

Symbolic Evaluation of Expressions from Racah's Algebra

Stephan Fritzsche ^{1,2,3} 

¹ Helmholtz-Institut Jena, Fröbelstieg 3, D-07743 Jena, Germany; s.fritzsche@gsi.de

² GSI Helmholtzzentrum für Schwerionenforschung, 64291 Darmstadt, Germany

³ Theoretisch-Physikalisches Institut, Friedrich-Schiller-Universität Jena, D-07743 Jena, Germany

Abstract: Based on the rotational symmetry of isolated quantum systems, Racah's algebra plays a significant role in nuclear, atomic and molecular physics, and at several places elsewhere. For N -particle (quantum) systems, for example, this algebra helps carry out the integration over the angular coordinates analytically and, thus, to reduce them to systems with only N (radial) coordinates. However, the use of Racah's algebra quickly leads to complex expressions, which are written in terms of generalized Clebsch–Gordan coefficients, Wigner n -j symbols, (tensor) spherical harmonics and/or rotation matrices. While the evaluation of these expressions is straightforward in principle, it often becomes laborious and prone to making errors in practice. We here expand JAC, the Jena Atomic Calculator, to facilitate the sum-rule evaluation of typical expressions from Racah's algebra. A set of new and revised functions supports the simplification and subsequent use of such expressions in daily research work or as part of lengthy derivations. A few examples below show the recoupling of angular momenta and demonstrate how JAC can be readily applied to find compact expressions for further numerical studies. The present extension makes JAC a more flexible and powerful toolbox in order to deal with atomic and quantum many-particle systems.

Keywords: atomic structure; nuclear structure and reactions; open-shell system; Racah algebra; recoupling coefficient; rotation matrix; spherical harmonic; symbolic evaluation; Wigner n -j symbol



Citation: Fritzsche, S. Symbolic Evaluation of Expressions from Racah's Algebra. *Symmetry* **2021**, *13*, 1558. <https://doi.org/10.3390/sym13091558>

Academic Editors: Kayn Forbes and David S. Bradshaw

Received: 10 June 2021

Accepted: 2 August 2021

Published: 25 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Racah's Algebra in Atomic and Many-Body Physics

The conservation of angular momentum is known as a powerful concept for studying the properties and (dynamical) behavior of many-particle systems. For isolated N -particle quantum systems, for example, the theories of angular momentum and irreducible tensor operators help to simplify the problem, which generally depends on $3N$ spatial (and further spin-) coordinates, to a equivalent problem with just N (radial) coordinates. In quantum theory, therefore, the calculus of angular momentum has been found indispensable if more than a (very) few particles are involved in a given system.

Though formally very powerful, the theory of angular momentum often leads to algebraic expressions that quickly become complex and cannot be handled so easily. In fact, various mathematical techniques from group theory are required to find compact expressions, suitable for further (numerical) studies. Since major parts of the underlying (mathematical) techniques were established by Racah in the early 1940s [1–3], they are briefly referred to as Racah algebra techniques today. However, while these techniques can be purposeful applied, at least in principle, they quickly become burdensome and prone to errors. In practice, therefore, the simplification of lengthy expressions from the theory of angular momentum has often constrained the further analysis of quantum many-particle systems [4,5].

Apart from the complexity of many mathematical rules, difficulties in applying Racah's algebra arise especially from the number and symmetry of the underlying symbols as well as their phases (and phase conventions). To overcome these problems, various algebraic and graphical approaches have been worked out [6–9]. Here, we suggest (and implement) a symbolic approach within the framework of JAC, the Jena Atomic Calculator, which

facilitates atomic and cascade computations of different kinds and complexity. The present extension of JAC aims at simplifying typical expressions from Racah's algebra, and which may include also the summation over dummy indices.

