AJAC: Atomic data calculation tool in Python

Amani Tahat[a)†], Jordi Marti[a)], Kaher Tahat[b)], and Ali Khwaldeh[c)]

[a)] *Department of Physics and Nuclear Engineering, Technical University of Catalonia, Barcelona, Spain*
[b)] *Saudi Grown Investment Company, Quai du Seujet 30, Geneva 1201, Switzerland*
[c)] *Department of Computer Engineering, Faculty of Engineering, Philadelphia University, Amman, Jordan*

(Received 11 August 2012; revised manuscript received 13 September 2012)

In this work, new features and extensions of a currently used online atomic database management system are reported. A multiplatform flexible computation package is added to the present system, to allow the calculation of various atomic radiative and collisional processes, based on simplifying the use of some existing atomic codes adopted from the literature. The interaction between users and data is facilitated by a rather extensive Python graphical user interface working online and could be installed in personal computers of different classes. In particular, this study gives an overview of the use of one model of the package models (i.e., electron impact collisional excitation model). The accuracy of computing capability of the electron impact collisional excitation in the adopted model, which follows the distorted wave approximation approach, is enhanced by implementing the Dirac *R*-matrix approximation approach. The validity and utility of this approach are presented through a comparison of the current computed results with earlier available theoretical and experimental results. Finally, the source code is made available under the general public license and being distributed freely in the hope that it will be useful to a wide community of laboratory and astrophysical plasma diagnostics.

Keywords: software engineering, atomic data, atomic databases

PACS: 89.20.–a, 89.20.Bb, 89.20.Ff, 31.15.–p **DOI:** 10.1088/1674-1056/22/4/048901

1. Introduction

Since experimental values for most of the atomic data (e.g., energy level, radiative rates, and electron impact excitation rates) are very difficult to obtain from experiments,[1,2] one has to rely on data generated from theoretical models. In principle, results from theoretical models must be close to experimental values to ensure that the models are successful, in such a way that calculations with very similar models should render close results. In the meantime, if theoretical models were different, the results obtained from them might be completely different as well. However, for many required atomic parameters large discrepancies are often observed among different yet comparable calculations, and therefore it is difficult for the researcher to use any data and to determine which data to use. For application accuracy is recommended so that atomic data may be confidently applied. Currently, resourcing atomic data is a challenge and needs to be improved to meet users? needs. In the literature, atomic data are often spread over a wide range of journals and this makes the task of finding the required data difficult for a user, apart from the problems of the assessment of the data as discussed in Ref. [2]. Additionally, due to a significant increase in storage power of computers and rapid development of computing techniques during the past decade or so, a typical calculation generates so much data that no journal can publish it entirely, although some do publish in their electronic versions but in a difficult way for data mining. To overcome this difficulty, a few websites store a significant quantity of atomic data for a large range of ions under a variety of atomic parameters that are required for the modeling of plasmas (e.g, energy level, radiative rates, lifetimes) and case studies such as the widely used websites especially for collisional data: CHIANTI [http://chianti.nrl.navy.mil/] and ADAS [http://open.adas.ac.uk]. These websites also provide a wide range of software for producing the atomic data, and hence are widely used by those who generate, assess, and/or apply the atomic data. Consequently, these websites and repositories of data may have some disadvantages and become difficult to use. i.e., the data may be incomplete, may not be assessed by independent experts, and are often unpublished so that the data available on a website may not at all times be good compared with the data available in the literature and the user still needs to verify them. The other problem is that updating and revising the website must be done continuously, and since this is staff-intensive work, there could be a considerable delay before the published data are incorporated into the databank. Therefore, for an active researcher who wants to use the latest and/or the best available atomic data, there is no choice except to search for the data themselves. Herein, and as an attempt at providing possible solutions for these problems, we have recently created and published online the atomic database management system DBMS[3] [http://aims-thinkenergy.org/amani/index.lua]. Supported with several atomic physics packages and other software related to using different theoretical approaches, the DBMS is still under construction, and this work has come to supersede the previous version. The differences between this

†Corresponding author. E-mail: amani.tahat@upc.edu; amanitahat@yahoo.com

© 2013 Chinese Physical Society and IOP Publishing Ltd

DBMS and other atomic databases are as follows. 1) It not only is a frozen data repository to save the atomic data tables but also allows regenerating the tables in different conditions by using different physical approximations. For example if someone searches for the value of difference between energy levels for the "Na-like iron ions" to transit from 1s to 2s using another database, he might find a specific value ($E = 2.3$ eV), or a relevant table. However, using this DBMS will allow updating the tables to obtain other values of E through using different computational methods and different input parameters. 2) Creating user profile/accounts, then generating his required data, the DBMS will store it directly in the database following the used computation theory with different extensions, with the ability to share the data with the public, or groups, as well as allowing the searching data inside the output files not only to have a file name, ..., etc., but this DBMS allows collaborative online data mining inside the system and using Python codes via the internet, which is conducive to finding atomic data mining from the literature and adding them to a user's profile. This DBMS has been fully described in a recently published data mining Python book[4] and conference papers.[5] The idea of this DBMS comes to solve several problems of atomic data resources and provides a work station with all facilities where the user need not be an expert in the physical theories and/or programming language. Therefore, particularly for a user who is not an expert in atomic data, the current DBMS websites and repositories of data will be very helpful. On the other hand, downloading the atomic software is allowable for use in any personal computer under several operating systems (Windows, UNIX, MAC plus Android). Moreover this DBMS with all its web calculators could be run and installed on smart phones (iPhone, etc.) as an application but the user needs to frequently maximize the phone memory. Therefore, this system will be a good option for an active researcher who wants to use the latest and/or the best available atomic data. The recent modifications in the ADBMS library according to adding a new technique for solving the Schrödinger equation have been described in Ref. [6]. A new freely distributed software has been recently published and described in Refs. [7] and [8] and also in the present paper. This study will focus on introducing the new features of the online ADBMS and its relevant new atomic tools.

2. Model and method

2.1. Flexible Python tool (AJAC)

The core idea of the current atomic tool is to design a user-friendly graphical user interface (GUI) so as to facilitate a user's interaction with the input/out data for some existing atomic codes[9] based on different theoretical methods. HTAC program serves as a starting point of creating this tool which was primarily developed to support the use of one model of the flexible atomic code (FAC), called the structure model.[10,11] Later, a Python scripting GUI was created for testing the functionality of HTAC/GUI,[12] which led to the discovery of some programming mistakes in the FAC library which were due to its complexity and fewer documentations. Consequently, HTAC is no longer being developed but a revised version of it has been linked to the current atomic tool in the meantime, and athe current tool is working online with highly qualified computing capabilities as a part of the above mentioned comprehensive online atomic database management system (DBMS).[3] Moreover, (HTAC and AJAC) can be freely downloaded from (http://aims-thinkenergy.org/amani/index.lua/section/4).

The present work is to introduce the recent updates of an online atomic database management system via a convenient and flexible Python tool (AJAC) that was developed based mainly on the maintenance of a powerful open source atomic code from the literature, namely the flexible atomic code (FAC) code.[13] The validity of the FAC code as a reliable atomic code was assessed through several studies,[14–19] and it has been used in a wide range of physical applications (e.g., magnetic fusion, astronomy, spectral diagnostic laboratory plasmas), and atomic databases (e.g., CHIANTI) even though it has some shortcomings that may reduce its strength.[2] The present results came to further illustrate the reliability of FAC via use of the AJAC tool including file manipulations, viewing, and plotting capabilities, as well as support of the multi operating systems (Windows, UNIX, Android) in addition to the online facilities of calculating, storing, mining, and sharing the atomic data. Calculation methods will follow the FAC code plus new linked methods of the AJAC library. The details of theoretical background and computational method of the FAC code are not discussed in this study, but they can be found in Ref. [13]. Furthermore the FAC code package consists of seven atomic models:

a) atomic structure model,

b) electron impact collisional excitation (EIE) model,

c) photoionization and radiative recombiation model,

d) autoionization and dielectronic recombination model,

e) electron impact ionization model "Collisional Ionization",

f) polarization model,

g) collisional radiative model (CRM),

which are to be used for producing various atomic radiative and collisional processes including energy levels, radiative transition rates, collisional excitation and ionization by electron impact, photoionization, autoionization, and their respective inverse processes. It also includes a collisional radiative model (CRM) to construct synthetic spectra including level population, rate coefficient, and spectrum for plasmas under

different physical conditions. The present tool facilitates the use of these models via a user friendly "GUI" and allows connection of the output of the first six models of FAC with that of the CRM model as an input. Furthermore, the AJAC allows running the electron impact excitation model of FAC code as it is based on the distorted wave approximation theory,[13] besides using a new approach, the so-called Dirac *R*-matrix approximation,[1] for the purpose of providing a complete and accurate data set mainly of energy levels, collisional strength, collision rate coefficients, and EIE cross sections. Herein, the AJAC Python GUI is linked to the FAC program to enhance its facilities and recombine its seven models in one working station, which, so far, are working separately, besides fixing the FAC bugs to let it work via AJAC tool and the AJAC web-based calculator version. It is also simplifying the way of composing the input file far from writing Python scripts as in the FAC package, with the help of examples and default input files, error message, reset, save, save as plot, and view output utilities. The most notable change is linking new calculation methods to its library, and gathering its output files easily to use and activate the database management system that is working online via an easy-to-use web-based GUI.

