Eur. Phys. J. D (2024) 78:75
https://doi.org/10.1140/epjd/s10053-024-00865-z

THE EUROPEAN
PHYSICAL JOURNAL D

Regular Article - Atomic Physics

Merits of atomic cascade computations

S. Fritzsche[1,2,3,a], A. K. Sahoo[1,4], L. Sharma[4], Z. W. Wu[1,5], and S. Schippers[6,7]

[1] Helmholtz-Institut Jena, Fröbelstieg 3, 07743 Jena, Germany
[2] GSI Helmholtzzentrum für Schwerionenforschung, 64291 Darmstadt, Germany
[3] Theoretisch-Physikalisches Institut, Friedrich-Schiller-Universität Jena, 07743 Jena, Germany
[4] Indian Institute of Technology Roorkee, Roorkee, Uttarakhand 247667, India
[5] Key Laboratory of Atomic and Molecular Physics and Functional Materials of Gansu Province, College of Physics and Electronic Engineering, Northwest Normal University, Lanzhou 730070, People's Republic of China
[6] I. Physikalisches Institut, Justus-Liebig-Universität Giessen, Heinrich-Buff-Ring 16, 35392 Giessen, Germany
[7] Helmholtz Forschungsakademie Hessen für FAIR (HFHF), GSI Gesellschaft für Schwerionenforschung, Campus Giessen, 35392 Giessen, Germany

Received 16 February 2024 / Accepted 7 May 2024 / Published online 14 June 2024

Abstract. Atomic cascades refer—first and foremost—to the stepwise de-excitation of excited atoms owing to the emission of electrons or photons. Apart from dedicated experiments at storage rings and synchrotrons, such cascades frequently occur in astro and plasma physics, material research, surface science and at various places elsewhere. In addition, moreover, "atomic cascades" have been found a useful concept for modeling atomic behavior under different conditions, for instance, when dealing with the photoabsorption of matter, the generation of synthesized spectra, or for determining a rather wide class of (plasma) rate coefficients. We here compile and discuss several atomic cascades (schemes) that help predict cross sections, rate coefficients, electron and photon spectra, or ion distributions. We also demonstrate how readily these schemes have been implemented within JAC, the Jena Atomic Calculator. Emphasis is placed on the classification of atomic cascades and their (quite) natural breakdown into cascade *computations*, to deal with the electronic structure and transition amplitudes of atoms and ions, as well as the cascade *simulation* of those properties and spectra, that are experimentally accessible. As an example, we show and discuss the computation of dielectronic recombination plasma rate coefficients for beryllium-like gold ions. The concept of atomic cascades and its implementation into JAC can be applied for most ions across the periodic table and will facilitate the modeling and interpretation of many forthcoming observations.

1 Introduction

Atomic cascades occur frequently in nature owing to the interaction of matter with particles and light. Such stepwise changes of an atomic and/or ionic ensemble are often "caused" either by the excitation of inner-shell electrons due to photon, electron, or proton impact [1,2], or by the capture of electron(s) into Rydberg orbitals as observed in many astrophysical environments [3–5]. If, for instance, an atom or ion is initially excited into the continuum of the next higher (or even a several times higher) charge state, it will stabilize itself via various decay processes toward some ground configuration [6,7]. Experimentally, this stabilization process is usually observed in terms of photon, electron or ion spectra, from which further information about the state and composition of matter is extracted afterward [8,9]. In astrophysics, for example, much of the information about distant stars originates from the photon emission of particular atoms or ions at well-defined wavelengths and may help reveal the temperature, density and chemical composition of these objects [10–12]. Other applications of atomic cascades refer to the diagnostics of plasmas [13,14], the neutralization of ions near surfaces [15,16], the radiation damage in biological matter [17] and up to the chemical evolution of the universe [18]. Figure 1 shows how the concept of atomic cascades may help to explore and utilize the underlying physics in different research areas.

Despite their frequent need and influence, a thorough analysis and modeling of (atomic) cascades is difficult and still a great challenge for atomic physics. The many facets and the complexity of such cascades therefore require a clear distinction between different scenarios (schemes) in changing stepwise either the excitation and/or the charge state of the given atoms and ions. This complexity of atomic cascades also suggests to separate the computation of their electronic structure from the simulation of particular properties. Often, moreover, it is essential to introduce and apply a hierarchy of cascade approaches in order to keep the computation

[a] e-mail: s.fritzsche@gsi.de (corresponding author)

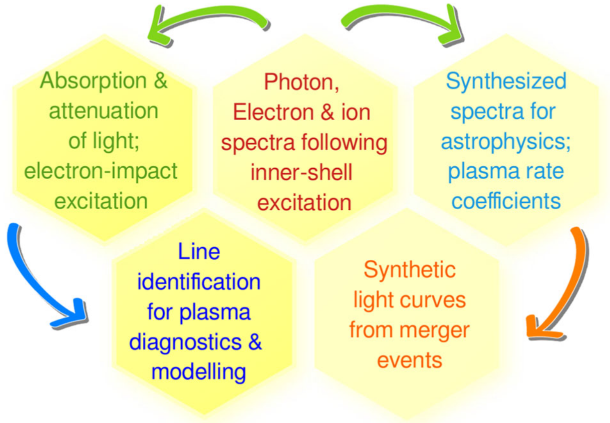

Absorption & attenuation of light; electron-impact excitation
Photon, Electron & ion spectra following inner-shell excitation
Synthesized spectra for astrophysics; plasma rate coefficients
Line identification for plasma diagnostics & modelling
Synthetic light curves from merger events

Fig. 1 Needs for modeling of atomic cascades in different fields of physics. The stepwise excitation and decay of ions in different charge states typically lead to spectra and observations, from which one wishes to extract further information about the interaction of the underlying matter and fields. The concept of such atomic cascades indeed arises in many research areas and can help to make their computational treatment more uniform

of such cascades feasible [19, 20]. Because of the inherent complexity of most, if not all, cascades, little has been done so far in the literature to deal with them systematically, nor that codes for doing so would be readily available. Apart from an initial placement of electrons into the Rydberg shells of some ion, perhaps, all cascade schemes are typically associated with electrons in the continuum, where the release of further Auger electrons may lead to (very) complex shell structures.

To facilitate a more systematic account of atomic cascades, we here aim to identify and classify the central elements and schemes in imitating the stepwise excitation and de-excitation of atoms. In particular, we shall explain how these cascades can be employed within Jac, the Jena Atomic Calculator [21] that supports the computation of (a good number of) atomic properties and processes [22, 23]. Our classification of physical scenarios, the so-called cascade schemes and the indicated hierarchy of computational approaches is flexible enough to support further refinements as well as to deal with complex shell structures of the atoms and ions, for which the excitation and decay dynamics need to be modeled explicitly.

In the next section, we first take rather a technical perspective to (introduce) the concept of atomic cascades. This includes a classification of cascade schemes to encompass useful de-excitation scenarios as well as the distinction of various (cascade) approaches to deal with the fine-structure levels and the evaluation of their amplitudes. This section also explains the, in practice, very relevant separation of the cascade computations from all subsequent simulations in order to eventually obtain all the properties and spectra of interest. The implementation of this concept within the framework of Jac is then described in Sect. 3 by starting with a short reminder on some previously established features of this toolbox. In particular, this section describes the evaluation of dielectronic recombination plasma rate coefficients of beryllium-like Au^{75+} ions. A short summary and a few conclusions are finally given in Sect. 4.

In the text below, emphasis is placed on (the concept of) atomic cascades and how they may help to model physics that have been found difficult or even inaccessible in all previous atomic structure codes.

2 Atomic cascades

2.1 Modeling of cascades: building blocks

In atomic physics, the term *cascade* has been utilized to describe and understand the stepwise excitation and/or decay dynamics of free atoms or ions. In contrast to a single (atomic) transition, which connects two well-defined—initial- and final-state—levels with each other, a cascade comprises at least three or more levels of an atom or ion, and often also in a number of subsequent charge states. These levels are generally connected by different atomic processes, such as the (auto-) ionization or photon emission, owing to the inter-electronic repulsion of electrons or their (spontaneous) interaction with the radiation field.

For a large number of levels, therefore, any cascade exhibits an immanent complexity, even if one wishes to consider only the major excitation and/or decay paths [24–26]. Aside from the representation of the atomic levels and multiplets involved, this complexity arises first of all from the number of quantum-mechanically possible decay paths, whose relative importance has been found difficult to estimate in advance. It is this computational effort, at one hand, that suggests to distinguish and keep the computation of the electronic structure and transition amplitudes between individual levels separate from any "postprocessing" of these amplitudes that is required to finally attain the properties and spectra of interest. The consequent use of these quantum amplitudes, on the other hand, also enables one to formulate (and implement) all cascade computations in a language that remains close to the *formal* theory [27].

The stepwise transition of atoms from one level (and charge state) to the next can be modeled most readily by means of so-called *pathways*, i.e., a sequence of three or more levels that brings the ion from some well-defined initial to any of the final levels, as allowed within the given model. Any two subsequent levels in such a pathway are hereby connected by just *one* process and, hence, by the set of associated (many-electron) transition amplitudes. In our discussion below, these levels are often referred to as initial, (one or several) intermediate as well as the final level of the pathway. With this definition, any pathway of the cascade just comprises a particular sequence of steps, independent of how likely this chain will appear in the evolution of the ions. Already from this short discussion, however, it becomes obvious that a rather

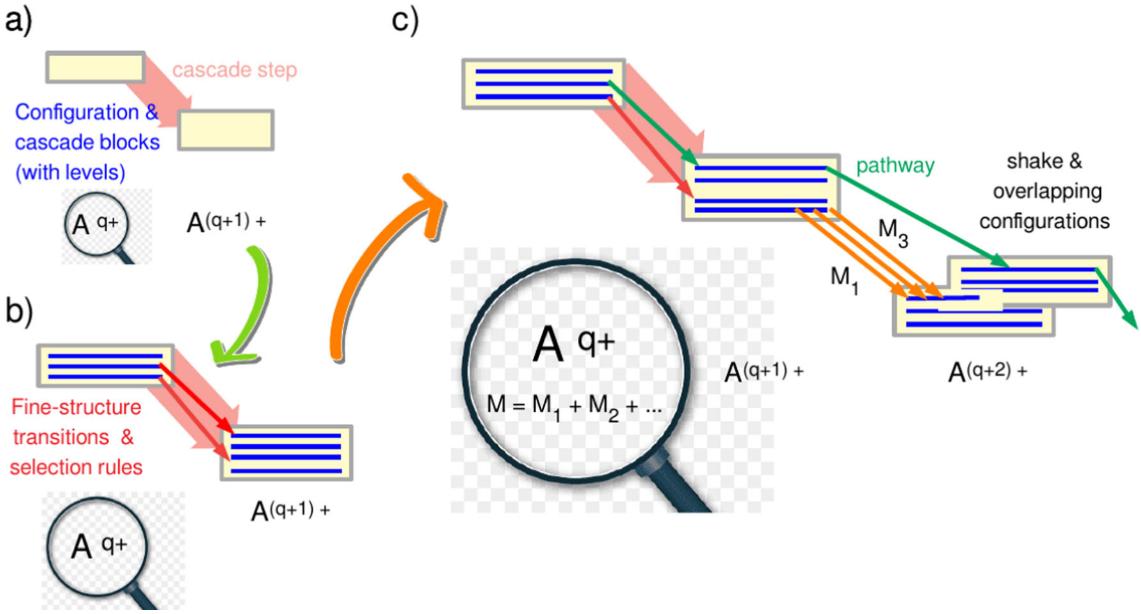


Fig. 2 A schematic and simplified view on the central building blocks of atomic cascades. **a** At moderate magnification, a cascade just appears as energetically ordered list of electron configurations, the so-called cascade blocks (yellow boxes). Pairs of these configurations are connected stepwise by different processes that lead to the absorption or emission of photons and electrons. If the magnification increases, **b** first the transitions between individual fine-structure levels (red and green arrows) and, eventually, **c** even the individual channels (orange lines) and their associated many-electron amplitudes, $\mathcal{M} = \mathcal{M}_1 + \mathcal{M}_2 + \cdots$ become visible. Moreover, different processes can connect the same fine-structure levels and then leads to different pathways

large number of pathways will naturally occur for most schemes, and that this is independent of whether a cascade arises from either the capture of initially free electrons or just owing to the stepwise decay of inner-shell holes [28–30].