Like JAC in its entirety, we provide an interactive and user-friendly approach that neither requires detailed knowledge about the group-theoretical background (that leads to these expressions) nor prior experience with the techniques for their simplification. To explain the goals and strategies behind this computer-algebraic approach, we briefly recall the advantages, difficulties and the form of typical Racah expressions in Section 2, and by mentioning a few strategies for their simplification. In Section 3, we then outline the basic features of JAC, and which additional (data) structures and functions are needed in order to support the symbolic manipulation of expressions from Racah's algebra. All manipulations are based on the symmetries as well as a large number of sum and integration rules for the Wigner n -j symbols and rotation matrices that (may) help to successively reduce the length of Racah expressions. Moreover, a few examples in Section 4 demonstrate how the present extension to JAC can be applied in order to simplify recoupling coefficients or (many-particle) matrix elements and diagrams, as they frequently occur in atomic physics and/or quantum electrodynamics (QED). Finally, a short summary is given in Section 5.

2. Making Use of Racah's Algebra

2.1. Advantages of Using Racah's Algebra

After the pioneering work by Eugene P. Wigner in the late 1930s, especially Giulio Racah [1–3] developed an algebraic calculus that forms today one of the fundamentals for dealing with quantum many-particle systems. Over the years, in fact, Racah's algebra has been utilized in a large number of applications and in quite different fields of physics, from nuclear reactions [10] to atomic and molecular physics [11–13], to quantum chemistry [14], or even in solid state physics [15]. However, not many details need to be provided here about the mathematical background of angular momentum (theory), which has been the subject of a good number of textbooks and monographs [16,17]. We just refer to the text of Wigner [5] and the (very comprehensive) compilation by Varshalovich, Moskalev and Khersonskii [18].

In the (re)coupling of angular momenta, for example, and as frequently applied in nuclear and atomic structure theory, Racah's algebra leads to expressions that are usually written in terms of generalized Clebsch–Gordan coefficients and/or Wigner n -j symbols as well as the Wigner rotation matrices and spherical harmonics. In these derivations, indeed, rather a large number of these symbols (may) occur in the first instance, along with multiple summations, Kronecker and triangular deltas $\delta(j_1, j_2, j_3)$ as well as various further (reduced) coefficients. Detailed insight into the structure and symmetry of these expressions is then required to find equivalent, while more compact, expressions and finally those that are suitable for further work. It is this—usually necessary—reduction of complex expressions into an equivalent but compact form that makes Racah's algebra so powerful but also rather a great challenge, and sometimes cumbersome in practice. Furthermore, similar demands on the evaluation of expressions from Racah's algebra arise in the spin-angular integration of many-particle matrix elements [19,20], in molecular [21] and QED perturbation expansions [22], or in deriving spin- and/or angular correlations functions in quantum scattering theory [23,24], and where the rotational symmetry helps reduce the complexity. In all these cases, however, the elegance of Racah's algebra typically comes for the price that one needs to deal with quite cumbersome algebraic manipulations.

2.2. Typical Expressions and Difficulties from Racah's Algebra

To deal with typical expressions from Racah's algebra, below we shall make use of (so-called) Racah expressions.

$$\begin{aligned} \text{Racahexpr} := & \sum_{j_1, j_2, l_1, \dots} (-1)^{2j_1 - j_2 + \dots} j_1^{3/2} [j_2] \dots \begin{pmatrix} \cdot & \cdot & j_1 \\ \cdot & \cdot & \cdot \end{pmatrix} \begin{pmatrix} j_1 & j_2 & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix} \left\{ \begin{matrix} \cdot & j_3 & \cdot \\ j_1 & \cdot & \cdot \\ J & \cdot & j_2 \end{matrix} \right\} \dots \\ & \times \int d\Omega_1 Y_{\ell_1 m_1}(\Omega_1) Y_{\ell_2 m_2}(\Omega_2) \int d\beta d_{p_3 q_3}^{(j_3)}(\beta) d_{p_4 q_4}^{(j_4)}(\beta') \dots \end{aligned} \quad (1)$$

which may comprise any number of Wigner 3-j, 6-j and 9-j symbols, spherical harmonics as well as Wigner's (small) rotation matrices, and where j_1, j_2, \dots and m_1, m_2, p_3, \dots refer to the formal angular-momentum and magnetic quantum numbers, respectively. In addition, Kronecker and triangular deltas, a multiple summation over angular momenta or the integration over angles (within certain domains) may occur in these expressions. Of course, the complexity of such Racah expressions increases rapidly as more Wigner or other symbols are involved in the (product) terms. We here introduce this explicit notation of a Racah expression with the goal for providing the user with an internal data type, which facilitates automatic manipulations and which is flexible enough to support a wide range of applications.

