e.g., CI).

STEP 4: run CKFYED

STEP 5: copy Auger_CK_fyield.tab to CI.ACKF.tab

copy fyield_term.dat to CI_term.dat

copy fyield_config.dat to CI_config.dat

Repeat step (2) to (5) for all ions

STEP 6: Combine CI_term.dat, CII_term.dat, ..., CV_term.dat together into a file C.fyield_term.dat.

Combine CI_config.dat, CII_config.dat, ..., CV_config.dat together into a file C.fyield_config.dat

Data file C.fyield_term.dat or C.fyield_config.dat is accepted by computer CKFTAB as input.

12. CKFTAB

12.1. Program Outline

CKFTAB does data organization to generate a data table for analyzing x-ray spectra. CKFTAB first reads all energy levels from data files #.lev which are the same as those for ATTABLE calculations. Then it searches for all autoionizing levels and assigns them with corresponding fluorescence yields.

12.2. Input

The inputs for CKFTAB include atomic energy levels and fluorescence yields. Hence, the input for CKFTAB consists of the following files:

(1) levels.inp — the same as that for ATTABLE;

(2) #.lev — the same as those for ATTABLE;

(3) fyield.dat — fluorescence yield data file.

12.3. Output

The output file of CKFTAB is 'ackf.table'. 'ackf.table' is in the format accepted by the CRE code. data file 'ackf.table' along with 'atomic.dat', 'pixfit.dat', and 'beams.dat' form a complete *atomic model* for studying inner-shell emission spectra of ion beam heating plasmas.

13. Running Codes in a Chain Manner

In order to generate an atomic model for plasma spectroscopy analysis and/or equation of state and opacity calculations, atomic data for atomic structure and different types of atomic processes are needed. These data are usually calculated by using different codes. Depending on what atomic model the user wants to create, the computer codes in the ATBASE package should be run in a chain in one of the following combinations:

(1) STATE/ATBASE/ATTABLE

(2) STATE/ATBASE/DWBORN/ATTABLE

(3) STATE/ATBASE/ATTABLE/CKYED/CKFTAB

(4) STATE/ATBASE/DWBORN/ATTABLE/CKYED/CKFTAB

(5) STATE/ATBASE/ATTABLE/MICPSSR/BEAMTAB/CKYED/CKFTAB

(6) STATE/ATBASE/DWBORN/ATTABLE/MICPSSR/BEAMTAB/CKYED/CKFTAB

Details of these combinations are described in this section.

13.1. Simple Atomic Model

To create a simple atomic model for analyzing emission/absorption thermal spectra for a plasma or calculating equations of state and opacities, the user should run program chain STATE/ATBASE/ATTABLE.

STEP 1: run STATE to generate CONFIG.INP and HYGEN_CFG for a specific ionization state;

STEP 2: run ATBASE to generate atomic energy levels, oscillator strengths and photoionization cross sections;

STEP 3: if the user wants an atomic model in configuration-average structure, then

copy ev_nl.dat to #.lev,

copy fv_nl.dat to #.fda,

copy photoionization_x to #.pda;

if the user wants an atomic model in LS term structure, then

copy ev_ls.dat to #.lev,

copy fv_ls.dat to #.fda,

copy photoionization_x to #.pda;

if the user wants an atomic model in fine structure, then

copy ev_final.dat to #.lev,

copy fv_final.dat to #.fda,

copy photoionization_x to #.pda,

where # is the number of bound electrons of the ion (e.g., #=6 for neutral carbon, #=5 for CII #=4 for CIII, etc.).

Repeat STEP (1), (2), and (3) for all ions of the element.

STEP 4: check all the #.lev files, writing down the number of energy levels of each ion to ATTABLE input file LEVELS.INP.

STEP 5: run ATTABLE to generate ATOMIC.DAT and PIXFIT.DAT.

The atomic model created in this way is a simple model which does not include electron collisional coupling for forbidden transitions. It should be noted that complete electron collisional coupling between levels (including both electric dipole allowed and forbidden transitions) is very important for the occupation of metastable levels. If one wants to do very detailed spectrum analysis for some specific lines, all related levels should be coupled in a complete manner.

13.2. Complete Coupling Atomic Model

To create a complete coupling atomic model for detailed spectroscopy analysis, one should run program chain STATE/ATBASE/DWBORN/ATTABLE.

STEP 1: run STATE to generate CONFIG.INP and HYGEN_CFG for a specific ionization state;

STEP 2: run ATBASE to generate atomic energy levels, oscillator strengths and photoionization cross sections;

STEP 3: if the user wants an atomic model in configuration-average structure, then

copy ev_nl.dat to #.lev,

copy fv_nl.dat to #.fda,

copy photoionization_x to #.pda,

copy wfunt_dat to #.wfunt;

if the user wants an atomic model in LS term structure, then

copy ev_ls.dat to #.lev,

copy fv_ls.dat to #.fda,

copy photoionization_x to #.pda,

copy wfunt_dat to #.wfunt;

if the user wants an atomic model in fine structure, then

copy ev_final.dat to #.lev,

copy fv_final.dat to #.fda,

copy photoionization_x to #.pda,

copy wfunt_dat to #.wfunt,

where # is the number of bound electrons of the ion (e.g., #=6 for neutral carbon, #=5 for CII #=4 for CIII, etc.).

Repeat STEP (1), (2), and (3) for all ions of the element.

STEP 4: Determine ionization states of your particular interest, then

copy #.lev to dwborn.sts,

copy #.wfunt to wfunt.dat,

where #.lev is atomic structure data table for the ion you would like to have complete coupling.

STEP 5: run DWBORN. Then copy streng.dat to #.streng

Repeat STEP (4) and (5) for all ions of interest.

STEP 6: Run ATTABLE to create ATOMIC.DAT and PIXFIT.DAT

13.3. Atomic Model for Analyzing X-ray Line Emission Spectra From High Energy Density Plasmas

The intensities of x-ray line emissions are determined by fluorescence yields instead of spontaneous decay rates. Hence, to create an atomic model for analyzing x-ray line emission spectra, one needs to provide fluorescence yields for all autoionizing levels. This can be done by running the program chain

STATE/ATBASE/ATTABLE/CKYED/CKFTAB

or

STATE/ATBASE/DWBORN/ATTABLE/CKYED/CKFTAB.