Figure 2 displays the central ingredients of cascades by revealing the fine structure of all atomic transitions. Indeed, these ingredients serve as building blocks in the implementation below and will enable us to easily distinguish between different cascade schemes. They also help deduce necessary simplifications in modeling a given cascade, if its computational complexity increases too rapidly. In the atomic shell model (Fig. 2a, and seen with the lowest magnification in glasses), the occupation of the (sub-) shells is simply compiled in terms of electron configurations, such as $1s^2 2s^2 2p^6 3s^2$ for the ground configuration of atomic magnesium and magnesium-like ions in order to characterize the gross structure of the electrons. The use of electron configurations is hereby very helpful to describe all basic features of a cascade and to split it into well computable parts. Usually, these lists of configurations are generated automatically and (may) serve also as starting points to *group* configurations into the (so-called) cascade blocks.

From a refined viewpoint (Fig. 2b), each electron configuration generally comprises—one or several—levels that are slightly split in energy and are known as the fine structure of the ions. Here, inner-shell holes typically contribute much stronger to the fine-structure splitting than the valence-shell electrons and, hence, to the predicted properties of a cascade. At this magnification in glasses, further details are resolved: Each cascade is now formed by a list of energetically sorted levels (blue lines in yellow boxes), as associated with each block of the cascade, which are connected by fine-structure transitions (red and green arrows) between the same or just neighbored charge states. Each atomic transition connects two well-defined initial and final levels but, more often than not, also contain different channels (sublines; cf. Fig. 2c), for instance, due to the occurrence of different multipoles in the decomposition of the radiation field or a partial-wave expansion of an outgoing electron wave. The need of all these transitions (amplitudes) also highlights the role of the *pathways* (green arrows) as a well-ordered sequence of levels in order to make the computation of cascades transparent.

To conclude this brief discussion, any atomic cascade can always be modeled by a set of pathways, together with the (nonzero) quantum amplitudes that *pairwise* relate the levels in each pathway. From these amplitudes, the rates and cross sections can then be readily derived whenever required. The definition of the pathways above will also enable us to account for further processes in future, such as two-photon transitions, radiative or double Auger emissions, or several others, if they appear significant and if their rates (cross sections) can be estimated efficiently. Furthermore, such an initial analysis of a cascade helps the user to grasp the "size" of cascade computations and to decide, which of these computations are still feasible (and which perhaps not).

Table 1 Cascade schemes as (partly) implemented and supported by the JAC toolbox

Cascade scheme and brief explanation
Dielectronic capture (scheme): to model the formation of doubly excited levels due to the resonant capture of one additional electron, and up to a maximum excitation (= free-electron) energy with regard to the ground level of the original ion: $A^{q+} + e^- \longrightarrow A^{(q-1)+*}$; cf. `DielectronicCaptureScheme()`
Dielectronic recombination: to model both the formation of and radiative stabilization due to the resonant capture of one additional electron. All doubly excited levels are taken into account for the excitation from a given list of subshells, and up to a maximum excitation energy [32–34]: $A^{q+} + e^- \longrightarrow A^{(q-1)+*} \longrightarrow A^{(q-1)+(*)} + \hbar\omega$; cf. `DielectronicRecombinationScheme()`
Electron excitation: to model electron excitation cross sections and rate coefficients including both the direct EIE and resonant contributions due to the dielectronic capture of an electron with subsequent re-autoionization: $A^{q+} + e_i^- \longrightarrow A^{q+*} + e_f^-$ & $A^{q+} + e_i^- \longrightarrow A^{(q-1)+(*)} \longrightarrow A^{q+*} + e_f^-$; cf. `ElectronExcitationScheme()`
Electron ionization: to model electron ionization cross sections and rate coefficients, including the direct EII and resonant contributions to the ionization of atoms and ions. Resonant contributions may arise from the EIE with subsequent autoionization and from the dielectronic capture of an electron with subsequent double autoionization: $A^{q+} + e_i^- \longrightarrow A^{(q+1)+*} + e_r^- + e_f^-$ & $A^{q+} + e_i^- \longrightarrow A^{q+(*)} + e_f^- \longrightarrow A^{(q+1)+*} + e_r^- + e_f^-$ & $A^{q+} + e_i^- \longrightarrow A^{(q-1)+} \longrightarrow A^{(q+1)+*} + e_r^- + e_f^-$; cf. `ElectronIonizationScheme()`
Hollow-ion (scheme): to model the photoemission and autoionization of ions, if one or several electrons are initially placed in different (Rydberg) subshells [35]; cf. `HollowIonScheme()`
Impact excitation: to model the direct electron-impact excitation (EIE): $A^{q+} + e_i^- \longrightarrow A^{q+*} + e_f^-$ in terms of collisions strengths or EIE cross sections [36,37]; cf. `ImpactExcitationScheme()`
Impact ionization: to model the direct electron-impact ionization (EII; collision strength) of atoms: $A^{q+} + e_i^- \longrightarrow A^{(q+1)+*} + e_r^- + e_f^-$; cf. `ImpactIonizationScheme()`
Photoabsorption: to model photoabsorption spectra, including the direct photoionization as well as resonant contributions which arise from the photoexcitation of the target atom and its subsequent autoionization: $A^{q+} + \hbar\omega \longrightarrow A^{(q+1)+*} + e_p^-$ & $A^{q+} + \hbar\omega \longrightarrow A^{q+*} \longrightarrow A^{(q+1)+*} + e_p^-$; cf. `PhotoAbsorptionScheme()`
Photoexcitation: to model the photoexcitation of atoms: $A^{q+} + \hbar\omega \longrightarrow A^{q+(*)}$; cf. `PhotoExcitationScheme()`
Photoionization: to model the direct photoionization of atoms: $A^{q+} + \hbar\omega \longrightarrow A^{(q+1)+*} + e_p^-$; cf. `PhotoIonizationScheme()`
Radiative recombination: to model the (direct) radiative capture of one electron into different shells and levels of the recombined ion, up to a maximum kinetic energy of the initially free electron [38]: $A^{q+} + e_i^- \longrightarrow A^{(q-1)+*}$; cf. `RadiativeRecombinationScheme()`. Unlike in the dielectronic capture scheme above, *no* further excitation of the initially bound electrons is considered here
Stepwise decay: to model the decay of an (inner-shell) hole state by subsequent photon and/or electron emission, until a given number of electrons have been released and/or the ions cannot further decay to any lower level: $A^{q+*} \longrightarrow A^{q+(*')} + \hbar\omega \longrightarrow \ldots$ & $A^{q+*} \longrightarrow A^{(q+1)+} + e_a^- \longrightarrow \ldots \longrightarrow A^{(q+r)+} + e_{a_1}^- + \ldots + e_{a_r}^-$ as well as mixtures of these decay paths. This scheme usually assumes a prior excitation of the ion that is not part of the subsequent decay dynamics but need to be treated independently; cf. `StepwiseDecayScheme()`

These schemes are internally distinguished by different (concrete) data `types` <: `Cascade.AbstractCascadeScheme` to model the behavior and to branch into different parts of the code

2.2 Cascades schemes

A systematic account of atomic cascades requires first of all to distinguish and select between different *schemes* that are suitable to model the relevant processes altogether in the evolution of the atoms and ions. A scheme hereby describes a physical scenario that is typical for a whole class of observations, such as the photoabsorption or photorecombination of ions, or the photon and electron (emission) spectra of atoms following their inner-shell excitation. Arguably, the stepwise decay of an inner-shell excited atom or ion is the simplest form of such a cascade scheme, even if it may give rise to quite different observations. Apart from the emission spectra above, the stepwise decay of inner-shell holes typically leads also to a characteristic ion distribution, the formation (and decay) of metastable levels, different coincidence spectra as well as several other observations. All these characteristic features can be traced back to rather similar (or even the same) set of transitions data and make the stepwise decay a *prototype* for further cascade scenarios.

Indeed, other cascade schemes exist and have been found relevant for case studies in astro and plasma physics. Examples refer to the dielectronic recombination (DR) as well as the electron-impact excitation (EIE) and ionization (EII) schemes. In the DR scheme, a positive ion captures an electron at various resonant energies to form doubly excited levels, which subsequently stabilize under photon emission. Both the resonant electron capture and the following photon emission are associated with quantum amplitudes and may result in a (very) large number of different pathways. The overall DR spectra, cross sections and plasma rate coefficients therefore require to evaluate many of these pathways, though dependent on the range and distribution of the available electrons, the accuracy of the requested rate coefficients, the shell structure of ions or, not least, the computational power and efficiency at user's side.

Several of these cascade schemes, like the impact excitation of atoms and ions, even form a hierarchy as they can be induced either directly by EIE or indirectly via the dielectronic capture of an electron and with a subsequent re-autoionization into a different level. Table 1 lists the presently identified and—at least, partly—implemented schemes which have been added to the JAC toolbox since our initial setup of cascades [31]. These schemes can be modeled in JAC and further refined with regard to the number of pathways, the kind of allowed amplitudes and processes, or the properties to be considered in future.—Other characteristic schemes will be added in future, if their need arises and if the associated cascade computations remain overall feasible.

2.3 Cascades approaches

Typically, most of the computational effort in modeling atomic cascades arises from the large (or even huge) number of fine-structure transitions and the computation of all associated (many-electron) amplitudes. As outlined above, the fine structure of atoms rapidly increases with the number of open shells, i.e., if additional inner-shell electrons are excited or released due to a photoexcitation or photoionization, or some autoionization. For ions with multiple inner-shell holes, therefore, several thousand or even more of such amplitudes might, in practice, be required to just proceed a few steps ahead in the cascade (decay). This is in strong contrast to almost all previous atomic-structure calculation, which are often based on just a handful of such many-electron transition amplitudes.