However, this release of AJAC is not only a GUI of FAC, i.e., with only an addition of GUI. The two programs differ a lot in programming structure and concepts, but do give quantitatively consistent results. As the coding efforts have moved on, numerous new features have been constantly added. Some existing capabilities are also expanded, especially in the program library, including the optimization routine (still under test); a new algorithm is incorporated into the construction of the diagonal sector Hamiltonian matrices where the time taken for the construction is dominated by the computation of the Slater integrals for reducing the time of computations and increasing the speed of generating the output data, based on the Slater integrals when finding the electron impact excitation data in FAC based on implementing the extended frequency-dependent quadrature rules (EFDQR)[20] and the CP methods (CPM) for the numerical solution of Schrödinger problems[21] with two-dimensional (2D) *R*-matrix theory.[22] Herein, CPM will allow production of eigenvalues, normalizing eigenfunctions, and calculating the first derivative of the eigenfunctions in a few tens of steps using a fixed step size.[20] This technique will lead to the achievement of high performance computing and grid architectures in order to study electron scattering from atoms, and ions, and in the meantime it will simplify the usage of parallel programming in order to run the program in a multiprocessor. Another technique has been successfully used in this aspect to speed up the computations of the Hamiltonian matrix in the Schrödinger equation via MPI parallel programming.[6] A tour of the features available in separated papers is made best by going through the tutorial data provided in the AJAC manual and the published open source code of AJAC. On the other hand, figure 1 shows the main window of AJAC. Furthermore, the scheme of the newly produced GUI of the FAC code via AJAC is illustrated in Fig. 2, where the programming languages and operating systems are also shown.

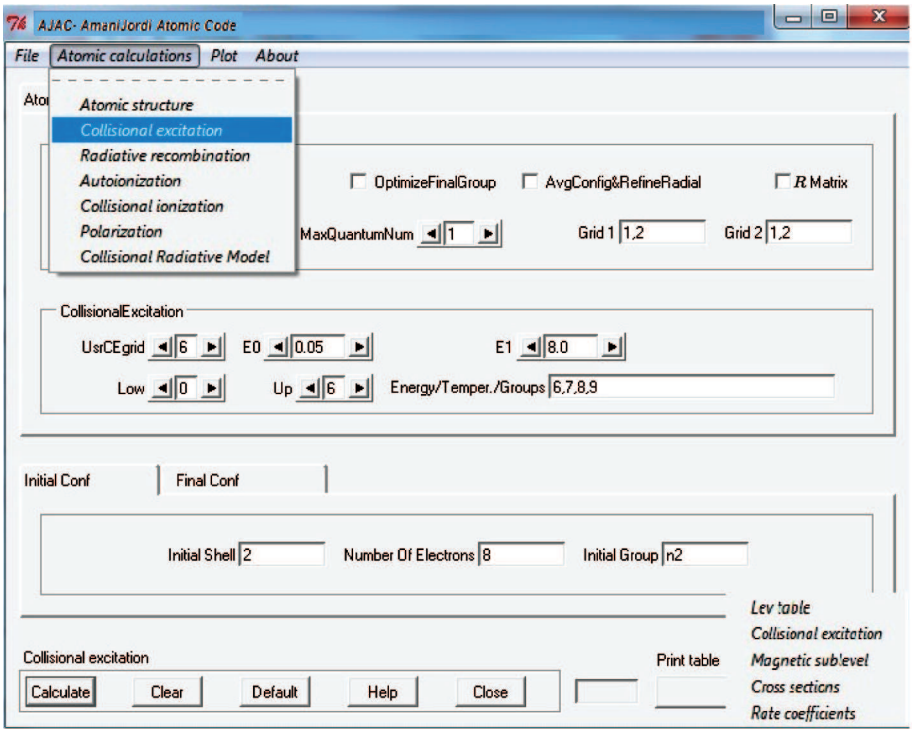


Fig. 1. (color online) Main window of the AJAC atomic tool.

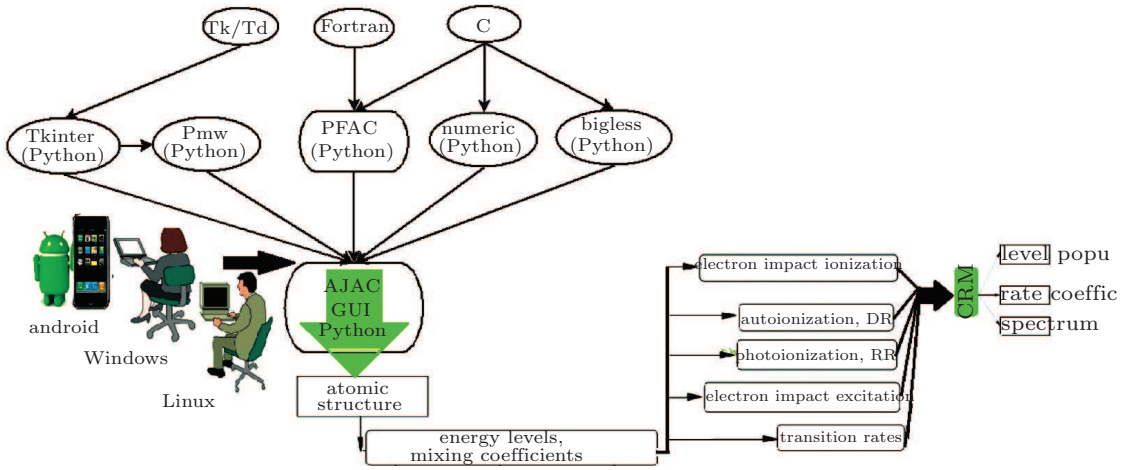


Fig. 2. (color online) The scheme of the newly produced GUI of the FAC code via AJAC, programming languages follow charts and allowable operating systems.

2.2. Accurate data set for electron impact excitation collisional strengths

For increasing accuracy, and as AJAC is used to generate the collisional data to be stored on the DBMS, we focus our attention on tabulating the collisional excitation data in a regular way so that we work with the collision strength rather than the cross section.

The collision strength Ω_{ij} between a given couple of levels, i.e., initial (i) and final (j) states as such $\Omega_{ij} = \Omega_{ji}$, is related to the more well-known quantity collision cross section σ_{ij} by

$$\Omega_{ij}(E) = k_i^2 \omega_i \delta_{ij} (\pi a_0^2), \tag{1}$$

where, a_0, ω_i, and k_i^2 represent the Bohr radius, statistical weight of level (initial state), and the colliding energy of the electron respectively. Furthermore Ω_{ij} is a dimensionless quantity which will lead to straightforward intercomparisons between various calculations.

Additionally, values of Ω_{ij} will be averaged over a Maxwellian distribution in a hot plasma because electrons have a wide distribution of velocities, and therefore the effective collision strength may be determined from

$$\gamma(T_\varepsilon) = \int_0^\infty \Omega(E) \exp(-E_j/kT_\mathrm{e}) \mathrm{d}(E_j/kT_\mathrm{e}), \tag{2}$$

where T_e is the electron temperature in Kelvin, E_j the electron energy with respect to the final (excited) state, and k the Boltzmann constant. Here, if the value of γ is known, the corresponding results for the excitation and de-excitation rates can be easily obtained. The value of γ might be enhanced up to a factor of ten (or even more)[2] when the contribution of resonance, especially for the forbidden transitions, is considered.