In this case, the atomic model consists of three data tables: ATOMIC.DAT, PIXIFT.DAT, and ACKF.TABLE.

13.4. Atomic Model for Analyzing X-ray Line Emission Spectra of Ion Beam Heated Plasmas

For ion beam heated plasmas, x-ray line emissions are produced by inner-shell ion impact ionizations. The atomic model should include inner-shell ion impact ionization processes. To create such an atomic model, the user should run program chain

STATE/ATBASE/ATTABLE/CKYED/CKFTAB/MICPSSR/BEAMTAB

or

STATE/ATBASE/DWBORN/ATTABLE/CKYED/CKFTAB/MICPSSR/BEAMTAB.

In this case, the atomic model consists of four data tables: ATOMIC.DAT, PIXIFT.DAT, ACKF.TABLE, and BEAMS.TAB.

Acknowledgment

Support for this work has been provided by the Forschungszentrum Karlsruhe through Fusion Power Associates.

References

1. P. Wang, "Computation and Application of Atomic Data for Inertial Confinement Fusion Plasmas," Ph.D. Thesis, Nuclear Engineering and Engineering Physics Department, University of Wisconsin, Madison, May (1991).

2. R. D. Cowan, "The Theory of Atomic Structure and Spectra," University of California Press, Los Angeles, 1981.

3. R. D. Cowan (private communication).

4. C. F. Fischer, Comput. Phys. Commun. **14**, 145 (1978).

5. A. Burgess, H. P. Summer, Mon. Not. R. Astr. Soc. **174**, 345 (1976).

6. A. Burgess, M. C. Chidicjimo, Mon. Not. R. Astr. Soc. **203**, 1269 (1983).

7. A. C. Mert, R. D. Cowan, and N. H. Magee, Los Alamos National Laboratory Report, LA-6220-MS (1976).

8. I. I. Sobelman, et al, "Excitation of Atoms and Broadening of Spectral Lines," Springer-Verlag, New York (1981).

9. W. Brandt and G. Lapicki, Phys. Rev. **A23**, 1717 (1981).

10. G. Lapicki and A.R. Zander, Phys. Rev. **A23**, 2072 (1981).

11. J.M. Hansteen and O.P. Hosebekk, J. Phys. Rev. Lett. **29**, 1361.

Table 1

ATBASE Data Files

Default Unit No.	Default Name (UNIX)	Type	Description
1	ev.dat	intermediate output	energy level data to be organized
2	tape2e	intermediate data	data transfer
3		SCRATCH	scratch file
4	fv.dat	intermediate output	oscillator strength data to be organized
5		SCRATCH	scratch file
7	autoii.rate	output	autoionization rates
9	outpt_lsj	output	complete output for spectrum calculations
10	input_lsj	intermediate input	constraint parameters for spectrum calculation. This file is automatically generated within the calculation.
11	outgine	intermediate data	data transfer
12		SCRATCH	scratch file
13		SCRATCH	scratch file
18	tape19	intermediate data	data transfer
19		SCRATCH	scratch file
20		SCRATCH	scratch file
21	atbase.inp	input	primary namelist input file
22	config.inp	input	configuration table input file
31	SCRATCH	SCRATCH	scratch file
32		SCRATCH	scratch file
41		SCRATCH	scratch file

Table 1 (Continued)

Default Unit No.	Default Name (UNIX)	Type	Description
50	input_scf	intermediate input	constraint parameters for SCF calculation. This file is automatically generated within the calculation.
51	outpt_scf	output	complete output for SCF calculations
52	param_scf	intermedate output	configuration-average structure data
53	wfunt_dat	output	radial wavefunctions in binary format
54	input_integl	intermediate input	constraint parameters for SCF calculation This file is automatically generated within the calculation.
55	outpt_integl	output	complete output for radial integral calculations
56	out2ing	intermediate data	data transfer
63	ev_lsj.dat	intermediate data	fine structure energy levels
64	ev_ls.dat	output	LS term structure energy levels
65	fv_lsj.dat	intermediate data	f-values for transitions between LSJ levels
66	fv_ls.dat	output	f-values for transitions between LS terms
67	ev_nl.dat	output	configuration-average energy levels
68	fv_nl.dat	output	f-values for transition arrays
69	hygen_cfg	input	configuration table for hydrogenic states
71	cfp.tab	input	fractional-parentage-coefficient table
72	tape72_cfp1	intermediate data	data transfer
73	tape73_cfp2	intermediate data	data transfer
74	tape74_cfp3	intermediate data	data transfer

Table 1 (Continued)

Default Unit No.	Default Name (UNIX)	Type	Description
75	ev_lsj.table	output	table for fine structure energy levels
76	ev_ls.table	output	table for LS term structure energy levels
77	fv_lsj.table	output	table for fine structure f-values
78	fv_ls.table	output	table for LS term structure f-values
79	ev_nl.table	output	table for configuration-average energy levels
80	photoionization_x	output	photoionization cross sections
81	config.fff	intermediate data	data transfer
82	wfc_out	intermediate output	continuum wavefunction calculation parameters
83	fv_nl.table	output	table for array f-values
90	ev_final.dat	output	energy level for user specifying structure
91	fv_final.dat	output	f-values for user specifying level structure
92	ev_final.tab	output	energy level table for user specifying level structure
93	fv_final.tab	output	f-value table for user specifying level structure
94	ev_nl.summary	output	configuration-average structure data for use in multiple ionization spectrum calculations
98		SCRATCH	scratch file
99	ev.expdat	input	experimental energy level data