Below, we shall mainly apply the multi-configuration Dirac–Hartree–Fock (MCDHF) and configuration–interaction methods to make the wave-function expansions explicit for all levels involved in a cascade. Not much needs to be said about these methods which have been described elsewhere [39–42]. Single- and/or multi-configuration expansions of wave functions have the great advantage that they help formulate any cascade right from the beginning in terms of many-electron *amplitudes* (matrix elements), and that these methods can be equally applied also to open-shell structures across the periodic table.

Nonetheless, the high computational demands in dealing with the fine-structure (transition) amplitudes of any cascade make it necessary and desirable to distinguish between various cascade *approaches* and to formulate a hierarchy of useful conditions and relations among them. These approaches mainly differ in the generation and use of the orbital functions, the setup of the Hamiltonian matrix or the treatment of electronic correlations (configuration mixing) for the representation of the atomic states. Of course, the use of these approaches should allow for a systematic improvement of the overall cascade models, in spite of all remaining limitations. Apart from incorporating further quantum processes and steps into an atomic cascade, improvements in the modeling usually refer to the representation of the atomic states and, hence, to either a clear reduction or increase of the required computational time.

In the JAC toolbox, a hierarchy of four cascade *approaches* has been defined in order to deal with different cascade scenarios. These four approaches mainly determine, up to which extent interelectronic correlations are taken into account. They have been described in Ref. [31] and can be controlled within this toolbox by providing a proper set of parameters prior to the computations. These four approaches were designed in a manner that (i) all cascade models should be computable in at least the simplest approach and (ii) that the "costs" of any of these approaches appear (almost) negligible, when compared with the next better approach in this hierarchy.

In brief, the four cascade approaches in JAC can be summarized as follows [31]:

(a) Average single-configuration approach (`Average SCA`):
This is the simplest one, which approximates all atomic levels by means of single-configuration state function (CSF) and a *common* set of orbitals as generated from the mean field of the initial atom or ion. This approach neglects indeed all configuration mixing between the bound-state levels and, in addition, restricts (by default) the computations to the Coulomb interaction among the electrons, the electric-dipole (E1) transition amplitudes as well as to just a single set of continuum orbitals for each *step* of the cascade. While this `AverageSCA` <: `Cascade.AbstractCascadeApproach` approach appears feasible for (almost) all atoms and ions from the periodic table, further work is needed to better understand how well this (very) simplified approximation can indeed help to describe the physics of the various schemes.

(b) Single-configuration approach (`SCA`):
This approach still applies a *common* set of orbitals in order to represent the atomic levels of the same stage of ionization. It includes configuration mixing within each (cascade) block but not the interaction among different blocks. Whereas the other limitations are quite similar to the `AverageSCA` approach above, all continuum orbitals are generated here for the correct fine-structure (transition) energies [43].

(c) Multiple-configuration approach (`UserMCA`):

This approach has not been implemented yet but will facilitate the incorporation of configuration mixings (electron–electron correlations) between user-selected blocks. This approach helps the user to *group* different configurations together, based on physical insight into the particular cascade process [44].

(d) Multiple-configuration shake approach (`ShakeMCA`):

This has not been designed in detail yet but aims to incorporate also *shake-down* and *shake-up* configurations. These *shake* configurations can then be treated technically either as individual blocks of configurations or together with other configurations of the decay cascade. No attempt has yet been undertaken in Jac to make this approach explicit for any of the given schemes in Table 1.

Some further refinement of these approaches is usually possible by choosing different physical parameters, e.g., by considering additional processes, other multipoles or simply by a better treatment of the continuum orbitals. For most ions, moreover, the mean computational effort is expected to increase by (at least) one or even several orders of magnitudes in going from one approach to the next, more sophisticated one. Apart from the implementation of additional schemes, any further modeling of cascades will notably depend on the full implementation, test and use of these different approaches.

2.4 Atomic cascade computations versus simulations

Atomic cascades are known to affect physical observations in very different ways. The large variety of experimental setups, measurements and outcomes make it generally impractical to implement a dedicated "cascade model" separately for each experiment. Often, different outcomes are recorded with the same setup but by taking different runs of the experiment (with regard to geometry, resolution, or the properties of the emitted particles).

This variety of possible outcomes suggests to separate the expensive computation of the (many-electron) transition amplitudes from all subsequent post-processing of the atomic data as required in order to derive the information and spectra of interest. In our modeling of cascades, we make this distinction explicit by separating the cascade *computations* from the subsequent simulation(s). The cascade computations first generate all necessary transition amplitudes and usually cause the major effort since they demand a proper self-consistent representation of all atomic levels involved. These computations eventually result in one or several lists of fine-structure (resolved) "lines" as associated with the considered processes and steps of the cascade, which then serves as interface to all subsequent cascade simulations.

Cascade simulations, in contrast, are usually stimulated and implemented in close contact with experiment. They have steadily increased in number with regard to the properties that can be derived from the code. Over the years, a "library" of different (simulation) properties have been considered in Jac. Without any right of completeness, these simulations can be divided into a few groups, including:

Photon and electron spectra:
To generate and display the effective cross sections or relative intensities of photons and electrons as function of energy.

Coincidence spectra:
The same as above but for measuring the (relative) electron–electron or electron–photon intensities. In principle, these spectra could be extended to multiple coincidences [45–47], although little work has been done so far along these lines.

Ion distributions: Such distributions can be determined either for selected initial levels or by taking the average of all levels from a leading hole-state configuration. There have been several measurements of ion charge distributions following inner-shell photoexcitation and ionization [48–51].

Population dynamics of atomic levels:
Different methods (including an iteration of the level population; rate equations; Monte Carlo) can be applied in order to propagate the initial level population through a cascade tree and to derive all spectra of interest.

Plasma rate coefficients: to readily describe the mean rate of the ionization or capture of electrons in a plasma. Such plasma rate coefficients

$$\alpha^{\,(\text{process})} \propto \sqrt{2\pi\, T}^{\,-3/2} \int d\varepsilon \varepsilon \sigma(\varepsilon;\, \text{process}) e^{-\frac{\varepsilon}{k\,T}}$$

are implemented for the radiative and dielectronic recombination of ions in plasma [52,53] and could be established also for photo- and electron-impact ionization. Apart from plasma rates, such rate coefficients can be defined also for storage rings, ion traps and elsewhere.

Time-dependent spectra:
To simulate the mean relaxation time after which a given percentage of the initial population of ions has relaxed into their ground configurations. Such time-dependent spectra can be used also to simulate the photon emission in crossed-beam experiments.

The cascade simulations are usually much faster than the prior computations, and often similar simulations are performed repeatedly for different photon energies or populations of the initial levels in order to account for different resonances. Below, we shall explain some of these simulations in further detail in order to indicate possible applications of the Jac toolbox. This short discussion of feasible and (partly) implemented properties also underlines the added value of the separation between cascade computations and simulations.

3 Implementation of atomic cascades within Jac

3.1 The Jac toolbox

Several, though for sure not all, of the cascade schemes and approaches from above have been realized during recent years within Jac, the Jena Atomic Calculator. This toolbox facilitates atomic (structure) calculations of different kinds and complexity and hence enables the user to compute level energies, energy shifts, or a good number of rates and cross sections. In particular, Jac applies configuration interaction and multi-configuration Dirac–Hartree–Fock wave functions to represent all atomic states that occur during the excitation or capture processes [22,40]. Moreover, since Jac is based on Dirac's relativistic equation, this toolbox is suitable also for medium and heavy elements. It also provides a flexible user interface and a quite simple communication between the different parts of the program. The Jac toolbox has been described elsewhere [21] and can be downloaded from the web [54], including the present upgrade.

Jac's implementation is based on Julia, a new programming language for scientific computing. This language includes a number of (modern) features, such as dynamic types, optional type annotations, type-specializing, just-in-time compilation of code, dynamic code loading as well as garbage collection, to name just a few [55,56]. In this language, functions mainly operate on data types and data structures. When compared with other programming languages, Julia's type system is known, in particular, as one of its strongest features, in which abstract data types help establish a hierarchy of relationships between data and actions and, hence, to model *behavior* of the code by just providing (concrete) subtype arguments.

A primary goal in developing Jac was to provide an easy-to-use and powerful platform in order to extend atomic theory toward the modeling of atomic cascades. For atomic cascades, as explained above, it is crucial to integrate different atomic processes within a single computational framework and to ensure good (self-) consistency of the generated data. To enlarge Jac's capabilities, we here work through and expand the data structure `Cascade.Computation` to enable the user to specify a (cascade) scheme and approach, along with all further input parameters. This follows in spirit our first implementation [21], in which the data structure `Atomic.Computation` was introduced to calculate the energies and representation of atomic levels, together with a number of transition amplitudes and characteristic properties.

3.2 Role of a descriptive language

The conceptional and computational complexity of most atomic cascades require first of all to map out (and to make use of) a concise language that combines a proper categorization of the given scenario with a modern code design, a reasonably detailed documentation of the code as well as features for integrated testing [57,58]. In particular, the computations as well as the (generated) data should both be handled internally similar to how they are used in daily work, In short, this language needs to be *descriptive*, that is: (i) user-friendly, (ii) emphasizing the underlying atomic physics and (iii) by avoiding most technical jargon, as very typical for most established codes from the literature. Moreover, such a language should be (iv) suitable for interactive work and (v) simple enough for both frequent and seldom use of the code. Together with the Jac toolbox, we here aim to provide such a language in order to support cascade computations for most, if not all, atoms and ions across the periodic table.

For modeling atomic cascades, in particular, such a descriptive language must account for (and fulfill) the following requests: (a) It has to support atoms and ions with several open shells, (b) deal quite equally with different atomic properties and processes, (c) give access to different models and approximations and, last but not least, (d) help to convert the physical units to the needs of the user, at both input and output time. In addition, a descriptive atomic language should support also a transparent data flow with and within the program, independent of the particular cascade scheme.

To achieve these goals, well-designed data structures are indeed crucial for any modern implementation and, explicitly also, for dealing with atomic cascades. These structures are used as (language) elements in order to describe and control the desired computations, with a notion that is readily understandable to most atomic physicists (and potential users of the code). For electronic structure calculations, selected (examples of such) data types are an `Orbital` to represent the quantum numbers and radial components of the (one-electron) orbital functions, an atomic `Basis` to specify a set of many-electron CSF, or even a `Level` for the full wave-function representation of a single atomic state (level) function E, $|\alpha \mathbb{J} M\rangle$ as they appear in the setup and evaluation of all transition amplitudes [27,40].