The notion of a Racah expression above includes as special cases also the well-known orthogonality relations (and integrals) of the Wigner n-j symbols and spherical harmonics. It also enables one to readily include Clebsch–Gordan coefficients $\langle j_1 m_1, j_2 m_2 | j_3 m_3 \rangle$ and rotation matrices $D_{pq}^{(j)}(\alpha, \beta, \gamma)$ that, apart from some additional phase and/or exponential, can be expressed in terms of the Wigner 3-j symbols and the reduced rotation matrices $d_{pq}^{(j)}(\beta)$, respectively. A simple manipulation and reduction of Racah expressions (1) will therefore help in many different applications.

Of course, the overall symmetry of a Racah expression follows directly from the symmetries of its underlying Wigner n-j and other symbols. For example, each of the Wigner 6-j symbols can be (re-) written, by properly exchanging columns and rows, in 24 different but equivalent forms. Apart from this (i) very high symmetry of Racah expressions, including different phases and weight factors, further difficulties arise moreover from (ii) the complexity of many mathematical rules, which are hard to remember, as well as (iii) the number of different notations and phase conventions, and which often cause confusion when going through some derivation. With the present algebraic evaluation of such Racah expressions, we here wish to overcome (parts of) these difficulties.

2.3. Techniques for Simplifying Racah Algebra Expressions

While the theory of angular momentum is quite well understood today, the manipulation and simplification of typical Racah expressions requires experience and is often laborious in detail. As mentioned before, many standard techniques in using Racah's algebra result in quite lengthy expressions, involving a multiple summation over products of Clebsch–Gordan coefficients and/or Wigner n-j symbols. A typical goal for the algebraic manipulation is to recognize the equivalence of a certain part of such Racah algebra expression with one side of a given sum or orthogonality rules and to replace this part, so that the overall expression takes a simpler form. Obviously, this requires a careful inspection and analysis of the full Racah expression, including the summation over dummy indices, the occurrence of relevant angular momenta in other parts of the overall expression as well as the correct treatment of all phases and weight factors of the overall expression. Rather similar requirements occur, moreover, if an integration over angles should be simplified algebraically.

While the sum and orthogonality rules are usually known in some standard form from the literature [5,18], they are not of direct help without that the Wigner symbols are properly re-written and arranged, so that an algebraic equivalence can be easily recognized. In practice, this requires to cycle through all symmetric forms of a Racah expression and to analyze its individual parts for possible replacements. Four different strategies are therefore considered in the implementation below in order to simplify Racah expressions, and where a simplification refers to lowering the number of summation indices, the number of Wigner symbols and rotation matrices and/or the number of integrals in the overall expression. The algebraic evaluation of a given Racah expression then terminates if no further replacement can be found towards such a simpler expression.

In the JAC toolbox below, the four strategies for simplifying Racah expressions can be summarized as follows:

- (i) Use of so-called special values of the Wigner n -j symbols, if some of the arguments fulfill certain fixed relations. For this strategy, each Wigner n -j symbol can be analyzed separately and, if the special value applies, replaced by a simple algebraic expression, and without that the summation indices or other parts of the Racah expression has to be considered.
- (ii) Use of orthogonality and sum rules for the Wigner n -j symbols. This requires a very careful analysis of the Racah expression as a whole, since all summation variables not only need to be in the correct position in the (arguments of the) Wigner n -j symbols, but they must also contribute to a correct phase and weight of the overall expression. Moreover, the same variables may not occur in other Wigner symbols of the Racah expression, which are not part of the selected rule.
- (iii) Use of orthogonality and sum rules for the spherical harmonics and Wigner rotation matrices. Apart from requests similar to strategy (ii), the proper angular dependence of the functions involved has to be ensured for a valid simplification of the Racah expression.
- (iv) Use of integration rules; these rules can be formally treated quite similar to the former sum rules, if we assume that all sums and integrals can be interchanged in such expressions. For (iii) and (iv), only a small subset of the known sum and integration rules have so far been implemented in JAC.