Table 2
Detailed Description for atbase.inp

```
*                                                                       *
*  This is the primary input file for ATBASE, an atomic physics calculation code      *
*                                                                       *
&atbaseinp
*                                                                       *
*  (1)  switch parameters (on=1, off=0)
       iswich(1) = 1      ! f-values
       iswich(2) = 1      ! phoionization cross sections,=2:faster for high-Z
       iswich(3) = 0      ! autoionization rate
       iswich(4) = 0      ! electron impact excitation cross sections
       iswich(5) = 0      ! =1: UTA data only; =0: all; -1: UTA for a single shell
       iswich(6) = 0,     ! =1:incorporating NIST experimental data; =0:no
*
*
*  (2)  Self-consistent-field modle selections:
*                         ! 'imodl1(i)' is for primary configurations
       imodl1(1) = 0,     ! =0: non-relativistic calculation
*                         ! =1: semi-relativistic calculation
*
       imodl1(2) = 4,     ! =0: using Hartree modle
*                         ! =1: HS
*                         ! =2: HFS
*                         ! =3: HX
*                         ! =4: HF
*      imodl1(3) = 10,    ! =0: no correlation effect, corrf=0.0
*                         ! >0: include correlation effect corrf=imodl1(3)/10
*
*                         ! 'imodl2(i)' is for CI configurations
       imodl2(1) = 1,     !
       imodl2(2) = 4,     ! same as imodl1(i)
       imodl2(3) = 0,     ! if imodl2(3)=0 uses default setup
*
*
*  (3)  SCF convergence criterions
       tolscf = 5.0e-8,   ! maximum permissible value pf change in —RU— for ending SCF
       tole = 1.0e-10,    ! maximum permissible fractional change in the value of eigenvalues
       maxscf = 90,       ! maximum number of SCF interation cycles
*
```

Table 2 (Continued)

```
*   SCALING FACTORS FOR SLATER INTEGRALS
*
*   (4)   Scaling factors for slater integrals (to use default setup, set them equal 0 here)
          ifact(1) = 0,       ! Fk(ii)
          ifact(2) = 0,       ! zeta
          ifact(3) = 0,       ! Fk(ij)
          ifact(4) = 0,       ! Gk(ij)
          ifact(5) = 0        ! Rk
*
*   CONTROL PARAMAETERS FOR OSCILLATOR STRENGTH CALCULATIONS
*
*   (5)   oscillator strength calculation
*         imag          =0: do not calculate fv for magnetic-dipole transitions
*                       =1: calculate fv for M2 transitions of the first parity
*                       =2: calculate fv for M2 transitions of the second parity
*                       =3: calculate fv for M2 transitions of both parities
*
*         iquad         =0: do not calculate fv for electric quadrupole transitions
*                       =1: calculate fv for E4 transitions of the first parity
*                       =2:                 E4                  second
*                       =3:                 E4                  both
*
*         gfvmin: only those gfv > gfvmin will be output
*
*         dmin: delete spectrum lines for which S/X < dmin, an appropriate
*               value of dmin to delete weak lines is 0.005 to 0.05

          imag          = 0,
          iquad         = 0,
          gfvmin        = 5.0e-5,
          dmin          = 0.001
*
/
*   &end
*   _____ FINISH INPUT _____*
```

Table 3
Detailed Description for config.inp

(1) atomic number Z.
13

(2) charge state of the ion (0 = neutral, Z–1 = hydrogen-like ion)
7 — means Al^{+7}

(3) total number of configurations

(4) configurations
(CI effects are considered for each group of configurations)

```
*
1   1   3   1002   2002   2102
1   2   2   1002   2104
1   3   4   1002   2001   2102   3201
*
2   1   3   1002   2001   2103
2   3   3   1002   2103   3201
2   3   4   1002   2002   2101   3001
2   3   4   1002   2002   2101   3201
*
3   1   4   1002   2002   2101   3001
3   2   4   1002   2002   2101   4001
3   2   4   1002   2002   2101   5001
3   2   4   1002   2002   2101   6001
3   2   4   1002   2002   2101   7001
3   3   3   1002   2103   3001
3   3   3   1002   2103   4001
3   3   3   1002   2103   5001
3   3   3   1002   2103   6001
3   3   3   1002   2103   7001
*
4   1   4   1002   2002   2101   3101
4   2   4   1002   2002   2101   4101
4   2   4   1002   2002   2101   5101
4   2   4   1002   2002   2101   6101
4   2   4   1002   2002   2101   7101
4   3   3   1002   2103   3101
4   3   3   1002   2103   4101
4   3   3   1002   2103   5101
4   3   3   1002   2103   6101
4   3   3   1002   2103   7101
```

Table 3 (Continued)

```
*
5   1   4   1002   2002   2101   3201
5   2   4   1002   2002   2101   4201
5   2   4   1002   2002   2101   5201
5   2   4   1002   2002   2101   6201
5   2   4   1002   2002   2101   7201
5   3   3   1002   2103   3201
5   3   3   1002   2103   4201
5   3   3   1002   2103   5201
5   3   3   1002   2103   6201
5   3   3   1002   2103   7201
*
6   1   4   1002   2002   2101   4301
6   2   4   1002   2002   2101   5301
6   2   4   1002   2002   2101   6301
6   1   4   1002   2002   2101   7301
6   3   3   1002   2103   4301
6   3   3   1002   2103   5301
6   3   3   1002   2103   6301
6   3   3   1002   2103   7301
*
(5) inner shell excited configurations
*
1   1   3   1001   2002   2103
1   2   2   1001   2105
1   3   4   1001   2001   2103   3001
1   3   4   1001   2001   2103   3201
1   2   4   1001   2002   2102   3101
*
2   1   3   1001   2001   2104
2   2   4   1001   2002   2102   3001
2   2   4   1001   2002   2102   3201
2   3   3   1001   2104   3001
2   3   3   1001   2104   3201
2   3   4   1001   2001   2103   3101
```