Here, we shall not discuss these data structures in any detail. In total, Jac is built upon $\lesssim$ 300 of these data structures, though most of them remain hidden to the user. A few, which are central to the implementation of atomic cascades in Jac, are shown and briefly explained in Table 2.

3.3 Implemented cascades schemes

In Sect. 2.2 above, we have discussed a number of physical scenarios (schemes) that can be understood and treated—more or less readily—as atomic cascades. These scenarios differ by the number and sequence of excited, ionized or captured electrons and usually support (very) different observations and communities. They are designed to encapsulate different atomic processes and a good deal of further sophistication, if applied to complex open-shell atoms and ions. The photoabsorption (scheme) of atoms, for example, should

Table 2 Selected data structures of the Jac toolbox that are relevant for cascade computations and simulations

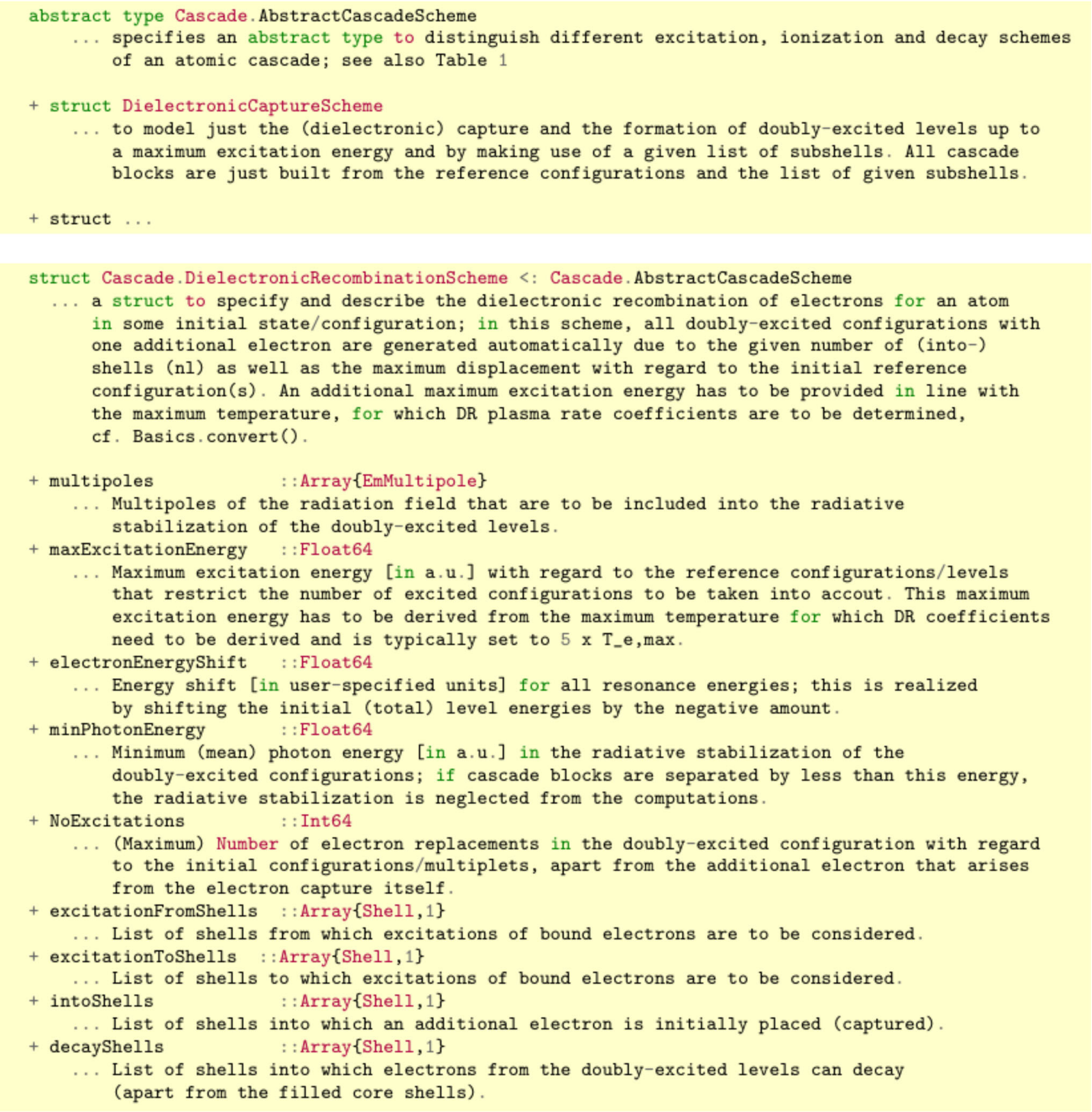
`struct` and brief explanation
`Cascade.AbstractCascadeApproach`: defines an abstract type for dealing with the cascade approach that is applied to the generation and evaluation of all many-electron amplitudes; it includes the (concrete) subtypes `Cascade.AverageSCA, SCA, ...`; cf. Sect. 2.3
`Cascade.AbstractCascadeScheme`: defines an abstract data type to discriminate between different excitation, ionization and decay schemes of an atomic cascade; see also Table 1 and Fig. 3
`Cascade.AbstractSimulationMethod`: an abstract data type to deal with the level occupation and properties that can be "simulated" from the given line list, for instance, `ProbPropagation` or `MonteCarlo`
`Cascade.AbstractSimulationProperty`: an abstract type to discriminate between and specify the property or distribution that need to be simulated, based on given `data::Cascade.Data` set; cf. Table 3 for allowed subtypes
`Cascade.AverageSCA`: a singleton type to specify the average SCA (cascade) approach. In this approach, all levels in the cascade are described in single-configuration and single-CSF approximation; cf. Sect. 2.3
`Cascade.Block`: defines a type for an individual block (or list) of configurations that are treated together within the cascade as they occur in the representation of the atomic levels
`Cascade.Computation`: defines a data structure for specifying the computation of a photoexcitation, photoionization, stepwise decay or several other cascade computations; this is the major data structure to be set (handled) by the user
`Cascade.Data`: defines a parametric type for communicating different line lists to the cascade simulations
`Cascade.DrRateCoefficients<:Cascade.AbstractSimulationProperty`: defines a type for simulating the DR plasma rate coefficients as function of the (free) electron energy and plasma temperature
`Cascade.DielectronicRecombinationScheme<:Cascade.AbstractCascadeScheme`: a struct to define and describe the dielectronic recombination of electrons for an atom in some initial state/configuration; cf. Fig. 3
`Cascade.Level`: defines a (simplified) level specification for dealing with fine-structure transitions in all cascade simulations
`Cascade.LineIndex`: defines a line index with regard to the various line lists of type `data::Cascade.Data`
`Cascade.Simulation`: a structure to deal with all simulations, based on given cascade data. This type is the counterpart to `Cascade.Computation`
`Cascade.SimulationSettings`: defines a data structure to specify all additional parameters for controlling a cascade simulation
`Cascade.Step`: a type for dealing with an individual step of an excitation and/or decay cascade; cf. Fig. 2. Internally, a step is determined by two lists of initial- and final-state configurations as well as by the associated atomic process
`Cascade.StepwiseDecayScheme<:Cascade.AbstractCascadeScheme`: a type to represent and generate a stepwise decay cascade; cf. Table 1

comprise the direct photoionization of electrons, the resonant photoexcitation of an inner-shell electron with subsequent autoionization as well as, perhaps, several other photoscattering processes. All these processes formally lead to the absorption of photons.

Whereas the distinction of different cascade schemes is reasonably straightforward, a "classification" of atomic cascades in terms of the finally observed spectra and distributions is much less feasible owing to the large number of possible measurements and experimental setups. As already seen for the photoabsorption (scheme) above, most cascade scenarios can be implemented only stepwise with regard to both the detailed modeling and the range of possible applications. We here aim to implement and support the cascades schemes from Table 1, and which are internally distinguished by different concrete types in order to branch into specific parts of the code. The use of an abstract type hereby helps to add new cascade schemes without the need to modify any code of the previously implemented schemes.

Whereas the "basic concept" of atomic cascades has been introduced before [31], let us here re-organize and expand the foregoing code. Table 1 briefly summarizes the currently supported cascade schemes in Jac. To this end, the goal was to identify and support further physical scenarios that can benefit from an efficient treatment of different atomic processes within the Jac program.—With this expansion, however, we also wish to open this concept to other applications as they may arise from the user's side.

Not all of the cascades schemes in Table 1 are (yet) equally supported, though. Two frequently applied cascade schemes are the `StepwiseDecayScheme` and `DielectronicRecombinationScheme`. To illustrate their internal use, Fig. 3 displays the data structure for dealing with the dielectronic recombination of atoms. Apart from the multipoles of the radiation field, this cascade scheme enables one, for example, to select the shells that can be occupied in the formation of the doubly excited levels. For these levels, to be generated internally, we distinguish between the `fromShells` and `toShells` for the excitation of the bound electron(s) as well as the `intoShells` for the capture of an initially free electron. To facilitate the computations, moreover, the user has to select the `decayShells`, into which electrons from the doubly excited levels can stabilize radiatively. In addition to these shell lists, the user can provide further information about the maximum excitation energy of the doubly excited levels with regard to the initial ground level, an energy shift of all resonances as well as the (maximum) number of the simultaneously



```
abstract type Cascade.AbstractCascadeScheme
    ... specifies an abstract type to distinguish different excitation, ionization and decay schemes
        of an atomic cascade; see also Table 1

+ struct DielectronicCaptureScheme
    ... to model just the (dielectronic) capture and the formation of doubly-excited levels up to
        a maximum excitation energy and by making use of a given list of subshells. All cascade
        blocks are just built from the reference configurations and the list of given subshells.

+ struct ...
```

```
struct Cascade.DielectronicRecombinationScheme <: Cascade.AbstractCascadeScheme
  ... a struct to specify and describe the dielectronic recombination of electrons for an atom
      in some initial state/configuration; in this scheme, all doubly-excited configurations with
      one additional electron are generated automatically due to the given number of (into-)
      shells (nl) as well as the maximum displacement with regard to the initial reference
      configuration(s). An additional maximum excitation energy has to be provided in line with
      the maximum temperature, for which DR plasma rate coefficients are to be determined,
      cf. Basics.convert().

+ multipoles             ::Array{EmMultipole}
    ... Multipoles of the radiation field that are to be included into the radiative
        stabilization of the doubly-excited levels.
+ maxExcitationEnergy    ::Float64
    ... Maximum excitation energy [in a.u.] with regard to the reference configurations/levels
        that restrict the number of excited configurations to be taken into accout. This maximum
        excitation energy has to be derived from the maximum temperature for which DR coefficients
        need to be derived and is typically set to 5 x T_e,max.
+ electronEnergyShift    ::Float64
    ... Energy shift [in user-specified units] for all resonance energies; this is realized
        by shifting the initial (total) level energies by the negative amount.
+ minPhotonEnergy        ::Float64
    ... Minimum (mean) photon energy [in a.u.] in the radiative stabilization of the
        doubly-excited configurations; if cascade blocks are separated by less than this energy,
        the radiative stabilization is neglected from the computations.
+ NoExcitations          ::Int64
    ... (Maximum) Number of electron replacements in the doubly-excited configuration with regard
        to the initial configurations/multiplets, apart from the additional electron that arises
        from the electron capture itself.
+ excitationFromShells   ::Array{Shell,1}
    ... List of shells from which excitations of bound electrons are to be considered.
+ excitationToShells   ::Array{Shell,1}
    ... List of shells to which excitations of bound electrons are to be considered.
+ intoShells             ::Array{Shell,1}
    ... List of shells into which an additional electron is initially placed (captured).
+ decayShells            ::Array{Shell,1}
    ... List of shells into which electrons from the doubly-excited levels can decay
        (apart from the filled core shells).
```

Fig. 3 Internal definition of the `Cascade.AbstractCascadeScheme` (upper panel) and `Cascade.DielectronicRecombinationScheme` (lower panel) within the Jac toolbox. Whereas the abstract data type helps control the behavior of the code, the concrete type below specifies and communicates all data that are needed for modeling the DR strengths and plasma rate coefficients. Default values are utilized for all subfields, if not overwritten explicitly by the user

allowed excitation of bound electrons. Analogue data structures have been introduced to deal also with cascade schemes.