Indeed, these four strategies help evaluate complex Racah expressions and often leads to some or even the most compact form. These strategies may likely fail, however, if certain phase or weight factors cannot be simplified or if arguments (angular momenta) contains ‘composed’ values, e.g., a summation or multiplication of different symbols and numbers. No attempt has been undertaken (yet) to deal with such elaborate Racah expressions, nor to deal with the symmetries due to Regge [25] for the Wigner 3-j and 6-j symbols. While the Regge symmetries are known to the JAC toolbox, they are less relevant for symbolic evaluations, though very lengthy to deal with. In JAC, we therefore keep the distinction between the classical symmetries and those due to Regge as an optional flag.

The four strategies above systemize the usual manipulation of expressions from Racah’s algebra that is often based on a good deal of practical experience and effort in order to find the proper rules and replacements. They are (partly) different from the graphical rules that have been established as well in the literature [6,26], and which make the structure of certain expressions more readily visible in course of the evaluations; however, these graphical rules require again additional knowledge and are prone to making errors as the phases and weights need often to be treated separately. Below, we shall explain how these strategies are implemented within the JAC toolbox and can now be utilized in order to solve various research and textbook problems.

3. Tools and Implementation

3.1. Brief Overview about JAC

The JAC toolbox has been developed as a general and easy-to-use toolbox for the atomic physics community in order to perform (relativistic) atomic structure calculations of different kinds and complexity. In particular, this toolbox integrates different atomic processes within a single computational framework and, thus, ensures a proper (self)consistency of the generated data. For many (standard) computations, JAC provides an interface which is equally accessible for working spectroscopists, theoreticians and code developers. The JAC tool box is based on Julia, a recent programming language for scientific computing, and can typically be applied without much prior knowledge of the code. Little needs to be said about the general design of the JAC code that has been described elsewhere [27] and that can readily be downloaded from the web [28].

The design of these tools has been based on an analysis of typical user requirements and a hierarchical structure of the code. Apart from interactive (atomic) computations, JAC also supports various representations of atomic states, including mean-field orbitals, restricted code spaces [29,30] and approximate Green function expansions [31], to name just a few. It also facilitates the computation and simulation of atomic cascades [32–34], by just following different excitation schemes. Figure 1 displays a brief overview about the JAC toolbox and its major features. Here, we expand this toolbox to now support also the symbolic evaluation of expressions from Racah's algebra. This extension makes JAC particularly suitable in order to combine derivations and computations within the same framework. It makes use of the SymEngine package [35] of the Julia platform. It also replaces and expands a former code for similar symbolic evaluations but which was limited to the Wigner n -j (and equivalent) symbols [36–38].

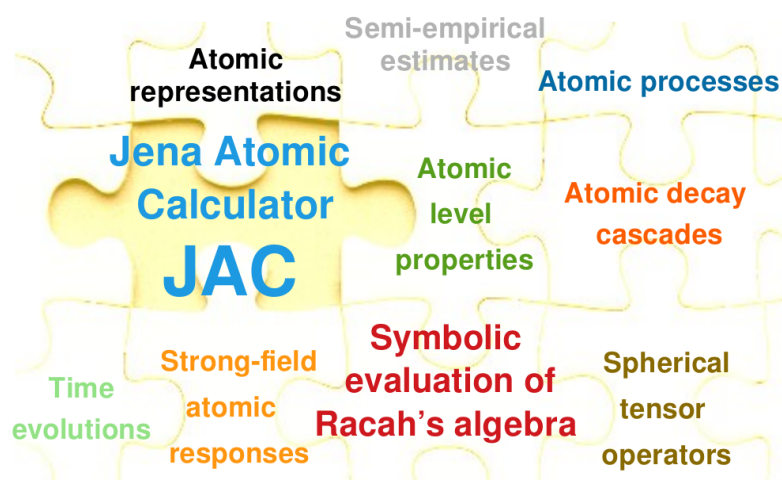


Figure 1. Overview about the JAC toolbox [27] for calculating atomic and ionic structures, processes and cascades, based on Dirac's equation. Apart from relativistic atomic computations, the JAC toolbox now supports also the symbolic evaluation of expressions from Racah's algebra. In the future, moreover, we plan to facilitate the symbolic decomposition and use of spherical tensor operators.