Table 4
Detailed Explanation of ev.expdat

```
*
*    ******************** experimental energy levels of He I ********************
*                                  (in the unit of 1/cm)
*
*    (1) total number of levels
14
*
*    (2) level index, # of shells, electronic configuration
*    (3) 2Sp+1, Lp, 2S+1, L, J, E(LSJ)
*
1   1   1002
1     0    1    0    0.0    0.00
*
2    2   1001   2001
2     0    1    0    0.0    166271.70
*
3    2   1001   3001
2     0    1    0    0.0    184859.06
*
4    2   1001   4001
2     0    1    0    0.0    190934.50
*
5   2   1001   5001
2     0    1    0    0.0    193657.78
*
6   2   1001   2101
2     0    1    1    1.0    171129.148
```

Table 4 (Continued)

```
*
7   2   1001   3101
2   0   1   1   1.0   186203.62
*
8   2   1001   4101
2   0   1   1   1.0   191486.95
*
9   2   1001   5101
2   0   1   1   1.0   193936.75
*
10   2   1001   3201
2   0   1   2   2.0   186099.22
*
11   2   1001   4201
2   0   1   2   2.0   191440.71
*
12   2   1001   5201
2   0   1   2   2.0   193912.54
*
13   2   1001   4301
2   0   1   3   3.0   191447.24
*
14   2   1001   5301
2   0   1   3   3.0   193914.31
*
*
* end
```

Table 5
Kinetic Energy Mesh of the Ejected Electron for Photoionization Cross Section Calculation

1.000E-02	1.778E-02	3.162E-02	5.623E-02	1.000E-01	1.166E-01	1.359E-01	1.585E-01	1.848E-01	2.154E-01
2.512E-01	2.929E-01	3.415E-01	3.981E-01	4.642E-01	5.412E-01	6.310E-01	7.356E-01	8.577E-01	1.000E+00
1.166E+00	1.359E+00	1.585E+00	1.848E+00	2.154E+00	2.512E+00	2.929E+00	3.415E+00	3.981E+00	4.642E+00
5.412E+00	6.310E+00	7.356E+00	8.577E+00	1.000E+01	1.259E+01	1.585E+01	1.995E+01	2.512E+01	3.162E+01
3.981E+01	5.012E+01	6.310E+01	7.943E+01	1.000E+02	1.380E+02	1.904E+02	2.627E+02	3.624E+02	5.000E+02

This dimensionless kinetic energy mesh is used universally for all photoionization cross section calculations. The corresponding photon energy mesh is determined from the following relations:

$$h\nu(i) = I_{nl}(1 + e_k(i)) \text{ if } I_{nl} \leq 2$$

$$h\nu(i) = I_{nl}(1 + e_k(i)/2) \text{ if } I_{nl} > 2$$

where, I_{nl} is binding energy of the nl subshell, $e_k(i)$ is the ith mesh point value in above table.

Table 6
RUN_ATBASE

```
# This is a shell command file containing a list of shell commands.
# These shell commands control the execution process and movement of relevant I/O files
#
# Before typing 'sh RUN_ATBASE', check the following:
# (1) is xatdata.exe in the current working directory ?
# (2) is cfp.tab in the current working directory ?
# (3) what is the path to access all the #.sts files ?
# (4) where do you want to store the output files ?
#
# In this example, we assume that all the #.sts files are in the directory
#       $HOME/ATBASE/DATA/Al,
# and we want to store all the output in the same directory.
#
# * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
#
# ··· (I) He-like Al
# cp -f $HOME/ATBASE/DATA/Al/2.sts       config.inp
./xatdata.exe > jobout

 cp -f ev.raw          $HOME/ATBASE/DATA/Al/2.ev.raw
 cp -f fv.raw          $HOME/ATBASE/DATA/Al/2.fv.raw
 cp -f ev_nl.raw       $HOME/ATBASE/DATA/Al/2.ev_nl.raw
 cp -f pbf_x.dat       $HOME/ATBASE/DATA/Al/2.pda
 cp -f fv_LSJ.tab      $HOME/ATBASE/DATA/Al/2.fv_LSJ.tab
 cp -f fv_LS.tab       $HOME/ATBASE/DATA/Al/2.fv_LS.tab
 cp -f aai_LSJ.dat     $HOME/ATBASE/DATA/Al/2.aai_LSJ.dat
 cp -f aai_LS.tab      $HOME/ATBASE/DATA/Al/2.aai_LS.tab
 cp -f aai_LSJ.tab     $HOME/ATBASE/DATA/Al/2.aai_LSJ.tab
 cp -f ebbx.raw        $HOME/ATBASE/DATA/Al/2.ebbx.raw
 cp -f ebbx_LS.tab     $HOME/ATBASE/DATA/Al/2.ebbx_LS.tab
 cp -f ebbx_LSJ.tab    $HOME/ATBASE/DATA/Al/2.ebbx_LSJ.tab
```