3.4 Cascade computations and implemented approaches

The cascade computations have been separated from the subsequent simulations with the aim to formalize the modeling of cascades and to make them (computational) feasible. They require and need to generate a self-consistent field and a representation for all the levels which contribute to the overall cascade. In a cascade computation, therefore, the given scheme, the occupation of the ions in terms of reference configurations as well as the computational approach is first analyzed for the requested (elementary) steps and processes, i.e., by making the underlying cascade tree explicit.

Internally, the cascade tree is always handled by (lists of) configurations that are generated automatically. This generation of the tree is followed by the electronic structure calculations of all (cascade) blocks as well as the evaluation of the associated transition amplitude. If spoken more technically, any cascade computation can be seen also as a sequence of `Atomic.Computation`'s which are stepwise performed in order to generate (and collect) the lines "lists" of the associated transition amplitudes.

Use of cascade computations:
Like the atomic computations in Jac, a `Cascade.Computation` has been defined in order to support a simple control of the physical scheme, the approach and all the computational steps of a cascade. The upper panel of Fig. 4 displays the internal definition of this data structure in terms of its subfields that need to be specified by the user. Beside a name (string), the nuclear model and the radial grid, a `Cascade.Computation` is set out first of all by its `scheme::Cascade.AbstractCascadeScheme` and `approach::Cascade.AbstractCascade Approach` as it has been explained in the last two subsections.
Specifying the initial state of a cascade:
The initial state of the ion (ensemble) can be drawn up either by a list of initial configurations `(initialConfigs)` or multiplets `(initial Multiplets)`, though not both at a given time. These configuration define the "roots" of the cascade tree, from which all other configurations, levels, pathways, etc., can (and have to be) derived automatically.
Refinement of a cascade:
A refinement of this cascade tree is typically possible by choosing different physical parameters, e.g., by considering more processes, other multipoles or simply by a better treatment of the continuum for each individual approach.
Combination of cascade computations:
Sometimes, several of these cascade computations have to be performed before all the necessary transition data are indeed generated. This applies, for instance, if one wishes to combine an initial excitation of the atom or ion with its subsequent decay [59,60]. In all these cascade computations, however, no attempt is made internally in order to decide that all relevant data are actually computed *nor* that all computed data are relevant for the simulations below. The proper choice and control of the cascade computations require physical perception by the user and cannot be formalized completely.
Results from a cascade:
Any cascade computation returns one (or a few) "line" lists which comprise lines of the same type, although in arbitrary order with regard to their transition energies, level specification, or even the charge state of the ions. These "line" lists are not only expensive but can be (very) lengthy; they are usually stored on disk and form the "input" for all subsequent cascade simulations. These lists are instances of the parametric type `Cascade.Data{T}` to facilitate the data communication between the different parts of the code and to keep the transition data available for subsequent simulations.

3.5 Implemented cascade simulations

Insight into the setup of the experiment, its geometry, target preparation, detectors or into possible limitations of the measurements is required in order to predict and quantify the outcome of any measurement theoretically. This is particularly true if atomic cascades affect the observations. This request makes it also necessary to typically adopt the simulation to the given setup and to carefully control all of its input.

The variety of experimental setups also results in a rather large number of properties and spectra which can be derived by means of cascade simulations. Table 3 compiles and briefly explains several of the frequently required properties and distributions as (partly) implemented in the Jac toolbox. They are shown in this table along with their associated concrete data
`type` <: `Cascade.AbstractSimulationProperty` in order to distinguish them internally and to select the right code at execution time. These concrete types are employed also to transmit the relevant input and control parameters to Jac.

Not many details need to be told here about the list of properties and spectra, which (may) cover a wide range of applications and whose number will further increase in future. Julia's help facilities [for example: `? Cascade. AbsorptionCrossSection`] can always be invoked in order to obtain further information from the doc-strings that are provided for each data type. These doc-strings list and describe all parameters (subfields) that are used to make the requested data type explicit. The list of synthesized properties in Table 3 can be readily expanded by additional data types in order to refine the requests from experiment, if further parameters (subfields) are introduced and implemented into the code.

From the various properties in Table 3, let us just consider here the simulation of plasma rate coefficients as they are associated with the electron-impact ionization, the radiative and dielectronic recombination of electrons, or the photoionization by means of a blackbody radiation field. For these rate coefficients, one typically starts from a list of transition amplitudes between the ground configuration (level) and a large number of excited states, that contribute to the given process. These transition amplitudes are first combined into cross sections and are later convoluted with the state distribution of electrons or photons, as they are accessible in the plasma.

Further (simulation) properties will likely be supported in the future, if requested by the users. For example, modern photon and electron spectroscopy [61–63] often aim for coincidence measurements to

```
struct Cascade.Computation
  ... defines a type for a cascade computation, i.e. for the computation of a given (cascade)
      scheme and approach. The -- input and control -- data from this computation can be
      modified, adapted and refined to the practical needs, and this need to be done prior to
      the actual computations are carried out explictly. Initially, this struct just comprises the
      physical (meta-) data about to describe the cascade.

+ name               ::String             ... A name for the cascade
+ nuclearModel       ::Nuclear.Model      ... Model, charge and parameters of the nucleus.
+ grid               ::Radial.Grid        ... The radial grid to be used for the computation.
+ asfSettings        ::AsfSettings        ... Provides the settings for the SCF process.
+ scheme             ::Cascade.AbstractCascadeScheme
    ... Scheme of the atomic cascade such as the StepwiseDecayScheme; cf. Table 1.
+ approach           ::Cascade.AbstractCascadeApproach
    ... Computational approach/model that is applied to generate and evaluate the cascade;
        possible approaches are: {AverageSCA(), SCA(), ...}
+ initialConfs       ::Array{Configuration,1}
    ... List of one or several configurations that specifies the level(s) from which the cascade
        tree is derived  and from which the computation starts.
```

```
struct Cascade.Simulation
  ... specifies a simulation of some particular property by using the given cascade data.

+ name             ::String                       ... Name of the simulation.
+ property         ::Cascade.AbstractSimulationProperty
    ... Property that is to be considered in this simulation; cf. Table 3.
+ method           ::Cascade.AbstractSimulationMethod
    ... Method that is used in the cascade simulation; cf. Cascade.SimulationMethod.
+ settings         ::Cascade.SimulationSettings   ... Settings for performing these simulations.
+ computationData  ::Array{Dict{String,Any},1}    ... Data on which the simulations are performed.
```

```
struct Cascade.DrRateCoefficients <: Cascade.AbstractSimulationProperty
  ... defines a type for simulating the DR plasma rate coefficients as function of the plasma
      temperature and for a given level of the initial ion.

+ initialLevelNo       ::Int64
    ... Level No of initial level for which rate coefficients are to be computed.
+ electronEnergyShift  ::Float64
    ... (total) energy shifts that apply to all resonances when alpha^(DR) is computed.
+ temperatures         ::Array{Float64,1}
    ... temperatures [K] for which the DR plasma rate coefficieints to be calculated.
+ nDetailed            ::Int64
    ... principal quantum number of the 'last' shell for which Auger and radiatiative amplitudes
        have been calculated by the Cascade.Computation; all contributions of this shell are
        scaled for nDetailed < n <= nMax also for higher shells by a simple scaling rule.
+ nMax                 ::Int64
    ... Maximum n (principal quantum number), for which contributions are scaled; NO scaling
        is taken into account for nMax <= nDetailed.
+ resonanceSelection   ::Dielectronic.ResonanceSelection
    ... To select individual classes of resonances to be included into the plasma rate
        coefficients.
```

Fig. 4 Definition of the Cascade.Computation (upper panel) and Cascade.Simulation (middle panel) as well as the `Cascade.DrRateCoefficients` within the JAC toolbox. See text for further discussion

help reveal dominant decay channels or level structures, which cannot be accessed so readily otherwise. In particular, it will be desirable to generate two-dimensional coincidence maps of photon and/or electron intensities (photon–photon, photon–electron and electron–electron coincidence), for which very different *cuts* through the coincidence data can be addressed. Moreover, plasma rate coefficients for other processes might be worthwhile to consider as well.