3.2. Data Structures for Symbolic Evaluations

Since the present extension to JAC focuses upon the symbolic evaluation of Racah expressions, i.e., their stepwise simplification towards an equivalent but more compact form, a number of new data structures were required. Apart from the Wigner n -j symbols, spherical harmonics or Wigner rotation matrices, we need to be able to deal with Racah expressions as a whole. All these data structures are defined in the module RacahAlgebra and are based on `symbols::Basic`, the central data type of SymEngine [35], though other symbolic packages might be supported as well in the future.

For example, Figure 2 displays the data structures W3j for a Wigner 3-j symbol with symbolic arguments $\begin{pmatrix} j_1 & j_2 & j_3 \\ m_1 & m_2 & m_3 \end{pmatrix}$, Ylm for a spherical harmonic $Y_{\ell m}(\vartheta, \varphi)$ as well as a RacahExpression as it is formally shown in expression (1). A few further data structures of the RacahAlgebra module (within JAC) are shown and briefly explained in Table 1. These data types facilitate the user to map the internal complexity of Racah's algebra upon the evaluations to be done. The internal definition of the data structures can be readily obtained at the Julia REPL [39] by typing, for instance,

```
? RacahExpression
```

```
struct RacahAlgebra.W3j ... a type for a Wigner 3-j symbol with symbolic arguments.

+ ja, jb, jc      ::Basic    ... angular momenta (quantum numbers).
+ ma, mb, mc      ::Basic    ... projections of the angular momenta above.

struct RacahAlgebra.Ylm ... a type for a spherical harmonic Y_lm(theta,phi) with symbolic arguments.

+ l, m           ::Basic    ... angular momenta (quantum numbers).
+ theta, phi     ::Basic    ... polar and azimuthal angle.

struct RacahAlgebra.RacahExpression ... a type for specifying a RacahExpression with symbolic
arguments.

+ summations      ::Array{Basic,1}    ... Summation indices.
+ integrals       ::Array{Integral,1}  ... Integrals.
+ phase          ::Basic              ... Phase of the Racah expression.
+ weight         ::Basic              ... Weight of the Racah expression.
+ deltas         ::Array{Kronecker,1}  ... List of Kronecker deltas.
+ triangles      ::Array{Triangle,1}   ... List of Triangular deltas.
+ w3js           ::Array{W3j,1}       ... List of Wigner 3-j symbols.
+ w6js           ::Array{W6j,1}       ... List of Wigner 6-j symbols.
+ w9js           ::Array{W9j,1}       ... List of Wigner 9-j symbols.
+ ylms           ::Array{Ylm,1}       ... List of spherical harmonics.
+ djpbs         ::Array{Djpb,1}       ... List of (small) Wigner rotation matrices.

-----
RacahAlgebra.RacahExpression() ... constructor for defining an empty RacahExpression, i.e. of
value 1 in a mathematical context.

-----
RacahAlgebra.RacahExpression(rex::RacahAlgebra.RacahExpression;

summations=..., integrals=..., phase=..., weight=..., deltas=..., triangles=...,
w3js=..., w6js=..., w9js=..., ylms=..., djpbs=...)

... constructor for modifying a given rex::RacahExpression by ``overwriting'' the explicitly
selected parts of the expression.
```

Figure 2. Important data structures from the RacahAlgebra module of JAC that help in the symbolic evaluation of expressions from Racah's algebra. The Wigner 3-j symbol W3j (upper panel) and spherical harmonics Ylm (middle panel), here defined with symbolic arguments::Basic from the SymEngine (Julia) package [35], are two crucial entities to specify and manipulate RacahExpression's (lower panel) as a whole. For the RacahExpression, we also display two simple constructors that are employed below to set up the expressions to be evaluated. See text for further discussions.