According to a study of the high energy configurations of Fe XVII–XXIII using the FAC code[23] which follows the DW method, the resonance contribution to the collision strength in FAC is approximated with a δ function and expressed as

$$\Omega_{ij} = \sum_d \pi g_d A^i_{di} \frac{A^i_{dj}}{\sum_i A^a_{di} + \sum_k A^r_{dk}} \delta(E - E_{id}), \tag{3}$$

where g_d, E_{id}, A^r, and A^i represent the statistical weight of the doubly excited state (d), the resonance energy, the radiative decay rate, and the autoionization rate, respectively. Therefore, the resonant excitation collision strength will be calculated with the independent process, isolated resonance approximation, in which the excitation from state (i) to (j) is treated as a two-step process: dielectronic capture forming a doubly excited state (d), followed by autoionization into state (i). Moreover, the resulting collision strengths, between initial state (i) and final state (j) are then averaged over a Maxwellian distribution of electron velocities to give the effective collision strengths as in Eq. (2). Herein, in FAC resonances with higher (n) are taken into account using the hydrogenic scaling laws of autoionization and radiative transition rates. The (2lq_1) configurations do not include those with two (2s holes). This means that the resonance contributions to the states such as (2pq3l) do not have adequate treatment.[23] However, such states, with small, direct and resonant excitation collision strengths, are generally less important. Although this method of FAC follows the DW approximation, it does not include the contributions from resonances, which could extensively affect the effective collision strengths. Thus the AJAC carries out the fully relativistic calculation by Dirac–Coulomb Hamiltonian in the R-matrix approach in order to provide collisional atomic data when the effects of resonances are considered. Figure 3 shows the structure of the "electron impact collisional excitation" model of AJAC.

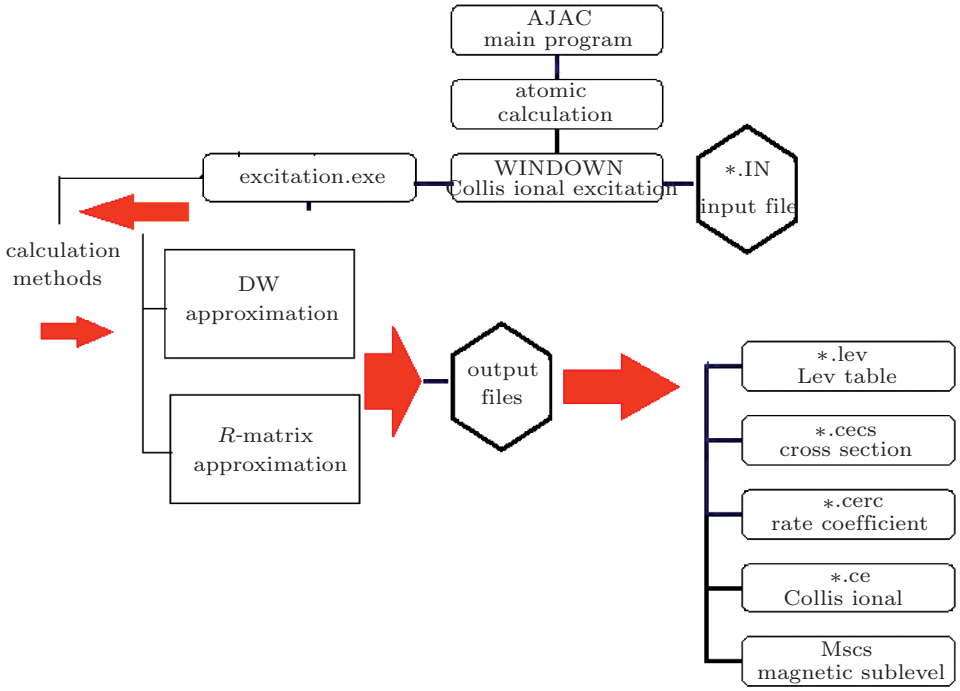


Fig. 3. (color online) Structure of the "electron impact collisional excitation? model of AJAC.

2.3. AJAC/*R*-matrix approach

According to the Dirac *R*-matrix theory descriptions,[22] the atomic system of this theory consists of one electron that could be in continuum and *N*-electron atomic target. The electron will be initially bound and finally free in the case of photoionization, but for the electron excitation process this electron will be free both initially and finally. In the *R*-matrix scheme it is assumed that a spherical boundary for *R*-matrix at $r = a$ divides the configuration space of all the particles into an external region and an internal region with (radius a) as shown in Fig. 4.

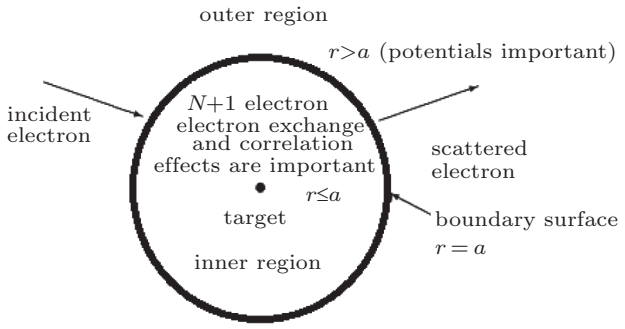


Fig. 4. (color online) Configuration space in *R*-matrix theory of electron atom scattering. In the external region (outer region) ($r > a$), where there is only Coulomb interaction and electron exchange with the atom is negligible, the scattering equations are considerably simplified though they are still non-trivial to solve in general. The internal region (inner region) ($r \leq a$), where there are the nuclear and Coulomb interactions, the free electron will be indistinguishable from the other *N* target electrons.

The *R*-matrix essentially gives the information about how the atomic electrons want an outer electron to enter and how they want an outer electron to leave the inner region. The atomic properties can be calculated by linking this information with the information from the outer electron: how it wants to enter and leave the inner region. This link then provides us with all significant atomic properties, such as energies and lifetimes of atomic states, collision strengths, and a variety of transition probabilities. Herein, for $(N + 1)$-atom structure calculation the electron wavefunctions will be constructed from a complete orthonormal set of bounds along with continuum one-electron functions. At *R*-matrix boundary, the bound functions will have zero amplitude although the continuum functions satisfy a specific *R*-matrix boundary condition there. On the other hand, "continuum" and "capture" are the two types of $(N + 1)$-electron wavefunctions. The possibility for the free electron to be temporarily captured by the *N*-target will be allowed by these wave functions. Further, the capture wavefunctions will be formed only from the bound one-electron functions. This will give rise to the important effect of resonance which dominates many of the low-energy scatterings. In the continuum states, the continuum one-electron functions are used to describe the extra electrons. Therefore, to obtain the energy independent *R*-matrix basis, this set of $(N + 1)$-electron wavefunctions will then be diagonalized over the Hamiltonian, so that the *R*-matrix can be calculated quickly for each energy including the resonance structure of processes based on the low energy range that is covered by the target states included in the problem. The results at higher energies may also be obtained in continuum states.[23] Therefore upon the above brief overview of the theoretical procedures, the *R*-

matrix computational calculations could be divided into two stages: inner region and outer region calculations; the inner calculations output will work as an input for the outer region calculations for the heaviest atomic targets, the Dirac equation must be solved to yield the *R*-matrix, *K*-matrix, and cross sections and collisional strength as well. The core of computing the collision strength using the *R*-matrix method via AJAC is presented in the flow diagram of Fig. 5 based on the AJAC input parameters, which is to be explained in Section 4 after explaining the theory of *R*-matrix and the related *K*-matrix.