Table 3 Properties, spectra and distributions that can be obtained from cascade simulations

Property and brief explanation
Absorption cross sections: for an atom in some given ground-state configuration as function of the incident photon energy; cf. `Cascade.AbsorptionCrossSection`
Dominant decay paths: to extract and report the major "decay paths" of a stepwise (decay) cascade in terms of the electron configurations involved; cf. `Cascade.DecayPaths`
DR plasma rate coefficients: to simulate $\alpha^{\,(\mathrm{DR})}\,(T;\, i)$ for a Maxwellian distribution of the free plasma electrons and the given temperature; cf. `Cascade.DrRateCoefficients`
Electron spectrum: to synthesize an electron spectrum as function of the electron energy for a given stepwise decay cascade; cf. `Cascade.ElectronIntensities`
Final-level distribution: to determine the distribution of the finally occupied levels by following the stepwise decay cascade of one or several initially populated level(s); cf. `Cascade.FinalLevelDistribution`
Ion distribution: to determine the distribution of ionic charge states by following the stepwise decay cascade of one or several initially populated level(s); cf. `Cascade.IonDistribution`
Mean line width: to determine the *mean* width of photon lines, for instance, if several lines overlap in a photon spectrum owing to different charge states, etc.; cf. `Cascade.MeanLineWidth`
Opacity estimation: to estimate, for a given ionic charge state, the expansion opacity as function of the incident photon frequency or the Rosseland opacity as function of temperature; cf. `Cascade.ExpansionOpacity,` `Cascade.RosselandOpacity`
Photo absorption spectrum: to synthesize a photoabsorption spectrum as function of the incoming photon energy; cf. `Cascade.PhotoAbsorptionSpectrum`
Photon spectrum: to synthesize a photon spectrum as function of the photon energy for a given stepwise decay cascade; cf. `Cascade.PhotonIntensities`
RR plasma rate coefficients: to simulate $\alpha^{\,(\mathrm{RR})}\,(T;\, i)$ for a Maxwellian distribution of the free plasma electrons and the given temperature; cf. `Cascade.RrRateCoefficients`

These properties or distributions are (partly) implemented and supported by the JAC toolbox. Here, they are listed alphabetically and explained in brief, along with their associated concrete `type <: Cascade.AbstractSimulationProperty` in order to distinguish them internally. See text for further discussion

3.6 Example: DR plasma rate coefficients for Au^{75+} ions

The resonant DR process has been found crucial for understanding many observations in astro and plasma physics. In this two-step process $A^{\,q+} \,+\, e^- \;\longrightarrow\; A^{\,(q-1)+\,*} \;\longrightarrow\; A^{\,(q-1)+\,(*)} \;+\; \hbar\omega$, an ion in the charge state $q+$ first captures resonantly an electron from the continuum and forms a doubly excited ion of the next lower charge state. In a second step, then, the excited ion either stabilizes under photon emission or re-autoionizes back to its original charge state. In multiply and highly charged ions, DR often competes with the (non-resonant) RR as the latter process gives rise to the same final states. More formally, we can describe the DR process of an N-electron target ion in terms of its initial level i, the $[(N+1)$-electron$]$ resonance level d as well as the (set of) final levels $\{f\}$ below of the re-ionization threshold of the recombined ion.

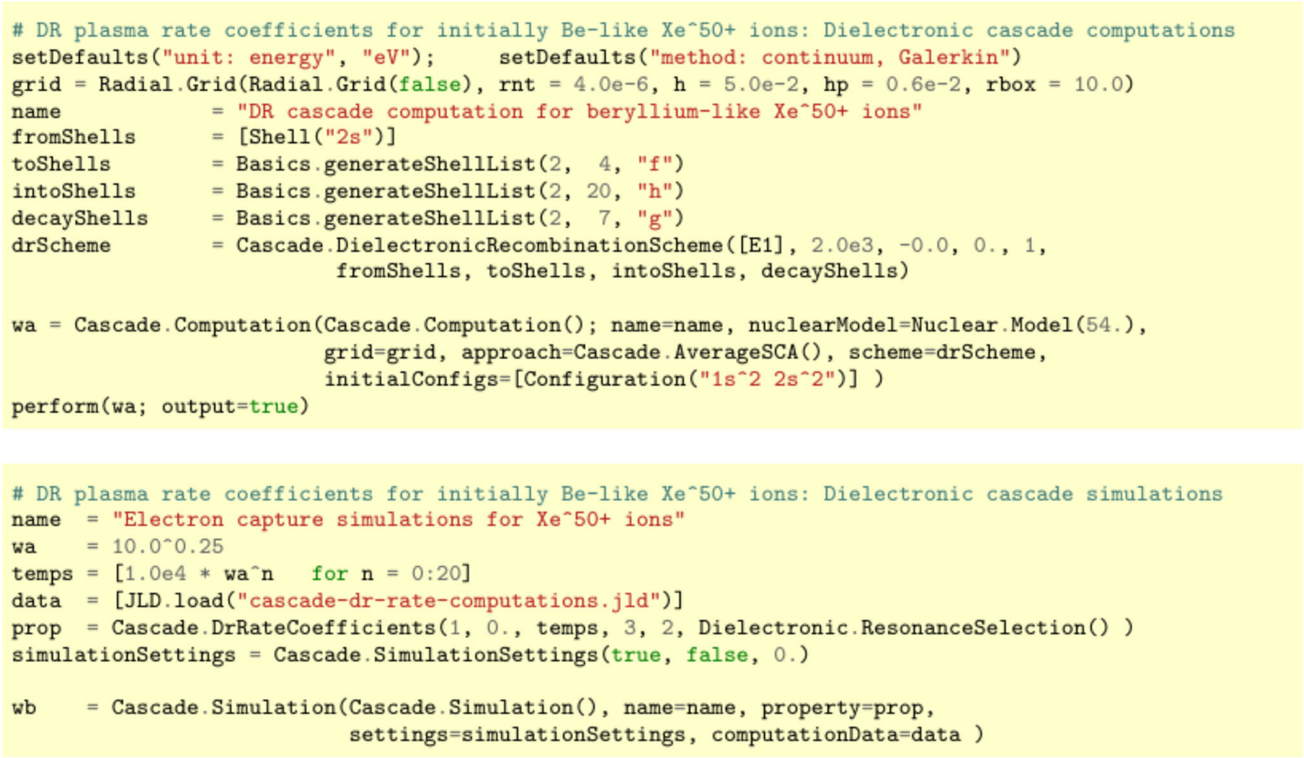
For a single ion per unit volume in the initial level i, the DR plasma rate coefficient $\alpha^{\,(\mathrm{DR})}\,(T_e;\, i)$ describes the rate to recombine at plasma temperature T_e due to the impact by a Maxwellian distributed electron. This rate coefficient can be readily expressed also as a sum over partial DR rate coefficients $\alpha^{\,(\mathrm{DR})}\,(T_e;\, i \,\to\, d)$ as associated with the electron capture into some well-defined resonance level d,

$$\alpha^{\,(\mathrm{DR})}\,(T_e;\, i) = \sum_d \alpha^{\,(\mathrm{DR})}\,(T_e;\, i \to d)$$

$$\alpha^{\,(\mathrm{DR})}\,(T_e;\, i \to d) = \frac{4}{(k_B\,T_e)^{\,3/2}\,\sqrt{2\pi\, m}}\,(E_d \,-\, E_i) \times S(i \to d)\, \exp\left(-\,\frac{E_d - E_i}{k_B\,T_e}\right) \qquad \left[\frac{\mathrm{cm}^3}{\mathrm{s}}\right] .$$

Here, $S(i \,\to\, d)$ is the (so-called) resonance strength and $\varepsilon_d \,=\, E_d \,-\, E_i$ the kinetic energy of the initially free electron that helps to form the resonance. Whereas the summation over d runs formally over all valence and inner-shell excited resonances of the $[(N+1)$-electron$]$ recombined ion, the number of resonances itself can be readily truncated owing to the temperature-dependent factor $\exp\left(-\,\frac{E_d - E_i}{k_B\,T_e}\right)$ to energies $\varepsilon_d \,<\, \varepsilon_d^{\,(\max)}$ of the free electron. Below, we shall use the maximum free-electron energy $\varepsilon_d^{\,(\max)}$ as one of the control parameters in the associated cascade computations. In practice, it has been found sufficient to include all resonances up to, say, 5–10 times the "mean" free-electron energy in the plasma in order to account for all relevant contributions from high-lying resonances.

Indeed, DR plasma rate coefficients have been calculated for many ions along different isoelectronic sequences. For beryllium-like ions, for instance, Colgan and coworkers [64] applied the R-matrix method in order to compute DR plasma rate coefficients for low- and medium-Z ions with $Z \,\leq\, 54$. For these ions, three types of excitations can be distinguished owing to the $\Delta n = 0$ and $\Delta n \neq 0$ valence-shell excitation as well as



```
# DR plasma rate coefficients for initially Be-like Xe^50+ ions: Dielectronic cascade computations
setDefaults("unit: energy", "eV");     setDefaults("method: continuum, Galerkin")
grid = Radial.Grid(Radial.Grid(false), rnt = 4.0e-6, h = 5.0e-2, hp = 0.6e-2, rbox = 10.0)
name            = "DR cascade computation for beryllium-like Xe^50+ ions"
fromShells      = [Shell("2s")]
toShells        = Basics.generateShellList(2,  4, "f")
intoShells      = Basics.generateShellList(2, 20, "h")
decayShells     = Basics.generateShellList(2,  7, "g")
drScheme        = Cascade.DielectronicRecombinationScheme([E1], 2.0e3, -0.0, 0., 1,
                          fromShells, toShells, intoShells, decayShells)

wa = Cascade.Computation(Cascade.Computation(); name=name, nuclearModel=Nuclear.Model(54.),
                         grid=grid, approach=Cascade.AverageSCA(), scheme=drScheme,
                         initialConfigs=[Configuration("1s^2 2s^2")] )
perform(wa; output=true)
```

```
# DR plasma rate coefficients for initially Be-like Xe^50+ ions: Dielectronic cascade simulations
name  = "Electron capture simulations for Xe^50+ ions"
wa    = 10.0^0.25
temps = [1.0e4 * wa^n   for n = 0:20]
data  = [JLD.load("cascade-dr-rate-computations.jld")]
prop  = Cascade.DrRateCoefficients(1, 0., temps, 3, 2, Dielectronic.ResonanceSelection() )
simulationSettings = Cascade.SimulationSettings(true, false, 0.)

wb    = Cascade.Simulation(Cascade.Simulation(), name=name, property=prop,
                           settings=simulationSettings, computationData=data )
```

Fig. 5 Input to the JAC toolbox for the cascade computation (upper panel) and simulation (lower panel) of DR plasma rate coefficients for beryllium-like Xe^{50+} ions. Here, rate coefficients are calculated for plasma temperatures $T_e \lesssim 10^9$ K, and by just including $2s \rightarrow 2p, \ldots, 4f$ valence-shell excitation with an electron capture into the $n\ell$ shells ($n = 2 \ldots, 20, \ell \leq h$). For the radiative stabilization, moreover, we here restrict the computations to the $n\ell$ shells with $n \leq 7$

the $1s \rightarrow n\ell$ inner-shell excitation. At low plasma temperatures, the DR plasma rate coefficients $\alpha^{(\mathrm{DR})}(T; i)$ is entirely determined by $2s \rightarrow 2p$ excitations, while the $2s \rightarrow 3\ell + 4\ell$ excitations become relevant at plasma temperatures $T \gtrsim 2 \cdot 10^7$ K. Finally, the $1s \rightarrow n\ell$ excitations are irrelevant for medium and heavy elements at all practical temperatures.

In JAC, we can readily perform analogue calculations for the DR plasma rate coefficients by following the settings of Fig. 5. This figure displays all the necessary input for both the required `Cascade.Computation` (upper panel) and `Cascade.Simulation` (lower panel); cf. also Fig. 4. Apart from some technical input with regard to a name, nuclear model ($Z = 54$), the radial grid as well as the configuration of the initial ion (as summarized before in Sect. 3.4), we need to specify the allowed shells for the excitation of the ionic core (`fromShells, toShells`), the capture of the initially free electrons (`intoShells`) as well as for their subsequent radiative stabilization (`decayShells`). From these shell lists, all the possible (5-electron) configurations are then generated automatically and their mean energies are compared with regard to $\varepsilon_d^{(\max)}$, i.e., the maximum excitation energy of the resonances that has to be considered in the cascade computations.