Well-designed data structures are indeed central for the communication with and the data transfer within the program. For the symbolic evaluation of Racah expressions, all Wigner n-j symbols, spherical harmonics and rotation matrices need to be specified (and kept) separately in order to allow the desired transformation. Apart from its phase and weight, a RacahExpression therefore comprises arrays of Kronecker and triangular deltas, Wigner 3-j, 6-j and 9-j symbols, spherical harmonics as well as the (small) Wigner rotation matrices. These arrays are analyzed first separately but also together in order to find out

possible re-writings of the overall expression. Frankly speaking, these data structures form the (language) elements in order to be able to recognize and apply the various rules in the evaluation of Racah expressions.

Apart from the explicitly defined data structures, Table 1 also tabulates a few related functions, such as the `ClebschGordan` or `Dfunction`. These functions formally appear quite similar to the data types above, but just re-write the given entity in terms of the Wigner 3-j or small rotation matrix, and by taking into account the proper phase and weight factors; these functions then always return a (single-term) `RacahExpression`.

Finally, the data type `Csq` just represents a coupling sequence $(j_1, j_2) j_{12}$ of three angular momenta, though it supports also a recursive use in order to specify, for example, the coupling of four angular momenta to some total one, $((\ell_1, \ell_2) L, (s_1, s_2) S) J$, or even of many more. This data type has been introduced to readily lay out complex recoupling coefficients, as we shall discuss in Section 4.1.

Table 1. Selected data structures of the `RacahAlgebra` module that help manipulate (the individual parts of) Racah expressions. Here, only a short explanation is given, while further details can be found by using Julia's help features, for instance, `? ClebschGordan` at the REPL [39].

struct & brief explanation.

`AbstractRecursionW3j`: defines an abstract type for the recursion rules of Wigner 3-j symbols; this type comprises the concrete (singleton) types: `RecursionW3jMagnetic` to select a recursion with regard to the magnetic quantum numbers; `RecursionW3jOneStep` for a step-1 recursion of the j -quantum numbers; `RecursionW3jHalfStep` for a step-1/2 recursion of the j -quantum numbers; and `RecursionW3jLouck` for a recursion due to Louck [40].

`ClebschGordan`: re-writes a Clebsch-Gordan coefficient $\langle j_1 m_1, j_2 m_2 | j_3 m_3 \rangle$ in terms of a Wigner 3-j symbol and, thus, a Racah expression.

`ClebschGordanExpansion`: re-writes a (standard) Clebsch-Gordan expansion

$|(j_1, j_2) j m\rangle = \sum_{m_1 m_2} |j_1 m_1, j_2 m_2\rangle \langle j_1 m_1, j_2 m_2 | j m\rangle$ in terms of a Wigner 3-j symbol and, thus, a Racah expression but without the basis vectors.

`Csq`: a type for a coupling sequence $(j_1, j_2) j_{12}$ of three angular momenta with symbolic arguments that can be employed recursively.

`Delta`: a type for a Kronecker delta $\delta(j_1, j_2)$ with symbolic arguments.

`Dfunction`: re-writes a Wigner rotation matrix $D_{pq}^{(j)}(\alpha, \beta, \gamma)$ into a small rotation matrix $d_{pq}^{(j)}(\beta)$ and, thus, a Racah expression.

`Djpq`: a type for a (small) Wigner rotation matrix $d_{pq}^{(j)}(\beta)$ with symbolic arguments.

`Integral`: a type for an integral $\int_{\beta_{\text{low}}}^{\beta_{\text{up}}} d\beta \dots$, typically over some angle, with symbolic arguments and given lower and upper bound.

`RacahExpression`: a type for a Racah expression (1) with symbolic arguments; cf. Figure 1.

`Triangle`: a type for a triangular delta $\delta(j_1, j_2, j_3)$ with symbolic arguments.

`W3j`: a type for a Wigner $\begin{pmatrix} j_a & j_b & j_c \\ m_a & m_b & m_c \end{pmatrix}$ symbol with symbolic arguments; cf. Figure 1.

`W6j`: a type for a Wigner $\begin{Bmatrix} a & b & c \\ d & e & f \end{Bmatrix}$ symbol with symbolic arguments.

`W9j`: a type for a Wigner $\begin{Bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{Bmatrix}$ symbol with symbolic arguments.

`Ylm`: a type for a spherical harmonic $Y_{\ell m}(\theta, \varphi)$ with symbolic arguments; cf. Figure 1.