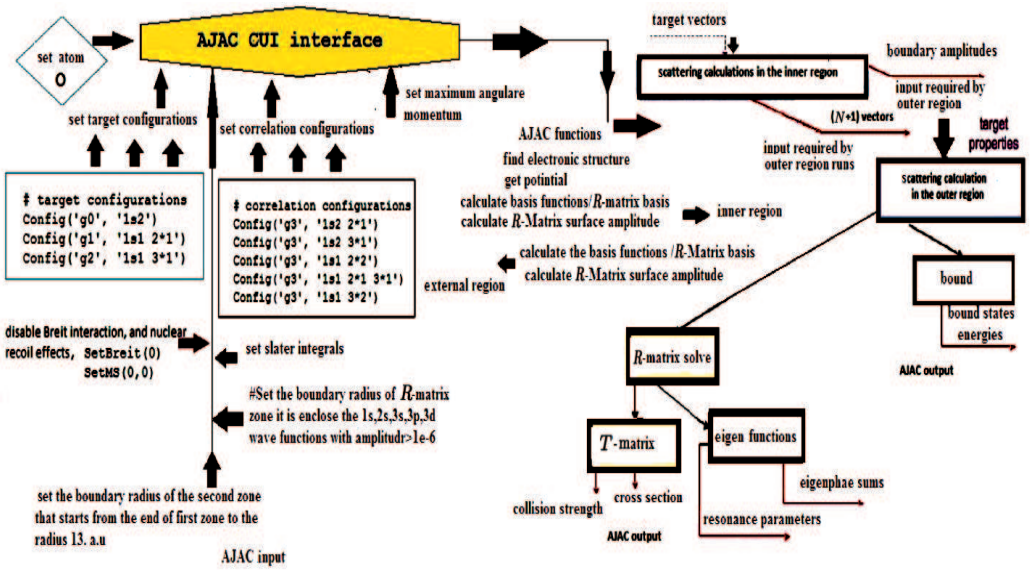


Fig. 5. (color online) Flow diagram of computing collisional strength Ω via (AJAC), including an input example for the scattering calculations in the outer region and inner region. The full explanation will be explained in Section 5.

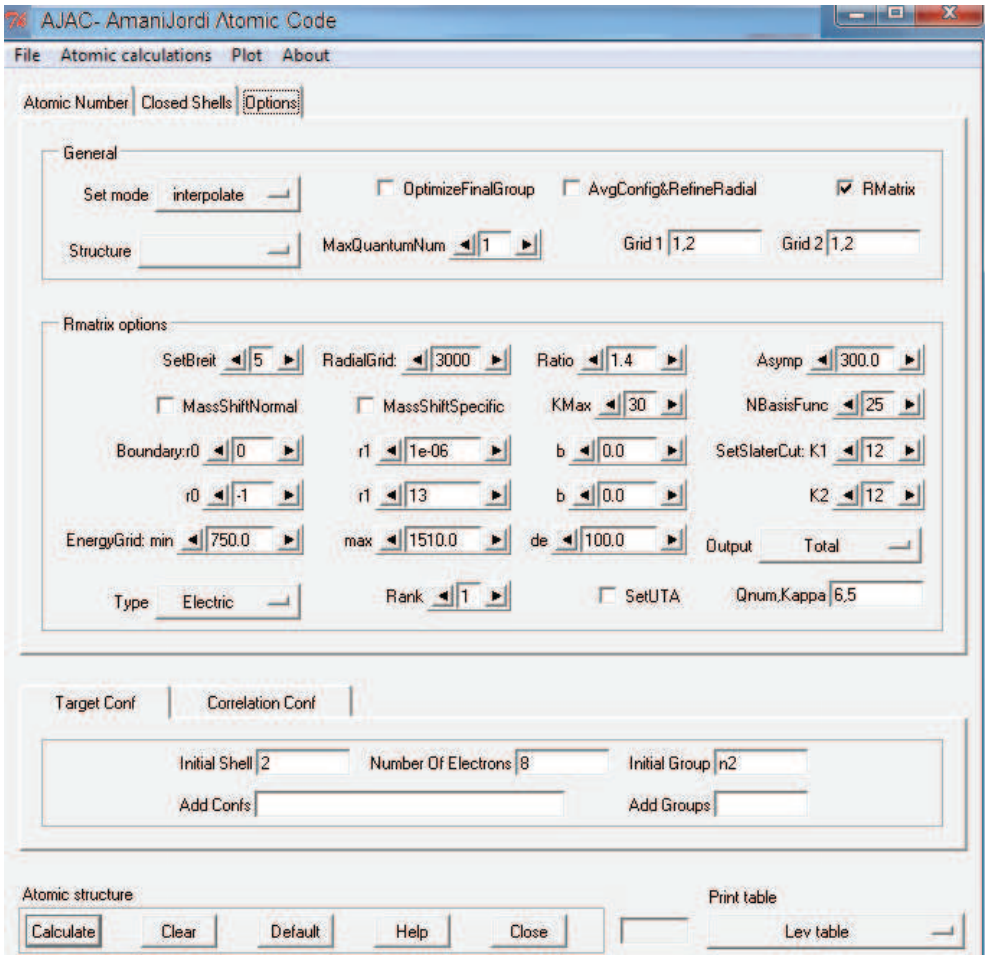


Fig. 6. (color online) *R*-matrix main window in AJAC.

Furthermore, the present computational theory follows the function of Dirac atomic R-matrix codes DARC[24] for computing the collisional strength Ω including the relativistic effects in a systematic way, in both the target description and the scattering model by calculating the R-matrix basis and surface amplitudes for any atom in a flexible way. This is done via an easy-to-use graphical interface shown in Fig. 6 using a combination of advanced programming languages (Python and C++) and some other related tools, as well as web design and database programming languages. DARC mostly produces reliable results, but in some sense it is outdated, contains hard-to-find and horrible bugs, uses an old-fashioned technology, and runs slowly, using lots of disc space and is inefficient.

3. Computational theory

3.1. Inner region

Relativistic effects (e.g., the spin–orbit interaction) need to be considered if the target with low atomic number Z can have a large nuclear potential as well as when non-relativistic solutions produce cancellation in physical quantities. For the time-independent equation solutions, the following equations are required:

$$H^{N+1}\psi = E\psi. \tag{4}$$

Here, for the $(N+1)$-electron system, the Dirac Hamiltonian could be written as (in atomic units)

$$H^{N+1} = \sum_{i=1}^{N+1} -c\alpha.\nabla_i + (\beta-1)c^2 - \frac{Z}{r_i} + \sum_{j=i+1}^{N+1} \frac{1}{|r_j - r_i|}, \tag{5}$$

$$\alpha = \begin{pmatrix} 0 & \sigma \\ \sigma & 0 \end{pmatrix}, \quad \beta = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{6}$$

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & \mathrm{i} \\ -\mathrm{i} & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{7}$$

where α and β are Dirac matrices constructed from Pauli spin and unit matrices, the i and j indexes of the individual electrons, and Z is the charge of an infinitely heavy point nucleus.

The final two-electron term is the Coulomb electron–electron repulsion. On the other hand, the first three one-electron terms in the Hamiltonian are i) momentum term, ii) mass term, iii) electron–nucleus Coulomb attraction.

The Hamiltonian matrix element can be split; therefore, angular integrals will be calculated using Racah algebra and radial integrals by numerical quadrature. Further relativistic corrections, such as the Breit interaction in addition to other quantum electrodynamic corrections (QED),[13] self-energy plus vacuum polarization, are not included in the Hamiltonian. Most equations are written here in Hartree atomic units (a.u.).

3.2. Model potentials

The computational problem could be radically reduced with large complex systems by using a model potential for describing the inner closed shell core of electrons. Consequently, integrals will only need to be calculated for valence electrons, that is, forming the excited states, as well as continuum electrons. So, the Dirac Hamiltonian becomes

$$H^{N+1} = \sum_{i=1}^{M+1} [-\mathrm{i}c\alpha\cdot\nabla_i + (\beta-1)c^2 - V(r_i)] + \sum_{j=i+1}^{M+1} \frac{1}{|r_j - r_i|}, \tag{8}$$

where M is the valence electron number in the target, and $V(r)$ the model potential satisfying the following boundary conditions: $V(r)_{r\to 0} \approx -Z/r$ and $V(r)_{r\to\infty} \approx -Z - N/r$. Although the basis that will be constructed from one-electron orbitals is reduced in size, the continuum functions have to be orthogonalized with the full set of bound orbitals.