Table 6 (Continued)

```
#
# ··· (II) Li-like Al
#
cp -f $HOME/ATBASE/DATA/Al/3.sts       config.inp
./xatdata.exe > jobout

  cp -f ev.raw          $HOME/ATBASE/DATA/Al/3.ev.raw
  cp -f fv.raw          $HOME/ATBASE/DATA/Al/3.fv.raw
  cp -f ev_nl.raw       $HOME/ATBASE/DATA/Al/3.ev_nl.raw
  cp -f pbf_x.dat       $HOME/ATBASE/DATA/Al/3.pda
  cp -f fv_LSJ.tab      $HOME/ATBASE/DATA/Al/3.fv_LSJ.tab
  cp -f fv_LS.tab       $HOME/ATBASE/DATA/Al/3.fv_LS.tab
  cp -f aai_LSJ.dat     $HOME/ATBASE/DATA/Al/3.aai_LSJ.dat
  cp -f aai_LS.tab      $HOME/ATBASE/DATA/Al/3.aai_LS.tab
  cp -f aai_LSJ.tab     $HOME/ATBASE/DATA/Al/3.aai_LSJ.tab
  cp -f ebbx.raw        $HOME/ATBASE/DATA/Al/3.ebbx.raw
  cp -f ebbx_LS.tab     $HOME/ATBASE/DATA/Al/3.ebbx_LS.tab
  cp -f ebbx_LSJ.tab    $HOME/ATBASE/DATA/Al/3.ebbx_LSJ.tab

#
#
# ··· (III) Be -like Al
#
cp -f $HOME/ATBASE/DATA/Al/4.sts       config.inp
./xatdata.exe > jobout

  cp -f ev.raw          $HOME/ATBASE/DATA/Al/4.ev.raw
  cp -f fv.raw          $HOME/ATBASE/DATA/Al/4.fv.raw
  cp -f ev_nl.raw       $HOME/ATBASE/DATA/Al/4.ev_nl.raw
  cp -f pbf_x.dat       $HOME/ATBASE/DATA/Al/4.pda
  cp -f fv_LSJ.tab      $HOME/ATBASE/DATA/Al/4.fv_LSJ.tab
  cp -f fv_LS.tab       $HOME/ATBASE/DATA/Al/4.fv_LS.tab
  cp -f aai_LSJ.dat     $HOME/ATBASE/DATA/Al/4.aai_LSJ.dat
  cp -f aai_LS.tab      $HOME/ATBASE/DATA/Al/4.aai_LS.tab
  cp -f aai_LSJ.tab     $HOME/ATBASE/DATA/Al/4.aai_LSJ.tab
  cp -f ebbx.raw        $HOME/ATBASE/DATA/Al/4.ebbx.raw
  cp -f ebbx_LS.tab     $HOME/ATBASE/DATA/Al/4.ebbx_LS.tab
  cp -f ebbx_LSJ.tab    $HOME/ATBASE/DATA/Al/4.ebbx_LSJ.tab
```

Table 6 (Continued)

```
#
# ··· (IV) B-like Al
#
cp -f $HOME/ATBASE/DATA/Al/5.sts       config.inp
./xatdata.exe > jobout

 cp -f ev.raw          $HOME/ATBASE/DATA/Al/5.ev.raw
 cp -f fv.raw          $HOME/ATBASE/DATA/Al/5.fv.raw
 cp -f ev_nl.raw       $HOME/ATBASE/DATA/Al/5.ev_nl.raw
 cp -f pbf_x.dat       $HOME/ATBASE/DATA/Al/5.pda
 cp -f fv_LSJ.tab      $HOME/ATBASE/DATA/Al/5.fv_LSJ.tab
 cp -f fv_LS.tab       $HOME/ATBASE/DATA/Al/5.fv_LS.tab
 cp -f aai_LSJ.dat     $HOME/ATBASE/DATA/Al/5.aai_LSJ.dat
 cp -f aai_LS.tab      $HOME/ATBASE/DATA/Al/5.aai_LS.tab
 cp -f aai_LSJ.tab     $HOME/ATBASE/DATA/Al/5.aai_LSJ.tab
 cp -f ebbx.raw        $HOME/ATBASE/DATA/Al/5.ebbx.raw
 cp -f ebbx_LS.tab     $HOME/ATBASE/DATA/Al/5.ebbx_LS.tab
 cp -f ebbx_LSJ.tab    $HOME/ATBASE/DATA/Al/5.ebbx_LSJ.tab
.
.
```

Table 7
MODEL_STRUCTURE
INPUT FILE FOR SETTING UP ATOMIC MODEL

```
*       (1)    iinnp=1, using interactive input
*                    =2, unsing this file to specify parameter
 2
*       ***************************************************************
*
*       (2)    specify atomic charge number Z
*
 11
*
*       (3)    specify fine structure boundary: nf,lf (for configurations with outer shell's (nl) equal
*              or smaller than the following numbers, levels are fine-structure resolved. If no fine-
*              structure levels, input: 0 0 )
*
 0 0
*
*       (4)    specify LS term boundary: nt,lt (for configurations with outer shell's (nl) equal
*              or smaller than the following number, levels are LS term-structure resolved. If no
*              fine-structure levels, input: 0 0 )
*
 4 3
*
*       (5)    configuration average structure boundary: nc (for configurations with outer shell's
*              principal quantum number equal or smaller than the following number, levels are in
*              configuration average structures)
*
 7
*
*       (6)    maximum principal quantum number: nmax only those levels with n <= nmax will
*              be considered in the model
*
 7
*
*       (7)    include inner-shell excited state? 1=yes, 0=no
 1
*
*       (8)    compute Auger rates and fluorescence yield? 1=yes, 0=no
 1
*
*       end
*       ***************************************************************
```

Table 8
RUN_ATMODEL

```
# This is a batch command file for runing atmodel
#
# step 0: compile ATMODEL.f to create an executable 'xatm'
#
# step 1: get all related raw data file from specific directory
#
# step 2: copy the specific structure file to → model.structure
#
# step 3: run the code
#
# step 4: move the output to the directory where you store the model data
#
# note: you can repeat above steps for all ions
#
# * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
# (1) He -like system
#
 cp -f /path/2.ev.raw          ev.raw
 cp -f /path/2.fv.raw          fv.raw
 cp -f /path/2.ev_nl.raw       ev_nl.raw
 cp -f /path/2.aai_LSJ.dat     aai_LSJ.dat

./xatm

 mv fv_mix.tab            /path/2.fvtab
 mv aai_mix.tab           /path/2.aitab
 mv fyi_mix.tab           /path/2.fytab
 mv EV_MODEL.dat          /path/2.lev
 mv FV_MODEL.dat          /path/2.fda
 mv AAI_MODEL.dat         /path/2.aai
 mv FYI_MODEL.dat         /path/2.fyi

#
# * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Table 8 (Continued)

```
#
# (2) Li -like system
#   cp -f /path/3.ev.raw        ev.raw
    cp -f /path/3.fv.raw        fv.raw
    cp -f /path/3.ev_nl.raw     ev_nl.raw
    cp -f /path/3.aai_LSJ.dat   aai_LSJ.dat

./xatm

 mv fv_mix.tab        /path/3.fvtab
 mv aai_mix.tab       /path/3.aitab
 mv fyi_mix.tab       /path/3.fytab
 mv EV_MODEL.dat      /path/3.lev
 mv FV_MODEL.dat      /path/3.fda
 mv AAI_MODEL.dat     /path/3.aai
 mv FYI_MODEL.dat     /path/3.fyi