Let us briefly remind here to the need of a reasonable representation of the doubly excited levels for the calculation of reliable DR strength and rate coefficients. This request refers especially to the total energies of the resonances, which are taken with regard to the initial ground level. As seen from Fig. 5, moreover, we still apply here for the sake of feasibility (at some notebook) the `AverageSCA` approach and include all resonances up to $2 \cdot 10^3$ Hartree as well as single excitations of the core electrons with regard to the initial reference configuration. For setting down this (and other) cascade computations interactively, a detailed explanation of all the input parameters can be obtained also from `? Cascade.Dielec tronicCaptureScheme`.

For the subsequent cascade simulations in the lower panel of Fig. 5, we just specify `Cascade.DrRateCoefficients(..)` as the property of interest, along with the initial level number of the ground level, a possible (energy) shift of all resonances with regard to the initial ground level as well as the temperatures, for which the $\alpha^{(\mathrm{DR})}(T_e; i)$ coefficients need to be determined. Individual contributions to the DR plasma rate coefficients could be analyzed by selecting certain classes of resonances, `cf. ResonanceSelection()`, but is set inactive in the present case. These simulations are quite fast, when compared with the corresponding computations, and simply return the plasma rate coefficients for the given temperatures (and transition data). Of course, analogue simulations can be performed for other temperatures of the plasma or, in principle, for also deriving the relative photon spectra that arise in course of the radiative stabilization.

Figure 6 compares the DR plasma rate coefficients from the present computations and simulations (red solid line) for beryllium-like Xe^{50+} ions with data by Colgan et al. ([64]; blue dashed line) and computations with the `autostructure` code ([65]; green dashed-dotted line). A reasonable agreement is found for these rate coefficients, although differences up to an order of magnitude occur especially at low plasma temperatures. This is likely related to the incomplete representation of the $2p\,5\ell$ resonances close to the autoionization threshold. Deviations of this order are still not rare for DR rate coefficients, and further work is needed to better combine the feasibility of computations with their required accuracy [66].

Similar computations can be performed also for beryllium-like Au^{75+} ions. Such heavy few-electron ions are of interest from a fundamental point of view (cf. Ref. [66]) since relativistic and quantum-electrodynamical (QED) effects scale with large powers of the nuclear charge. Electron-ion recombination experiments with beryllium-like Au^{75+} ions will be conducted in the near future at the heavy-storage ring CRYRING [30], which is part of the Facility for Antiproton and Ion Research (FAIR) in Darmstadt, Germany. The experiments will concentrate on low electron-ion collision energies, for which the experimental resolving power is highest. At these energies, the dominant recombination channels involve the capture of the initially free electron into Rydberg levels with principal quantum numbers of up to 20.

For a first overview, the left panel of Fig. 7 compares the DR plasma rate coefficients for the capture into different shell lists: Apart from a computation quite analogue to Figs. 5 and 6 ($n \leq 20$, $\ell \leq h$; red solid line), we display rate coefficients from a restricted computation with ($n \leq 10$, $\ell \leq h$; blue dashed line) as well as, again, `autostructure` calculation ([65]; green dashed-dotted line). At low temperatures, obviously, the plasma rate coefficients are strongly affected by the correct position of the low-lying resonances, which are treated differently in the various implementations.

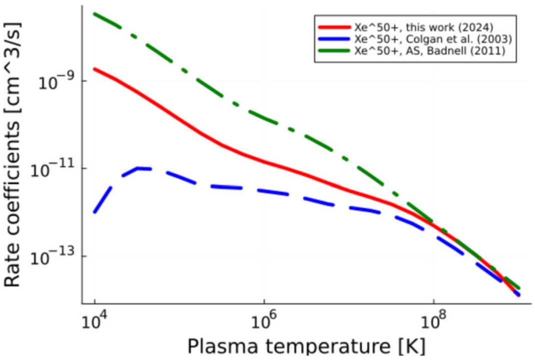


Fig. 6 DR plasma rate coefficients $\alpha^{(\mathrm{DR})}(T_e;\ i)$ for beryllium-like Xe^{50+} as function of temperature T [K]. Results from this work (red solid line) are compared with the calculations by Colgan et al. ([64]; blue dashed line) and computations with the `autostructure` code ([65]; green dotted line)

We can explore also the contributions of selected (subsets of) resonances by using the features of the `Dielectronic.ResonanceSelection(...)` data structure with proper parameters. Here, we shall not explain details of this selection (procedure) but refer the reader to the Julia help facilities [`?...`]. The right panel of Fig. 7 displays the results of the corresponding simulations; it compares the contributions of the $2s \to 2p$ excitation (red solid line) with those due to a $2s \to 3s + 3p + 3d$ (blue dashed line) and $2s \to 4s + 4p + 4d + 4f$ excitation (green dotted line), respectively. Of course, other classes could be selected as well. As seen from this figure, the $2s \to 2p$ is dominant at low temperatures but becomes small for temperatures $T_e \gtrsim 10^8$ K. Both panels of Fig. 7 nicely demonstrate how quite different information can be extracted from these cascade simulations, once a sufficiently large computation was performed, and without that further computations need to be done.

4 Summary and conclusions

A classification of atomic cascades into experimentally relevant schemes is presented to model the excitation and decay of atoms and ions under quite different circumstances. Apart from the stepwise de-excitation of inner-shell excited ions, this classification includes several frequently occurring physical scenarios, such as photoabsorption, photoionization or selected electron-impact excitation processes. It also includes dedicated schemes to facilitate the calculation of (plasma) rate coefficients of various kind, expansion opacities or the generation of synthetic spectra. To this end, this classification of atomic cascades is based on a clear distinction between cascade computations, to first generate all necessary transition data, and the subsequent simulations, in order to properly combine these data and to support comparison with experiment.

While a few of these schemes have already been implemented in Jac, the Jena Atomic Calculator, further work is needed to exploit and appreciate the features of other schemes, for instance, in order to predict the attenuation of radiation or the population dynamics of atoms and ions in different environments. Indeed, the (complete) accomplishment of these cascade schemes is a much larger task than what we can (and intend to) implement in the present version of Jac. Since this toolbox is available on GitHub [54], the code can be readily forked and expanded toward more complex cascades (schemes). Moreover, further work on a selective (quantum-mechanical) representation of the level structures, the so-called cascade approaches in Jac, will help to better understand the role of atomic interactions and inter-electronic correlations upon the observed properties and spectra.

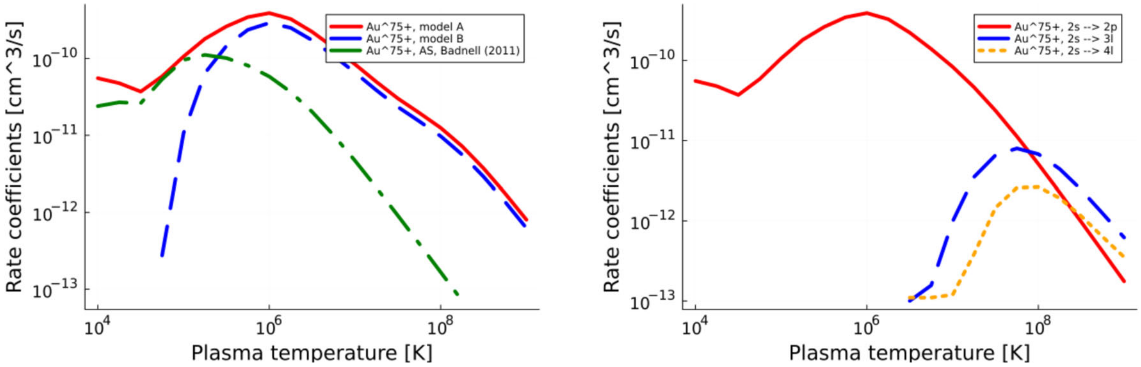


Fig. 7 DR plasma rate coefficients $\alpha^{(\mathrm{DR})}(T_e;\, i)$ for beryllium-like Au^{75+} as function of temperature T [K]. Left: Rate coefficients from a large-scale computation with electrons captured into all shells with $n \leq 20$, $\ell \leq h$ (model A; red solid line) are compared with data from a reduced computation ($n \leq 10$, $\ell \leq h$; model B; blue dashed line) as well as an autostructure calculation ([65]; green dashed-dotted line). Right: Contributions to the total rate coefficients due to different (classes of) excitation: $2s \rightarrow 2p$ excitation (red solid line); $2s \rightarrow 3s{+}3p{+}3d$ (blue dashed line) and $2s \rightarrow 4s{+}4p{+}4d{+}4f$ excitation (orange dotted line), respectively

When compared with other explicitly correlated methods, our classification and implementation in Jac helps overcome previous difficulties with the generation and application of a sufficiently large number of final states and/ or electron continua in the treatment of electron or photon-impact processes. In addition, the design of this toolbox enables the user to consider a number of other elementary processes or to deal with a combination of these processes, high multipoles in the coupling of atoms to the radiation field, a simplified control of the continuum, a proper choice of units as well as several other benefits. All these features make the Jac toolbox suitable for modeling astrophysical spectra or to support surveys over a wide range of impact energies and/or ions. For this, further emphasis will be placed mainly on those features that receive attention by the community.

Acknowledgements Financial support by the German Ministry for Education and Research (BMBF) via the Collaborative Research Center APPA (ErUM-FSP T05, grant nos. 05P21RGFA1 and 05P21SJFAA) is gratefully acknowledged.

Author contributions

SF designed the concept and implementation of this expansion to Jac and wrote the original draft. AKS and LS helped implementing test suites. ZWW and SS reviewed and revised the manuscript. All of the authors have read and agreed to the published version of the manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability Statement This manuscript has no associated data or the data will not be deposited. [Authors' comment: All data in Figs. 6 and 7 can be reconstructed by means of the JAC toolbox, provided at the website [54].]