3.3. Target states

The wave function of the $(N+1)$-electron could be constructed from N-electron target states, probably those including pseudo-states and the wavefunctions of a single particle for the continuum/valence electron. The target states will be the eigenfunctions of the following equation $H^N\Phi_i = E_i\Phi_i$. Here, i is an index of the eigenfunction that could be written as a configuration-interaction expansion of a basis set of N-electron functions $\Phi_i = \sum_k b_{ik}\psi_k$ based on a particular total angular momentum in addition to the overall parity. Here, b_{ik} denotes the configuration mixing coefficient which can be found by diagonalizing the target Hamiltonian matrix. On the other hand ψ_k is the configuration state function that has been obtained by the antisymmetric coupling together of N orthonormal single particle orbitals (N).[25] Pseudo-orbitals might be included for discretely approximating the continuum states that contribute to correlation. Further, the single particle functions have the form

$$\phi(r) = \frac{1}{r}\begin{pmatrix} p(r)x_{km} \\ \mathrm{i}Q(r)x_{\tilde{k}m} \end{pmatrix}, \tag{9}$$

where the continuum function is denoted by ϕ but the bound orbital by $\underline{\phi}$. Moreover, x_{km} is the two-component spinor that is composed of spherical harmonics and Clebsch–Gordan coefficients. To set the continuum orbital boundary condition on the surface of the internal region, only the continuum orbitals are non-zero. The solution of the following differential equations is the single particle continuum function basis:

$$\frac{\mathrm{d}p_i}{\mathrm{d}r} = -\frac{k}{r}p_i + \left(2c + \frac{\varepsilon_i}{c} - \frac{V(r)}{c}\right)Q_i - \frac{1}{c}\sum_{l=1}^{n}\lambda_{il}\underline{Q_l},$$
$$\frac{\mathrm{d}Q_i}{\mathrm{d}r} = \frac{k}{r}Q_i - \left(\frac{\varepsilon_i}{c} - \frac{V(r)}{c}\right)p_i + \frac{1}{c}\sum_{l=1}^{n}\lambda_{il}\underline{p_l}, \tag{10}$$

where i is the index of the continuum function, l the index of the bound orbital, and λ_{il} the Lagrange multiplier chosen such that the continuum and bound wavefunctions are orthogonal.

By combining the differential equations and using the orthogonality of the continuum functions where $\int_0^a (p_i \underline{p}_j + Q_i \underline{Q}_j) \mathrm{d}r = 0$, the boundary condition at $r = a$ is obtained as $Q_i(a)/p_i(a) = (b+k)/2ac$, where b is an arbitrary constant for all i's. Furthermore, the boundary condition will be reduced to

$$\frac{\mathrm{d}p_i}{\mathrm{d}r} = \frac{b}{a} p_i(a)$$

in the nonrelativistic limit. The boundary condition at the origin will be given by

$$\begin{aligned} p_i \; r &\xrightarrow{\approx} 0 \; p_i r^{\gamma}, \\ Q_i \; r &\xrightarrow{\approx} 0 \begin{cases} \dfrac{c(k+\gamma)}{z} p_i r^{\gamma}, & k > 0, \\ \dfrac{z}{c(k-\gamma)} p_i r^{\gamma}, & k < 0, \end{cases} \end{aligned} \tag{11}$$

where p_i is normalization constant and $\gamma = \sqrt{k^2 - (z/c)^2}$.

The functions can be made orthonormal over the internal region so that the scattered electron wavefunctions will be expanded into a linear combination of this basis set. Finally the radial parts of the bound orbitals will be the input to the R-matrix code as numerical values on a suitable mesh of points.

3.4. Driving the *R*-matrix

The total wavefunction which satisfies the equation of motion (Eq. (4)), for any energy (E), is expanded in terms of the basis of the wave function $\Psi_k = \sum_{ij} c_{ijk} A[\Phi_i, \phi_{ij}] + \sum_m d_{mk} \theta_m$ into $\psi_E = \sum_K A_{Ek} \psi_k$, which could be obtained after diagonalizing the $(N+1)$-Hamiltonian matrix over the internal region with eigenvalue of E_k^{N+1} and eigenvector $(c_{ijk}; d_{mk})$. Here, A is an asymmetrization operator for accounting electron exchange, k, j, and i are the eigenvector index, the continuum basis function index, and the channel index respectively. Defining the surface amplitudes as w_{ik} by

$$\frac{1}{r} \begin{pmatrix} w_{ik}(r) x_{km} \\ i v_{ik}(r) x_{\bar{k}m} \end{pmatrix} = \sum_j c_{ijk} \phi_{ij}(r);$$

and the wavefunction of the scattered electron in channel i could be written as an expansion

$$\begin{pmatrix} p_i(r) \\ Q_i(r) \end{pmatrix} = \sum_k A_{Ek} \begin{pmatrix} w_{ik}(r) \\ v_{ik}(r) \end{pmatrix};$$

then using the Hamiltonian and the boundary conditions of equation

$$\frac{Q_i(a)}{p_i(a)} = \frac{b+k}{2ac},$$

an expression for the expansion coefficients is eventually found to be

$$\begin{aligned} A_{Ek} &= \frac{1}{E_K^{N+1} - E} \left[\langle \Psi_E | H^{N+1} | \Psi_k \rangle - \langle \Psi_k | H^{N+1} | \Psi_E \rangle \right] \\ &= \frac{1}{E_k^{N+1} - E} \int_0^a \left[\sum_i c \frac{\mathrm{d}}{\mathrm{d}r} (p_{ivik} - Q_{iwik}) \right] \mathrm{d}r \\ &= \frac{1}{2a(E_k^{N+1} - E)} \\ &\quad \times \sum_i w_{ik}(a) [2acQ_i(a) - (b + \kappa_i) P_i(a)]. \end{aligned} \tag{12}$$

The R-matrix expression could be determined in terms of the large radial component of the continuum electron at the boundary

$$P_i(a) = \sum_j R_{ij} [2acQ_j(a) - (b + \kappa_i) p_j(a)],$$

where

$$R_{ij} = \frac{1}{2a} \sum_k \frac{w_{ik}(a) w_{jk}(a)}{E_k^{N+1} - E}$$

for the full R-matrix expansion. According to Szmytkowski and Hinze[26] R-matrix expansion is not quite correct in fact, because the truncation of the expansion of the R-matrix will lead to producing the largest error in the method because the Dirac R-matrix expansion of the wavefunction does not converge to the function on the surface boundary but the contribution of the missing terms from the diagonal can be important, especially for elastic scattering, so that the so-called "Buttle correction" approximation is sufficiently added based on the formula discussed by Buttle in 1967.[27,28]

3.5. Buttle correction

To include the correction, the radial differential equations for the continuum functions of Eq. (10) are solved without imposing the boundary conditions at $r = a$. So, if the R-matrix is truncated to the first K terms (i.e., continuum functions), subsequently the correction added to the diagonal will be given by

$$\begin{aligned} R_{ii}^c &= \frac{1}{2a} \sum_{k=K+1} \frac{p_{ik(a)^2}}{\varepsilon_{ik} - \varepsilon_i^c} = \left(\frac{2acQ_i^c}{p_i^c(a)} - b - \kappa \right)^{-1} \\ &\quad - \frac{1}{2a} \sum_{k=1}^{K} \frac{p_{ik}(a)^2}{\varepsilon_{ik} - \varepsilon_i^c}. \end{aligned} \tag{13}$$

Thus, the correction will only be calculated exactly at two points between each pair of adjacent eigenvalues for an arbitrary channel energy (ε_i^c) with the wavefunctions denoted as p_i^c, and Q_i^c. So, when ε_i^c is between the eigenvalues (ε_{ik}), the function R_{ii}^c is the error correction term. In practice it will be smoothed and approximated by quadratic Lagrange interpolation.[27]

3.6. Outer region

For solving the electron scattering problem, in the outer region where the correlation effects vanish, $r \geq a$, the close-coupling expansion of the wave function is used in terms of

the channel basis function, following Ref. [29]. By neglecting the electron exchange the scattered electron wavefunction satisfies the radial differential equations,

$$\frac{\mathrm{d}}{\mathrm{d}r}p_i+\frac{\kappa}{r}p_i-\left(2c+\frac{\varepsilon_i}{c}+\frac{z}{cr}\right)Q_i=-\frac{1}{c}\sum_{j=1}^{n}\sum_{\lambda=1}^{\lambda_{\max}}\frac{a_{ij}^{\lambda}}{r^{\lambda+1}}Q_j, \quad (14)$$

$$\frac{\mathrm{d}}{\mathrm{d}r}Q_i-\frac{\kappa}{r}Q_i+\left(\frac{\varepsilon_i}{c}+\frac{z}{cr}\right)p_i=\frac{1}{c}\sum_{j=1}^{n}\sum_{\lambda=1}^{\lambda_{\max}}\frac{a_{ij}^{\lambda}}{r^{\lambda+1}}p_j, \quad (15)$$

where n is the number of channels, and $z=Z-N$ the residual charge.