# ***********************************************************
#
# (3) Be -like system
#
 cp -f /path/4.ev.raw        ev.raw
 cp -f /path/4.fv.raw        fv.raw
 cp -f /path/4.ev_nl.raw     ev_nl.raw
 cp -f /path/4.aai_LSJ.dat   aai_LSJ.dat

./xatm

 mv fv_mix.tab        /path/4.fvtab
 mv aai_mix.tab       /path/4.aitab
 mv fyi_mix.tab       /path/4.fytab
 mv EV_MODEL.dat      /path/4.lev
 mv FV_MODEL.dat      /path/4.fda
 mv AAI_MODEL.dat     /path/4.aai
 mv FYI_MODEL.dat     /path/4.fyi

# ***********************************************************
```

Table 8 (Continued)

```
#
# (4) B -like system
#
 cp -f /path/5.ev.raw          ev.raw
 cp -f /path/5.fv.raw          fv.raw
 cp -f /path/5.ev_nl.raw       ev_nl.raw
 cp -f /path/5.aai_LSJ.dat     aai_LSJ.dat

./xatm

 mv fv_mix.tab            /path/5.fvtab
 mv aai_mix.tab           /path/5.aitab
 mv fyi_mix.tab           /path/5.fytab
 mv EV_MODEL.dat          /path/5.lev
 mv FV_MODEL.dat          /path/5.fda
 mv AAI_MODEL.dat         /path/5.aai
 mv FYI_MODEL.dat         /path/5.fyi

# ********************************************************************
#
# (5) C -like system
#
 cp -f /path/6.ev.raw          ev.raw
 cp -f /path/6.fv.raw          fv.raw
 cp -f /path/6.ev_nl.raw       ev_nl.raw
 cp -f /path/6.aai_LSJ.dat     aai_LSJ.dat

./xatm

 mv fv_mix.tab            /path/6.fvtab
 mv aai_mix.tab           /path/6.aitab
 mv fyi_mix.tab           /path/6.fytab
 mv EV_MODEL.dat          /path/6.lev
 mv FV_MODEL.dat          /path/6.fda
 mv AAI_MODEL.dat         /path/6.aai
 mv FYI_MODEL.dat         /path/6.fyi

# ********************************************************************
```

Table 8 (Continued)

```
#
# (6) N -like system
#
 cp -f /path/7.ev.raw          ev.raw
 cp -f /path/7.fv.raw          fv.raw
 cp -f /path/7.ev_nl.raw       ev_nl.raw
 cp -f /path/7.aai_LSJ.dat     aai_LSJ.dat

./xatm


 mv fv_mix.tab          /path/7.fvtab
 mv aai_mix.tab         /path/7.aitab
 mv fyi_mix.tab         /path/7.fytab
 mv EV_MODEL.dat        /path/7.lev
 mv FV_MODEL.dat        /path/7.fda
 mv AAI_MODEL.dat       /path/7.aai
 mv FYI_MODEL.dat       /path/7.fyi

# * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
#
# (7) O -like system
#
 cp -f /path/8.ev.raw          ev.raw
 cp -f /path/8.fv.raw          fv.raw
 cp -f /path/8.ev_nl.raw       ev_nl.raw
 cp -f /path/8.aai_LSJ.dat     aai_LSJ.dat

./xatm

 mv fv_mix.tab          /path/8.fvtab
 mv aai_mix.tab         /path/8.aitab
 mv fyi_mix.tab         /path/8.fytab
 mv EV_MODEL.dat        /path/8.lev
 mv FV_MODEL.dat        /path/8.fda
 mv AAI_MODEL.dat       /path/8.aai
 mv FYI_MODEL.dat       /path/8.fyi

: : :
# * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Table 9
Input/Output Files in ATTABLE

Default Unit No.	Default Name (UNIX)	Type	Description
1	attable.inp	input	namelist input file for ATTABLE
2	levels.inp	input	level index for the atomic model
3	atomic.dat	output	atomic model data table
4	pixfit.dat	output	photoionization cross sections
11 – 10+N*	#.lev	input	atomic structure data output from ATBASE calcualtions
10+N+1 – 10+2*N	#.fda	input	oscillator strengths
10+2*N+1 – 10+3*N	#.pda	input	photoionization cross sections
10+3*N+1 – 10+4*N	#.streng	input	collision strengths

* Note: Here N is the total number of ions included in the calculation. For example, if one only includes the first 20 ionization states of Au (Au^{+0} – Au^{+19}) in the calculation, then N = 20, unit 11 for 79.lev, unit 12 for 78.lev, unit 30 for 59.lev, unit 31 for 79.fda, unit 51 for 79.pda, unit 71 for 79.streng, etc.

Table 10
NAMELIST INPUT FILE FOR <<<<<<<ATTABLE>>>>>>>

```
&datainput
 *       =============== (1) atomic system specification ===============
 *
         Z        = 13,          ! atomic nuclear charge
 *
         nzstar   = 14,          ! number of charge states to be included
 *
         minzs    =0,            ! minimum charge state (neutral=0)
 *
         maxzs    =13,           ! maximum charge state (fully ionized=Z)

 *       =============== (2) option switchs ========================

         jpick    =0,            ! level selection option:
 *                               ! jpick=0, automaticly picking all
 *                               ! jpick=1, input from 'levels.inp'
 *
         irate    = 1,           ! irate=0, no rate coeff. are computed
 *                               !       =1, compute all rate coeff.
 *                               !
         iphox    = 0,           ! iphox=0, using old fitting formula for photoionization X,
                                 ! iphox=1, using 10 data points, should be used in high-Z cases
                                 !
         ioldf    = 0,           !       =0, old oscillator strength output format
 *                               !       =1, new format
 *
         iauto    = 1,           !       =0, no output of autoionization rates
 *                               !       =1, include output of autoionization rates
 *
 *       =============== (3) Temperature and density grids ==============
 *
         te0      =0.1,          ! minimum temperature (eV)
         ted      =3000,         ! maximum temperature (eV)
         ntemp    = 10,          ! number of temperature grid points (recommand 10)
         dlogte   = 0.555556,    ! log10 DT, if ted>te0, this quantity will
                                 ! be calculated within the code
 *
         de0      = 1.0e14,      ! minimum plasma ion density (1/cm**3)
         def      = 1.0e24,      ! maximum plasma ion density (1/cm**3)
         nden     = 4            ! number of density grid points (recommend 4)
 * /
 &end
 *       =============== THE END ============================
```

Table 11
Detailed Description of levels.inp

This file should be consistent with the parameter NZSTAR in attable.inp. There must be NZSTAR pair numbers in this file, each specifying the ion and the number of levels of the ion.