References

1. W.R. Bennett, P.J. Kindlmann, Radiative and collision-induced relaxation of atomic states in the $2p^5 3p$ configuration of neon. Phys. Rev. **149**, 38 (1966)
2. E. Veje, Possible additions to the spectra of core-excited neutral potassium, rubidium, and cesium. Phys. Rev. A **36**, 1474 (1987)
3. W.J. Fu, G.H. Wang, Y.X. Liu, Electron capture and its reverse process in hot and dense astronuclear matter. Astrophys. J. **678**, 1517 (2008)
4. D. Shaikh, G.P. Zank, A energy cascades in a partially ionized astrophysical plasma. Astrophys. J. **688**, 683 (2008)
5. V.A. Sreckovic, M.S. Dimitrijevic, L.M. Ignjatovic, N.N. Bezuglov, A.N. Klyucharev, The collisional atomic processes of Rydberg hydrogen and helium atoms: astrophysical relevance. Galaxies **6**, 72 (2018)
6. G.E. Ice, G.S. Brown, G.B. Armen, M.H. Chen, B. Crasemann, J. Levin, D. Mitchell, Atomic inner-shell

threshold excitation with synchrotron radiation. AIP Conf. Proc. **94**, 105 (1982)
7. A. Langereis et al., Stabilization of electrons on Ar^{q+} ions after slow collisions with C$_{60}$. Phys. Rev. A **63**, 062725 (2021)
8. E. Träbert, Lifetime measurements of highly charged ions. Phys. Scr. **2002**(T100), 88 (2002)
9. E. Träbert, J. Hoffmann, C. Krantz, A. Wolf, Y. Ishikawa, J.A. Santana, Atomic lifetime measurements on forbidden transitions of Al-, Si-, P- and S-like ions at a heavy-ion storage ring. J. Phys. B **42**, 025002 (2009)
10. Y. Zhang, X.-W. Liu, R. Wesson, P.J. Storey, Y. Liu, I.J. Danziger, Electron temperatures and densities of planetary nebulae determined from the nebular hydrogen recombination spectrum and temperature and density variations. Mon. Notes R. Astron. Soc. **351**, 935 (2004)
11. J.J. Bochanski et al., The most distant stars in the Milky Way. Astrophys. J. Lett. **790**, L5 (2014)
12. Y. Ezoe, T. Ohashi, K. Mitsuda, High-resolution X-ray spectroscopy of astrophysical plasmas with X-ray microcalorimeters. Rev. Mod. Plasma Phys. **5**, 2367 (2021)
13. W.L. Wiese, Spectroscopic diagnostics of low temperature plasmas: techniques and required data. Spectrochim. Acta B **46**, 831 (1991)
14. B. Saha, S. Fritzsche, Influence of dense plasma on the low-lying transitions in Be-like ions: relativistic multiconfiguration Dirac–Fock calculation. J. Phys. B **40**, 259 (2007)
15. J. Limburg, J. Das, S. Schippers, R. Hoekstra, R. Morgenstern, Coster–Kronig transitions in hollow atoms created during highly charged ion-surface interactions. Phys. Rev. Lett. **73**, 786 (1994)
16. R.A. Wilhelm et al., Interatomic Coulombic decay: the mechanism for rapid deexcitation of hollow atoms. Phys. Rev. Lett. **119**, 103401 (2017)
17. M.-H. Chan et al., Tunable single-atom nanozyme catalytic system for biological applications of therapy and diagnosis. Mat. Today Adv. **17**, 100342 (2023)
18. C. Battistini, T. Bensby, The origin and evolution of r- and s-process elements in the Milky Way stellar disk. Astron. Astrophys. **586**, A49 (2016)
19. S. Fritzsche, Large-scale accurate structure calculations for open-shell atoms. Phys. Scr. **T100**, 37 (2002)
20. A.P. Chaynikov, A.G. Kochur, A.I. Dudenko, V.A. Yavna, Cascade energy reemission after inner-shell ionization of atomic gold. Role of photo- and cascade-produced electrons in radiosensitization using gold-containing agents. J. Quant. Spectrosc. Radiat. Transf. **302**, 108561 (2023)
21. S. Fritzsche, A fresh computational approach to atomic structures, processes and cascades. Comput. Phys. Commun. **240**, 1 (2019)
22. S. Fritzsche, The RATIP program for relativistic calculations of atomic transition, ionization and recombination properties. Comput. Phys. Commun. **183**, 1525 (2012)
23. K. Verhaegh et al., An improved understanding of the roles of atomic processes and power balance in divertor target ion current loss during detachment. Nucl. Fusion **59**, 126038 (2019)
24. M.M. Jakas, Many-body effects in atomic-collision cascades. Phys. Rev. Lett. **55**, 1782 (1985)
25. S. Schippers et al., Near L-edge single and multiple photoionization of singly charged iron ions. Astrophys. J. **849**, 5 (2017)
26. R. Beerwerth et al., Near L-edge single and multiple photoionization of triply charged iron ions. Astrophys. J. **887**, 189 (2019)
27. S. Fritzsche, Application of symmetry-adapted atomic amplitudes. Atoms (Basel) **10**, 127 (2022)
28. N.M. Kabachnik, S. Fritzsche, A.N. Grum-Grzhimailo, M. Meyer, K. Ueda, Coherence and correlations in photoinduced Auger and fluorescence cascades in atoms. Phys. Rep. **451**, 155 (2007)
29. J. Andersson et al., Triple ionization of atomic Cd involving $4p^{-1}$ and $4s^{-1}$ inner-shell holes. Phys. Rev. A **92**, 023414 (2015)
30. M. Lestinsky et al., Physics book: CRYRING@ESR. Eur. Phys. J. Spec. Top. **225**, 797 (2016)
31. S. Fritzsche, P. Palmeri, S. Schippers, Atomic cascade computations. Symmetry (Basel) **13**, 520 (2021)
32. N.R. Badnell, Radiative recombination data for modeling dynamic finite-density plasmas. Astron. Astrophys. Suppl. Ser. **167**, 334 (2006)
33. S. Fritzsche, A. Surzhykov, T. Stöhlker, Dominance of the Breit interaction in the X-ray emission of highly charged ions following dielectronic recombination. Phys. Rev. Lett. **103**, 113001 (2009)
34. S. Fritzsche, Dielectronic recombination strengths and plasma rate coefficients of multiply charged ions. Astron. Astrophys. **656**, A163 (2021)
35. S. Fritzsche, Photon emission from hollow ions near surfaces. Atoms (Basel) **10**, 37 (2022)
36. L. Sharma, A. Surzhykov, R. Srivastava, S. Fritzsche, Electron-impact excitation of singly charged metal ions. Phys. Rev. A **83**, 062701 (2011)
37. S. Fritzsche, L.G. Jiao, Y.C. Wang, J.E. Sienkiewicz, Collision strengths of astrophysical interest for multiply charged ions. Atoms (Basel) **11**, 80 (2023)
38. S. Fritzsche, A.V. Maiorova, Z.W. Wu, Radiative recombination plasma rate coefficients for multiply charged ions. Atoms (Basel) **11**, 50 (2023)
39. I.P. Grant, Relativistic effects in atoms and molecules, in *Methods in Computational Chemistry*, vol. 2, ed. by S. Wilson (Plenum, New York, 1988), pp.1–38
40. I.P. Grant, *Relativistic Quantum Theory of Atoms and Molecules: Theory and Computation* (Springer, Berlin, 2007)
41. S. Fritzsche, Ratip—a toolbox for studying the properties of open-shell atoms and ions. J. Electron Spectrosc. Relat. Phenom. **114–116**, 1155 (2001)
42. J. Bieron et al., Ab initio MCDHF calculations of electron–nucleus interactions. Phys. Scr. **90**, 054011 (2015)
43. S. Fritzsche, B. Fricke, W.-D. Sepp, Reduced L$_1$ level-width and Coster–Kronig yields by relaxation and continuum interactions in atomic zinc. Phys. Rev. A **45**, 1465 (1992)
44. C.Z. Dong, S. Fritzsche, Relativistic, relaxation and correlation effects in spectra of Cu II. Phys. Rev. A **72**, 012507 (2005)
45. J.C. Levin et al., Argon-photoion–Auger-electron coincidence measurements following K-shell excitation by synchrotron radiation. Phys. Rev. Lett. **65**, 988 (1990)

46. E. Andersson et al., Formation of Kr^{3+} via core-valence doubly ionized intermediate states. Phys. Rev. A **85**, 032502 (2012)
47. Y. Hikosaka, S. Fritzsche, Coster–Kronig and super Coster–Kronig transitions from the Xe $4s$ core-hole state. Phys. Chem. Chem. Phys. **24**, 17535 (2022)
48. S. Schippers et al., Prominent role of multielectron processes in K-shell double and triple photodetachment of oxygen anions. Phys. Rev. A **94**, 041401(R) (2016)
49. A. Perry-Sassmannshausen et al., Multiple photodetachment of carbon anions via single and double core-hole creation. Phys. Rev. Lett. **124**, 083203 (2020)
50. S. Schippers et al., Near L-edge single and multiple photoionization of doubly charged iron ions. Astrophys. J. **908**, 52 (2021)
51. A. Müller et al., Role of L-shell single and double core-hole production and decay in m-fold ($1 \leq m \leq 6$) photoionization of the Ar^{+} ion Phys. Rev. A **104**, 033105 (2021)
52. D.A. Verner, G.J. Ferland, Atomic data for astrophysics. I. Radiative recombination rates for H-like, He-like, Li-like and Na-like ions over a broad range of temperature. Astrophys. J. Suppl. Ser. **103**, 467 (1996)
53. O. Zatsarinny et al., Dielectronic recombination data for dynamic finite-density plasmas. IX. The fluorine isoelectronic sequence. Astron. Astrophys. **447**, 379 (2006)
54. S. Fritzsche, JAC: User Guide, Compendium & Theoretical Background. https://github.com/OpenJAC/JAC.jl. Unpublished (10.01.2024)
55. https://docs.julialang.org/ (10.01.2024)
56. J. Bezanson et al., Julia: dynamism and performance reconciled by design. Proc. ACM Program. Lang. **2**, 120 (2018)
57. M. Mernik, J. Heering, A.M. Sloane, When and how to develop domain-specific languages. ACM Comput. Surv. (CSUR) **37**, 316 (2005)
58. V. Sousa, V. Amaral, B. Barroca, Towards a full implementation of a robust solution of a domain specific visual query language for HEP physics analysis. IEEE Nucl. Sci. Symp. Conf. Rec. **1**, 910 (2007)
59. M. Kitajima et al., Experimental and theoretical study of the Auger cascade following $3d \rightarrow 5p$ photoexcitation in krypton. J. Phys. B **34**, 3829 (2001)
60. F. Da Pieve et al., Study of electronic correlations in the Auger cascade decay from $\mathrm{Ne}^{*}1s^{-1}3p$. J. Phys. B **38**, 3619 (2005)
61. E. Andersson et al., Multielectron coincidence study of the double Auger decay of $3d$-ionized krypton. Phys. Rev. A **82**, 043418 (2010)
62. P. Lablanquie et al., Multi-electron spectroscopy: Auger decays of the argon 2s hole. Phys. Chem. Chem. Phys. **13**, 18355 (2011)
63. J. Andersson et al., Auger decay of $4d$ inner-shell holes in atomic Hg leading to triple ionization. Phys. Rev. A **96**, 012505 (2017)
64. J. Colgan, M.S. Pindzola, A.D. Whiteford, N.R. Badnell, Dielectronic recombination data for dynamic finite-density plasmas: III. The beryllium isoelectronic sequence. Astron. Astrophys. **412**, 597 (2003)
65. N.R. Badnell, A Breit–Pauli distorted wave implementation for autostructure. Comput. Phys. Commun. **182**, 1528 (2011)
66. D. Bernhardt et al., Spectroscopy of berylliumlike xenon ions using dielectronic recombination. J. Phys. B **48**, 144008 (2015)