The expansion of the long-range potential in terms of Legendre polynomials will lead

$$\sum_{k=1}^{N}\frac{1}{|r_k-r_{N+1}|}=\sum_{\lambda=1}^{\infty}\left(\frac{1}{r_{N+1}}\right)^{\lambda+1}\sum_{k=1}^{N}r_k^{\lambda}p_{\lambda}(\hat{r}_k\cdot\hat{r}_{N+1}), \quad (16)$$

where the asymptotic scattering coefficients are

$$a_{ij}^{\lambda}=\mathrm{Ang}\left\langle[\Phi_i,\phi_i]\right|\sum_{k=1}^{N}r_k^{\lambda}p_{\lambda}(\hat{r}_k\cdot\hat{r}_{N+1})\left|[\Phi_j,\phi_j]\right\rangle. \quad (17)$$

Here, $a_{ij}^{0}=N\delta_{ij}$ and Ang mean that the integral of the outer electron radial coordinate $r_{N+1}=r$ is left out. On the other hand, the multipole expansion of the long-range potential is truncated to $\lambda_{\max}$ terms in view of the fact that the range of λ for which $a_{ij}^{\lambda}\neq 0$ is determined by the angular momenta of the channels i and j.

3.7. Electron scattering

For completeness, the electron scattering cross section expression will be explained here by using the S-matrix and T-matrix to relate the K-matrix with particular total angular momentum J and overall parity π values, and it is given by

$$S=(1+\mathrm{i}K)(1-\mathrm{i}K)^{-1};\quad T=S-1. \quad (18)$$

Here, the K-matrix could be obtained by solving the differential equations through a matching process to the R-matrix similar to that given in Refs. [28] and [30]. For a transition of the target from state (r) to state (s), the partial collision strength which contributes to the cross section can be given as a summation over the channels from $(\alpha_r J_r^t j_r)$ to $(\alpha_S J_S^t j_S)$, where J^t and α represent the target N-electron state total angular momentum and the other quantum number that uniquely identifies a target state, respectively. Therefore the total collision strength will be represented as a sum of those partial collision strengths over two symmetry, $\Omega_{rS}^{J\pi}=(g/2)\sum_{j_r j_S}|T_{rs}|^2$; $\Omega_{rS}=\sum_{J\pi}\Omega_{rS}^{J\pi}$. Here, J is formed by coupling vectorially J^t and j, the total angular momentum of the additional electron ($N+1$ or target+ electron). For jj-coupling g equals $(2J+1)$, and Ω_{rS} should be symmetric. Then the total cross section for this transition will be given by

$$\sigma_{r\to S}=\frac{\pi a_0^2}{k_r^2 g_r}\Omega_{rS},$$

where $g_r=2J_r^t+1$ and a_0 is the Bohr radius converted from atomic units.

4. Example of computing collision strength Ω of (He-like O) calculated by using R-matrix/AJAC

The configuration of the (He-like O ions) as shown in Fig. 5 is used to explain the input and output of running the AJAC/R-matrix. The user needs to first set the atom symbol to (O) from the main window, then AJAC will directly use the following parameters about "O" during the calculations (atomic number: 8; atomic mass: 15.9994 amu).

AJAC will calculate "R-matrix basis" and surface "amplitudes for (He-like O) " as an output of the inner region calculations and then the AJAC will automatically use them as an input for the outer region calculations to obtain the collisional strength as an output including the resonance parameters.

The AJAC functions of inner region will run the target calculations which include (1s2, 1s1 2∗1), and (1s1 3∗1) configurations as soon as finalizing the input settings by AJAC user. The user needs to fill the input parameters and electronic configuration from the R-matrix main window as shown in Fig. 6. The important parameters are fully described in the AJAC manual (e.g., set grid points, set Breit, maximum angular momentum, set Slater integrals, set boundary radius of the first R-matrix first region which here includes the (1s, 2s, 2p, 3s, 3p, 3d) wavefunctions with amplitude $>(10^{-6})$ required for potential calculations, and set boundary radius of the second region starting from the end of first region, to the default radius 13.0 a.u.). On the one hand, to set the configuration someone needs to start with setting the following target configurations as an input to allow finding the configuration of ($'$g0$'$, $'$1s2$'$), ($'$g1$'$, $'$1s1 2∗1$'$) and ($'$g2$'$, $'$1s1 3∗1$'$) through specifying the correlation states ([$'$g0$'$, $'$g1$'$, $'$g2$'$], [$'$g3$'$]) of the targets separately as follows. The second step is to set the expected correlation configurations as explained in the theory part, thus allowing the AJAC configuration calculations of the following inputs correlation configurations ($'$g3$'$, $'$1s2 2∗1$'$), ($'$g3$'$, $'$1s2 3∗1$'$), ($'$g3$'$, $'$1s1 2∗2$'$), ($'$g3$'$, $'$1s1 2∗1 3∗1$'$), and ($'$g3$'$, $'$1s1 3∗2$'$). The methods of finding the electronic structure follow the configuration interaction method (CI).[13] AJAC will open a separate window to show the calculating processes including printing the basis functions for the second and first regions, and use the maximum angular momentum and functions per-kappa to generate the R-matrix surface amplitudes.

AJAC will consider these values as an input to start computing the collision strength by calling the C++ function from its library after pressing the button "calculation". These calculations need to diagonalize the *R*-matrix to generate eigen phases and some resonance parameters, then solve *R*-matrix equation sets as well as the associated *T*-matrix equations, the collision strength and cross section; moreover the bound state energy will be found also. Table 1 shows the samples of output of target configuration ($'$g0$'$, $'$1s2$'$), ($'$g1$'$, $'$1s1 2$*$1$'$), and correlation configuration ($'$g3$'$, $'$1s2 2$*$1$'$).

Table 1. AJAC output sample.

Atomic number $Z = 8$, $E = 0$, -1.60953951×10^3,	
Energy	Collision strength
7.50000000×10^2	5.77439824
8.50000000×10^2	1.25134775×10^1
9.50000000×10^2	2.55429188×10^1
1.05000000×10^3	4.46183535×10^1
1.15000000×10^3	1.14695450×10^2
1.25000000×10^3	1.36345772×10^2
1.35000000×10^3	8.92944520×10^1
1.45000000×10^3	1.33620418×10^2

5. Results and discussion

Here, the validity and utility of the present developed atomic tool are presented. The flexible graphical user interface of the AJAC package makes it easy to specify the input and schedule the computation processes, viewing and plotting capabilities of the atomic data, and therefore allowing typical calculations to be carried out with minimal understanding of theoretical atomic physics of the FAC code and the *R*-matrix theory as well. AJAC input parameters for each method of computation are fully described in the AJAC manual and FAC manual as well.[31,32] This study provides the following calculations for testing the validity of the AJAC tool. Extensive comparisons between the available results in the literature and the present calculations are made in order to assess the accuracy of the reported results.

5.1. Argon (Ar) atom (atomic number = 18; electronic configuration [[Ne] 3s^2 3p^6])

The AJAC is used also for calculating the energy differences for the excitation transitions from the lowest metastable state levels of the $3p^5 4s$ configuration to the ten higher-lying fine structure levels of the $3p^5 5p$ configuration of "argon" atom (Ar), at $J = 0, 2$. The results are listed in Table 2.

Table 2 presented the calculated energy differences of "argon". The table also includes the data from the NIST website (http://www.nist.gov/pml/data/asd.cfm) for the energy differences. As compared with NIST database energy levels, the AJAC ($3p_2$) and ($3p_3$) levels are inverted with a very small energy difference. The AJAC calculated energy differences are in reasonable agreement with the NIST experimental values as well as GRASP code values.