The first column: charge state of the ion #

The second column: number of levels for the ion #. This number can be found by checking the level index in data file #.lev.

0	25	— 0: neutral atom; 25: 25 levels are considered for the neutral atom
1	10	— 1: first ionization state; 10: 10 levels are considered for this ion
2	17	— 2: second ionization state;17: 17 levels are considered for this ion
:	:	:
:	:	:
nzstar-1	1	— The last ionization state to be considered.

Table 12
DWBORN Data Files

Default Unit No.	Default Name (UNIX)	Type	Description
1	dwnorn.inp	input	namelist input file for dwborn
2		scratch	scratch file
3	dwborn.sts	input	atomic state input file
4	debug.out	output	debugging output file
7	eaout.rate	output	rate coefficients and average collision strength for electron impact excitation
8	eaout.read	output	cross sections and collision strengths
9	eaout.plot	output	cross sections and collision strengths output for plotting x – incident electron energy y1 – cross section y2 – collision strength
10	streng.dat	output	collision strength in the format accepted by ATTABLE
11		scratch	scratch file
12		scratch	scratch file
13		scratch	scratch file
14		scratch	scratch file
15		scratch	scratch file

Table 13
Detailed Explanation of dwborn.inp

```
*
*       ************************************************************
*
*                                   DWBORN.INP
*       This is an input file for computer code DWBORN. DWBORN is a code for computing
*       electron impact excitation cross sections with distorted-wave Born method.
*
$bornin
*
*       ..............................target ion information ..............................
*
        element='Al',     ! target ion ID
        z = 13,           ! nuclear charge
        nee =3,           ! total number of bound electrons
*
*       ..............................incident electron energy ..............................
*
        units=1,          ! energy units: 1=2Ry, 2=eV
        ekmin = 0.001,    ! minimum energy
        ekmax = 100,      ! maximum energy
        kepoit = 15,      ! number of grid points in energy mesh
                          ! if kepoit=25, default Gaussian nodes
*
*       ..................................control switches ..................................
*
        contrl(1)=2,      ! =1: Born approximation (no exchange)
                          ! =2: distorted wave approximation (with exchange)
        contrl(2)=1,      ! =1: output in energy unit
                          ! =2: output in threshold unit E/DE

        contrl(3)=2,      ! =1: on-line calculate Hartree-Fock wavefunctions
                          ! =2: read in wavefunctions from supplied data file

        lspdf=6,          ! number of partial waves to be included
        seedbg=0,         ! debug output (0=no, 1=yes )
*
*       ....................temperature range for maxwellian average ....................
*
        temin = 10,       ! minimum temperature (eV)
        temax = 500,      ! maximum temperature (eV)

$ end
```

Table 14
Detailed Explanation of dwborn.sts

3	Number of levels to be considered
1 2 1001 2002	level index; number of shell; electronic configuration 1: the first level; 2: two shells; $1s^1 2s^2$
2 0 2	coupling sequence, always input number of shell, 0, number of shell
1 1 2 0 1 1 .5 -1.20397751E+02	left to right LS coupling scheme 0 1 2 0 1 2 J value; energy of the level E. If only considering LS term structure, set J = -1 if energy is not known, set E=0, program will do calculation to determine E.
2 3 100 2001 2101 3 0 3 1 1 2 1 1 2 1 3 2 0 1 2 0 1 3 0 3 4 .5 -1.20195938E+02	The second level
3 3 1001 2001 2101 3 0 3 1 1 2 1 1 2 1 3 2 0 1 2 0 1 3 0 3 4 1.5 -1.20185051E+02	The third level

Table 15
MICPSSR Data Files

Default Unit No.	Default Name (UNIX)	Type	Description
1	mi_cpssr.inp	input	namelist input file
2		SCRATCH	scratch file
3	Xsecton.tab	output	a readable data table for ion impact ionization cross sections
6	debug.out	output	debugging output file
13	Xsecton.dat	output	data table for ion impact ionization cross sections in the format accepted by BEAMTAB.

Table 16
Detailed Description of mi_cpssr.inp

```
*   ***************************************************
*                      NAMELIST INPUT FILE
*                            FOR
*          ION IMPACT IONIZATION CROSS SECTION CALCULATION
*                          direct ionization,
*                          electron capture,
*                         multiple ionization
*                    (plane-wave-Born model + CPSSR)
*   ***************************************************
*   &inputii
*
*   . . . . . (1) target atom information
*
    element='Al',          ! target ion ID
    z = 13,                ! target nuclear charge
    nee = 4,               ! number of bound electrons
    mass = 27,             ! target atomic weight
*
*   . . . . . (2) target state (electron configuration)
*               default setup for OI, AlI, ClI, ArI, and AuI
*
    ideft = 0,             ! =1 using default configuration for neutral atom
    shelln = 3,            ! number of subshells of the configuration
    nle(1) = 1002, 2001,   ! 1s2, 2s2
    nle(3) = 2101, 3002,   ! 2p6, 3s2
    nle(5) = 3101, 3205,   ! 3p6, 3d10 ···
    nle(7) = 4002, 4106,   !
    nle(9) = 4210, 4314,   ! 1000*n + 100*l + e
    nle(11)= 5002, 5106,   !
    nle(13)= 5210, 6001,   !
*
*   . . . . . (3) ionized shell information (up to 5 subshells)
*
    nid(1)=1,2,2           ! principal quantum # (n) of the ionized shell
    lid(1)=0,0,1           ! orbital quantum # (l) of the ionized shell
    vj(1) =-1,-1,-1,       ! j value of the ionized shell (for ls coupling
*                          ! set vj=-1)
```