While the data types (and functions) of the `RacahAlgebra` module occur rather independent from the numerical part of the JAC toolbox, they complement these tools and have made it more powerful. In total, there are at present about ~ 250 of these data structures in JAC, though most of them remain hidden to the user.

3.3. Sum Rule Evaluation

The four strategies in Section 2.3 also define a hierarchy for analyzing the various parts of a given Racah expression. A simplification of such parts is attempted by going through and by comparison with a set of internally given (sum) rules. Indeed, these sum rules, which are known for the Wigner n -j symbols, form the heart of the given symbolic evaluation. More than 40 of such sum rules from the literature have been implemented [18,41] and help simplify summations over single Wigner n -j symbols or over products with a different number of such symbols. Obviously, the evaluation of Racah expressions becomes rapidly expensive as more Wigner symbols are involved in the product terms. The most complex sum rule, currently implemented, allows to replace a 9-fold summation of a product of six 3-j symbols.

To make use of these sum rules, one generally needs to cycle through all symmetric forms of the selected part of the Racah expression. A loop through these equivalent forms, but with a different sequence of arguments in each case, ensures that the angular momenta and the position of the Wigner symbols may coincide with the standard form, as known from the literature, and may thus be replaced by some simpler expression. Once this equivalence has been proven, including the proper phase, weight and independence of all other parts with regard to the relevant quantum numbers, this part of the Racah expression can be replaced by the corresponding simpler structure (i.e., by the other and simpler “side” of the rule).

The simplification of a (Racah) expression always results again into a Racah expression, though hopefully a simpler and more compact one. The detailed analysis (and replacement) of the algebraic expression behind this simplification is implemented by means of the function `RacahAlgebra.evaluate()`, and which subsequently cycles through the implemented rules. Hereby, a simplification and replacement of Wigner 3-j symbols has a higher preference than the simplification of Wigner 6-j symbols, and these again have a higher preference than the Wigner 9-j symbols. At present, all these rules are just applied ones, even if further cycles might result in yet simpler expressions. It is left to the user to determine when the evaluation process is to be terminated. Table 2 lists the major functions of the `RacahAlgebra` module that help manipulate Racah expressions and that can be called individually as well. For most applications, however, the function `evaluate(expr: RacahExpression)` will be sufficient, and its return value can often immediately be used for further derivations and calculations within the theory of angular momentum.

Table 2. Functions of the `RacahAlgebra` module which deals with expressions from Racah’s algebra.

Function & brief explanation.
<code>evaluate()</code> : attempts to evaluate or simplify a Wigner n -j symbol, recoupling coefficient or a Racah expression as a whole by means of special values, if the keyword <code>special = true</code> is given, or by sum rules.
<code>equivalentForm()</code> : generates an (randomly chosen) equivalent form for a Wigner n -j symbol or a Racah expression by using either the classical symmetries or those due to Regge [25].
<code>purifyPhase()</code> : purifies the overall phase of a Racah expression so that the contribution of any angular momentum a occurs as $-a, a, 2a, 3a$. An equivalent phase is returned.
<code>selectW3j()</code> : selects one of various pre-defined Wigner 3-j symbols for which usually special values are known; this function has been implemented mainly for test purposes and is similarly available also for the other Wigner symbols.
<code>simplifyDeltas()</code> : simplifies, if possible, all Kronecker deltas δ_{j_1, j_2} in a Racah expression if a corresponding summation is to be carried out.
<code>simplifyTriangles()</code> : simplifies all triangular deltas $\delta(j_1, j_2, j_3)$ in a Racah expression.
<code>specialValue()</code> : attempts to find a <i>special value</i> for a given Wigner n -j symbol.
<code>symmetricForms()</code> : generates a list of all <i>equivalent</i> symmetric forms of a Wigner n -j symbol, spherical harmonic $Y_{\ell m}(\vartheta, \varphi)$ or Wigner rotation matrix $d_{pq}^{(j)}(\beta)$.