Table 2. Energy differences (eV) for the transitions of argon, NIST: experimental values from the NIST database; GRASP: values from (general-purpose relativistic atomic structure package) code.[31]

Transition	NIST	AJAC/DW	GRASP	AJAC/*R*-matrix
$1s_5$–$3p_{10}$	2.91564	2.79962	2.84473	2.87113
$1s_5$–$3p_9$	2.95069	2.88487	2.87598	2.89972
$1s_5$–$3p_8$	2.95771	2.89742	2.88965	2.90110
$1s_5$–$3p_7$	2.97656	2.90426	2.91211	2.99039
$1s_5$–$3p_6$	2.98056	2.90738	2.91699	2.99305
$1s_5$–$3p_5$	3.02759	3.01705	3.03809	3.05100
$1s_5$–$3p_4$	3.13229	3.09321	3.08887	3.09918
$1s_5$–$3p_3$	3.13994	3.08777	3.09961	3.09636
$1s_5$–$3p_2$	3.13877	3.10139	3.10156	3.12204
$1s_5$–$3p_1$	3.18976	3.39228	3.42383	3.41529
$1s_5$–$1s_3$	0.17481	0.21704	0.20215	0.26630

5.2. Ne-like iron (Fe16+, Fe XVII), (atomic number = 26; electronic configuration [[He]2s^2 2p^6])

For testing the validity of the AJAC, the collision strength transitions of an excitation cross section for (Ne-like iron) dipole allowed ($2 \rightarrow 3$) transitions are investigated, and the results are listed in Table 3. The first column is an index for the upper levels of transition and the lower level has been set to the ground state.

In Table 3, AJAC-DW results are close to the results of both Hagelstein and Jung' code[32] and also the results of Zhang *et al.*'s code[33] because the DW approximation method was adopted in these atomic codes for treating electron–ion collision processes[2] and generating the collision excitation data. On the other hand the AJAC *R*-matrix values are slightly higher than the AJAC-DW values by $\approx 4\%$. In general a comparison between the results of the two calculation methods of AJAC shows that they are in good agreement with the results of a comparative study from the literature[34] which focused on comparing the results of the FAC code from relativistic DW approximation and Dirac *R*-matrix theory to study the electron impact excitation properties, but only for the Fe^{21+} (FeXXII) and addressed that *R*-matrix values are higher than DW values, which demonstrates the FAC validity and the AJAC validity as well.

In this regard, the majority of atomic software adopts the DW approximation method and there are many versions of the DW method that could be found also in HULLAC package[35] and Chen's code.[36] Every code has its own merits and deficiencies. No particular code could fully meet all requirements. Consequently the use of a particular code must be dependent on its availability, and familiarity to the user, in addition to what parameters and level of accuracy are required. However, in a sense some codes are easier to use than others (particularly the FAC code), nevertheless, it is not too difficult to obtain the required results from the other codes. However, the desired level of accuracy for the results is difficult to achieve. To achieve the best accuracy someone needs to generate his atomic data based on using several atomic codes with different physical approximations so that he can compare the result and determine the accuracy.

Table 3. Comparison between the collision strengths of excitation cross sections for (Ne-like iron) dipole allowed ($2 \rightarrow 3$) transitions, computed from the AJAC and other codes.

Upper	Scattered electron energy/eV	Zhang *et al*. code	Hagelstein and Jung code	AJAC (DW) method	AJAC *R*-matrix method
$(2p_{3/2}3s_{1/2})_1$	100.0	1.85×10^{-3}	1.96×10^{-3}	2.03×10^{-3}	2.101×10^{-3}
	2500.0	7.98×10^{-3}	8.90×10^{-3}	9.23×10^{-3}	9.34×10^{-3}
$(2p1/23s1/2)_1$	100.0	1.66×10^{-3}	1.73×10^{-3}	1.79×10^{-3}	1.81×10^{-3}
	2500.0	6.61×10^{-3}	7.30×10^{-3}	7.69×10^{-3}	7.71×10^{-3}
$(2p3/23d_{3/2})_1$	100.0	5.80×10^{-3}	5.98×10^{-3}	5.39×10^{-3}	5.43×10^{-3}
	2500.0	1.66×10^{-3}	1.69×10^{-3}	1.58×10^{-3}	1.62×10^{-3}
$(2p3/23d_{5/2})_1$	100.0	2.44×10^{-2}	2.41×10^{-2}	2.49×10^{-2}	2.54×10^{-2}
	2500.0	5.88×10^{-2}	6.05×10^{-2}	6.21×10^{-2}	6.33×10^{-2}
$(2p1/23d_{3/2})_1$	100.0	9.68×10^{-2}	9.39×10^{-2}	9.57×10^{-2}	9.61×10^{-2}
	2500.0	2.46×10^{-1}	2.49×10^{-1}	2.47×10^{-1}	2.48×10^{-1}
$(2s1/23p1/2)_1$	100.0	9.39×10^{-4}	9.57×10^{-4}	9.04×10^{-4}	9.11×10^{-4}
	2500.0	1.84×10^{-3}	2.00×10^{-3}	1.78×10^{-3}	1.98×10^{-3}
$(2s1/23p3/2)_1$	100.0	2.27×10^{-3}	2.48×10^{-3}	2.17×10^{-3}	2.37×10^{-3}
	2500.0	1.34×10^{-2}	1.43×10^{-2}	1.31×10^{-2}	1.42×10^{-2}

5.3. Time testing: computing time of He-like O

After testing AJAC it is found that when calculating the *R*-matrix it takes a short time for some elements, but for others it is spending a long time and needs more than 256 Mb of RAM capacity because the nature of *R*-matrix method computational techniques makes the computing very slow, especially when computing the *R*-matrix-basis for each region. Here a time testing example is presented.

This example is to compare the execution time of AJAC calculation methods for computing the electron impact collisional excitation data for (He-like O) when the electronic configuration is (1∗2 2∗1). When using DW method with initial configuration (1s2) and final configuration (2∗1), the execution time is equal to (18.01 s). But, when using the *R*-matrix method at the target configuration (1s2, g2) and correlation configuration (1∗1 2∗1, g3), the execution time is equal to (2min:28s:21). Repeating these calculations using another electronic configuration for He-like O, ((1∗2 3∗1), DW initial configuration (1s2), final configuration (3∗1), the execution time is (8:86 s). But when using the *R*-matrix with target configuration (1s2, g2) and correlation configuration (1∗1 3∗1, g3), the execution time is (7 min:42 s:13). Therefore the computing time of DW method is less than that of *R*-matrix; the execution time of DW is very short and exceeds no more than a few seconds. But *R*-matrix computing time is very slow and takes a very long time compared with DW execution time. Also the *R*-matrix becomes slower when we increase the configuration levels (from level 1 to level 2) but it is faster than from level 1 to level 3. So, the *R*-matrix calculations are very slow and in a sense computationally impossible for complex (i.e. heavy) species, which is because of the difficulty of the theory and because it efficiently takes the resonance in electron–ion interaction into account which also leads to increasing the time of computations. The AJAC can solve this problem by adopting (2DRM–EFDQR) rules to increase the speed of *R*-matrix calculation. The user of AJAC can easily speed up the process of calculating the collisional strength by pointing the 2D *R*-matrix-checking box then filling the specifying field and pressing the option collisional strength from the print menu bar: this option will enable a fast computation of the included Slater integrals when setting the Slater integrals as an input parameter. Additionally, the use of the AJAC program and the "collisional excitation" model is described in detail in its manual, including features, input files, output files, clarifications of the parameters that must be determined by the user, the way of installing and running it.

Consequently linking the *R*-matrix method to AJAC and the ADBMS as well, will enhance the accuracy of the production of atomic data for the use of plasmas and astrophysics, based on the importance of resonances in level populations which is dependent on the transition considered, the plasma temperature and velocity distribution, because of the inaccuracy of the collision strength at high energy. The *R*-matrix results should be considered more accurately and should be used in spectral codes to complement the data for the higher energy configurations although the *R*-matrix approximation is more accurate in dealing with resonances than the approximation used by FAC.