Table 16 (Continued)

```
*
*   . . . . . (4) incident ion information
*
    massint = 7,     ! atomic weight
    zinct = 3,       ! nuclear charge of the projectile
    qinct = 3,       ! net charge of the projectile
*
*   . . . . . (5) beam energy
*
    units=2,         ! unit of energy (1=2Ry, 2=eV)
    ekmin=9.0e6,     ! minimum beam energy
    ekmax=5.0e6,     ! maximum beam energy
*
*
*   . . . . . (6) calculation parameter
*
    lspdf =4,        ! # of partial wave included (suggested value=4)
    kepoit=1,        ! energy mesh point used for the calculation
    seedbg=0,        ! debug switch: 1=on, 0=off
*
*
*   . . . . . (7) control switches
*
    contrl(1)=1,     ! contrl(1): binding & polarization effects (0=no,1=yes)
    contrl(2)=1,     ! contrl(2): Coulomb deflection effect (0=no,1=yes)
    contrl(3)=1,     ! contrl(3): relativistic corrections (0=no,1=yes)
    contrl(4)=1,     ! contrl(4): calculate multiple ionization X (0=no,1=yes)
    contrl(5)=1,     ! contrl(5): calculate electron capture X (0=no,1=yes)
&end
```

Table 17
CKFYED Data Files

Default Unit No.	Default Name (UNIX)	Type	Descriptions
1	yields.inp	input	namelist input file n
2		SCRATCH	scratch file
3	ckf.state	input	autoionizing configurations to be considered
4	Auger.states	intermediate	all autoionizing terms generated by code
5	debug.out	output	debugging output
8	ls.terms	not used	
10		SCRATCH	scratch file
11		SCRATCH	scratch file
12		SCRATCH	scratch file
13		SCRATCH	scratch file
14		SCRATCH	scratch file
90	Auger_CK_fyield.tab	output	readable table for Auger rates, Coster-Kroning rates and fluorescence yields
91	fyield_config.dat	output	configuration averaged fluorescence yields
92	fyield_term.dat	output	term-dependent fluorescence yields

Table 18
Description of yields.inp

```
*
*   ***************************************************
*
*   This is an input file for C-K and fluorescence yield calculation
*
*   ***************************************************
*
*
&ckfinp
    seedbg = 0,                         ! debug output switch (0=no, 1=yes)
*   ***************************************************
*                           atomic system
*   ***************************************************
    Z = 13,                             ! atomic nuclear charge
*   ***************************************************
*                           hole transitions
*   ***************************************************
    klmno = 1,                          ! shell index of the initial hole (K=1,L=2...)
    nih = 1,                            ! principal quantum # of the initial hole
    lih = 0,                            ! orbital quantum # of the initial hole

    nfhole = 2,                         ! # of final holes produced by radiative tran.
    nfh(1) = 2,3,                       ! principal quantum # of the final hole
    lfh(1) = 1,1,                       ! orbital quantum # of the final hole
*   ***************************************************
*                           jj-coupling configuration
*   ***************************************************
    jjcog = 0                           ! jj-coupling switch (0=no, 1=yes)
```

Table 18 (Continued)

```
*
*   If one is only interested in LS coupling system, ignore the following.
*   The following parameters are set particularly for Au system.
*   ---------------------------------------------------------------------
    issr = 22           ! # of shell of the jj-coupling config.
*   ··· electron configuration
    nler(1) = 1002,                                          ! K shell
    nler(2) = 2002,2102,2104,                                ! L shell
    nler(5) = 3002,3102,3104,3204,3206,                      ! M shell
    nler(10)= 4002,4102,4104,4204,4206,4306,4308,            ! N shell
    nler(17)= 5002,5102,5104,5204,5206,                      ! O shell
    nler(22)= 6001,                                          ! P shell

*   ··· binding energies
    ebr(1) = 2998.63,
    ebr(2) = 540.85, 517.66, 449.37,
    ebr(5) = 131.20, 121.44, 105.97, 89.68, 86.31,
    ebr(10) = 30.44, 26.11, 22.32, 15.12, 14.39, 4.87, 4.71,
    ebr(17) = 5.33, 3.81, 3.15, 0.982, 0.907,
    ebr(22) = 0.626,

*   ··· ratios of degeneracy to the non-relativistic shell
*   for example: L1 subshell=1, L2 subshell=0.33, L3 subshell=0.67)
    facr(1) = 1,
    facr(2) = 1, 0.33, 0.67,
    facr(5) = 1, 0.33, 0.67, 0.4, 0.6,
    facr(10)= 1, 0.33, 0.67, 0.4, 0.6, 0.43, 0.57,
    facr(17)= 1, 0.33, 0.67, 0.4, 0.6,
    facr(22)= 1,

*   shell index of the initial 'hole'
    isidr = 0

*   end of the input file

*   &end
```

Table 19
Description of ckf.state

```
*  ********************************************
*  This is an input file for CKF calculations
*  This file includes all the autoionizing configurations to be calculated
*  ********************************************
*
*  (1) number of the configurations to be considered
   4
*  (2) configurations : (# of shells), (1000*n + 100**l + e)
*  F-like Al
   3 1001 2002 2106          ! 3 shells, 1s^1 2s^2 2p^6
   4 1001 2002 2105 3001     ! 4 shells, 1s^1 2s^2 2p^5 3s^1
   4 1001 2002 2105 3101     ! 4 shells, 1s^1 2s^2 2p^5 3p^1
   4 1001 2002 2105 3201     ! 4 shells, 1s^1 2s^2 2p^5 3d^1
```

Note: Lines with $*$ in the first column are comment lines.