5.4. Utility

The AJAC has an application with two powerful user interfaces: a) a graphical user interface, GUI, working on computers based on multi-operating systems including Windows (Windows 98/ME/NT/XP/Vista/2000/2003/2007/Unix), b) a web-based GUI user interface that is working online and could be installed on smart phones as an application. The difference between the two interfaces is that the web interface could be run in any operating system which is able to run either Internet Explorer 6.0 or Firefox/Mozilla — without installing any client software but it has fewer interactive notification options

and its atomic calculator models (e.g., polarization, ionization, etc.) are still under construction. Also this interface has been supported with an integrated database system, and it has extra features such as allowing building an automatic database based on the generating outputs, allowing data searching and data mining, in addition to creating user accounts and sharing data. The computer-based GUI cannot be used for data searching but it can allow sorting output files following the calculating method in a folder that is related to the used method by default. Moreover it allows data presentation and plotting and various platforms. This software could be recognized as integrated software that provides solutions to reduce the shortcomings of the existing atomic software (FAC), to allow the user to have a greater benefit when using it than using the previous scripting version. Here, the user of the AJAC can easily draw all available required atomic data that may be extracted from the tabulated output files, at the moment, with the ability to save the different input and output files in the process of computations. The result is a large increase in the flexibility of computer use and a consequent rise in user productivity and time consumption when exporting data to another plotting program because some programs do not support such output files. The AJAC provides a series of graphs to illustrate the extracted atomic data by using several computational methods. For example, in the electron impact excitation model the user needs to first run the program "collisional" and then choose the computational method (DW or *R*-matrix) to produce the collisional excitation table, the table for a specific atomic number that has been stored in a file of type, (∗.ce , ∗.cers, ∗.cerc, ∗.cemc) in the case of using DW approach. Otherwise, another file extension will be generated, then the AJAC extracts the required input data from the tabulated output files automatically to be plotted by clicking the item plot from AJAC standard bar. The plot menu provides three options (create coefficients, ce energies strength, ce energies cross sections); selecting any of them means specifying the input data to be extracted from the output files. The use of the plotting part is available in AJAC manual and the plotting menu bar is shown in Fig. 7.

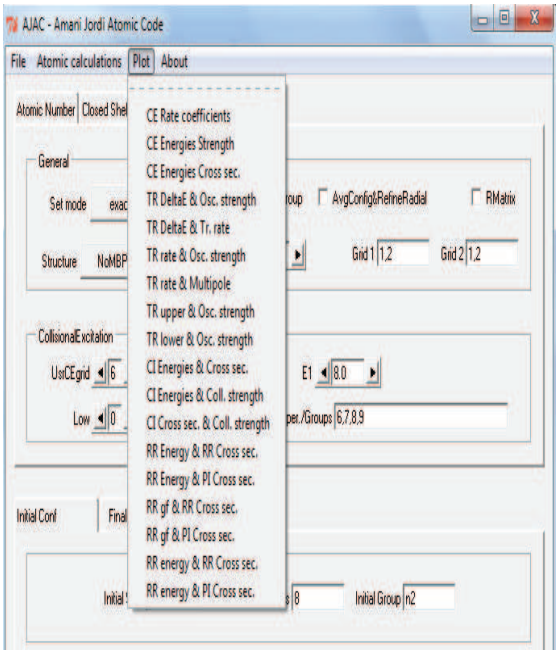


Fig. 7. (color online) Plot menu bar of AJAC.

6. Conclusion

In this study we successfully describe a freely distributed open source atomic tool, called (AJAC). The AJAC is presented as an easy-to-use graphical user's interface (GUI), serving as a part of an online atomic database management system, which comes into being mainly due to the attempt to maintain and simplify the use of an existing powerful complex atomic program, FAC. Therefore, the AJAC provides a graphical user's interface for the FAC code working online and is supported by an integrated database management system. The GUI provides a control platform for linking, in addition to managing, the use of FAC code models as well as their input and output. Brief theoretical background is presented. Successful implementation of the Dirac *R*-matrix method is linked to the AJAC library to enhance the features of the FAC code when running the electron impact excitation model, for the purpose of producing accurate atomic data by using different theoretical methods in one workstation with several utilities including plotting capabilities, file manipulations, and data searching. The AJAC will be important for coping with the requirements of plasma physics and astrophysics and could be a complementary resource of currently available atomic data resources. Finally, the source code could be used as a good example for software engineering GUI developers who are interested in creating Python GUI for computer programs with a FORTRAN and C++ library.

References

[1] Zombeck M V 2007 *Handbook of Space Astronomy and Astrophysics* (Cambridge: Cambridge University Press) p. 37
[2] Aggarwal K and Keenan F 2010, *Proceedings of a Workshop on Uncertainties in Atomic Data and How They Propagate in Chemical,* October, 25–27, 2010, Tenerife, Spain, p. 1
[3] Tahat A and Salah W 2011 *Int. J. Data Database Manage Syst.* **3** 1
[4] Tahat A and Khwaldeh A 2011 *Atomic Data Mining Numerical Methods, Source Code Sqlite with Python* (Irbid: Dar ALKetab ALThaqafi) p. 510
[5] Tahat. A, Marti J and Tahat M 2012 *Proceedings of the 2nd International Conference on Integrated Information IC-ININFO,* August 30–September 3, 2012, Budapest, Hungary, in press
[6] Khwaldeh A, Tahat A and Marti J 2012 *J. Appl. Sci.* **12** 2115
[7] Kawaldeh A, Tahat A, Marti J and Tahat M 2012 *Information Technology Journal* **11** 1553
[8] Tahat A, Salah W and Hallak A 2010 *Int. J. Pixe* **20** 63
[9] Hamasha S and Tahat A 2010 *Python Papers* **5** 1
[10] Hamasha S, Abu Allaban M and Tahat A 2011 *J. Appl. Sci.* **11** 2686
[11] Hamasha S and Naser A 2011 *J. Appl. Sci.* **11** 3413
[12] Tahat A, Salah W and Tahat M 2011 *Python Paper Source Codes* **3** 1
[13] Gu M F 2008 *Canadian Journal of Physics* **86** 675
[14] Shi X, Wang Y, Chen C and Gu M 2005 *Chin. Phys.* **14** 1009

[15] Gu M 2003 *Astrophys. J.* **590** 1131

[16] Foster A and Testa P 2011 arXiv: 1107.0470v2 [astro-ph.SR]

[17] Kinkhabwala A, Behar E, Sako M, Gu M F, Kahn S M and Paerels F B S 2003 arXiv: 0304332 [astro-ph]

[18] Li Z, Chang M, Min H, Yang C and Yan W 2009 *Chin. Phys. B* **18** 1647

[19] Zhang Z R, Cheng X L, Liu Z J, Yang J H and Li H F 2012 *Chin. Phys. B* **21** 013101

[20] Ixaru L, Scott N and Scott M 2006 *SIAM Journal on Scientific Computing (SISC)* **28** 1252

[21] Ixaru L 2000 *J. Comput. Appl. Math.* **125** 347

[22] Burke P G 2011 *Application to Atomic, Molecular and Optical Processes: R-Matrix Theory of Atomic Collisions* (Berlin: Springer-Verlag) p. 322

[23] Landi E and Gu M F 2006 *Astrophys. J.* **640** 1171

[24] Grant I P, McKenzie B J, Norrington P H, Mayers D F and Pyper N C 1980 *Comput. Phys. Commun.* **21** 207

[25] Dyall K, Grant I P, Johnson C T, Parpia F A and Plummer E P 1989 *Comput. Phys. Commun.* **55** 425

[26] Szmytkowski R and Hinze J 1996 *J. Phys. B: At. Mol. Phys.* **29** 761

[27] Buttle P J A 1967 *Phys. Rev.* **160** 719

[28] Chang J 1975 *J. Phys. B: At. Mol. Phys.* **8** 2327

[29] Gianturco F A and Huo W M 1995 *Computational Methods* for *Electron Molecule Collisions* (New York: Plenum) p. 239

[30] Burke P G, Hibbert A and Rob W D 1971 *J. Phys. B: At. Mol. Phys.* **4** 153

[31] Grant I P, McKenzie B J, Norrington P H, Mayers D F and Pyper N C 1980 *Comput. Phys. Commun.* **21** 207

[32] Hagelstein P L and Jung K R 1987 *Atom. Data Nucl. Data Tables* **37** 121

[33] Zhang H L, Sampson D H and Mohanty A K 1989 *Phys. Rev. A* **40** 616

[34] Gu M F 2004 *Phys. Rev. A* **70** 062704

[35] BarShalom A, Klapisch M and Oreg J 2001 *J. Quantum Spectrosc. Rad. Transfer* **71** 169

[36] Chen G 1996 *Phys. Rev. A* **53** 3227