3.4. Integration Rules for the Spherical Harmonics and Rotation Matrices

The full power of Racah's algebra becomes apparent only if the (re)coupling of angular momenta is combined with the use and calculus of irreducible tensor operators and statistical tensors. This combination leads to expressions, in which the Wigner n - j symbols occur together with the—summation and integration over—spherical harmonics and/or rotation matrices. In practice, of course, such a combined treatment also enhances the structure, variety and (possible) complexity of the expressions that need to be considered. It is therefore beyond the scope of the present work to deal with the full complexity of expressions, in which spherical harmonics and/or rotation matrices appear together with the Wigner n - j and other symbols from the theory of angular momentum.

In a first step, instead, we here include three orthogonality relations of the spherical harmonics that can be formulated with regard to a summation over angular momenta (quantum numbers). This step has been formalized in strategy (iii) but, otherwise, places quite similar requirements upon their implementation as the sum-rule evaluation above. Again, parts of the corresponding Racah expressions need to be analyzed (and replaced) separately owing to the given rules, and by taking all the symmetries of this part into account.

4. Derivation and Evaluation of Racah Expressions

Using JAC, the implementation of the RacahAlgebra module can be applied quite readily in daily research work or in teaching many-particle quantum mechanics. Here, we show how three problems from textbooks on atomic and angular-momentum theory can be solved without much prior knowledge about the (algebraic) techniques outlined above.

4.1. Recoupling of Angular Momenta

An isolated three-particle quantum system can often be characterized (and classified) to a good approximation by its individual angular momentum quantum numbers j_1, j_2, j_3 and the associated magnetic quantum numbers m_1, m_2, m_3 , and then described within their so-called product basis $\{|j_1 m_1, j_2 m_2, j_3 m_3\rangle\}$. Alternatively, such a system can also be classified by means of coupled angular momenta and their projections. Different coupling orders, such as $|((j_1, j_2) J_{12}, j_3) JM\rangle$ or $|(j_1, (j_2, j_3) J_{23}) JM\rangle$, might be utilized and are appropriate for different interparticle interactions [42,43]. Here, we shall not discuss the physics behind these different coupling schemes but just remind the reader that the coupled and product bases (of any N -particle system) are related to each other by a subsequent and standard Clebsch–Gordan expansion for two angular momenta at a given time:

$$|((j_1, j_2) J_{12}, j_3) JM\rangle = \sum_{m_1 m_2 m_3 M_{12}} |j_1 m_1, j_2 m_2, j_3 m_3\rangle \langle j_1 m_1, j_2 m_2 | J_{12} M_{12} \rangle \langle J_{12} M_{12}, j_3 m_3 | JM \rangle,$$

and analogue also for the $|(j_1, (j_2, j_3) J_{23}) JM\rangle$ basis states. Obviously, moreover, these expansions can be expressed in terms of the Wigner 3- j symbols, and by taking a consistent phase convention into account. In order to make a transformation from one to another coupled basis, one then often needs to evaluate the unitary transformation matrix

$$|(j_1, (j_2, j_3) J_{23}) JM\rangle = \sum_{J_{12}} |((j_1, j_2) J_{12}, j_3) JM\rangle \langle ((j_1, j_2) J_{12}, j_3) JM | (j_1, (j_2, j_3) J_{23}) JM \rangle,$$

and where the overlap integral $\langle ((j_1, j_2) J_{12}, j_3) JM | (j_1, (j_2, j_3) J_{23}) JM \rangle$ is known also as the recoupling coefficient in the literature. Indeed, the efficient evaluation of such recoupling coefficients is key in the computation of many-particle matrix elements for operators that are symmetric in the particular indices. In JAC, this recoupling coefficient can be simplified by the short (Julia) script:

```

j1 = Basic(:j1);    j2 = Basic(:j2);    j3 = Basic(:j3)
J12 = Basic(:J12);  J23 = Basic(:J23);  J  = Basic(:J)

leftCsq = RacahAlgebra.Csq(RacahAlgebra.Csq(j1, j2, J12), j3, J)
rightCsq = RacahAlgebra.Csq(j1, RacahAlgebra.Csq(j2, j3, J23), J)
wa      = RacahAlgebra.evaluate(leftCsq, rightCsq)

>> Lhs-sequence = Sum_[Basic[mj1, mj2, mj3, mJ12]] (-1)^(J12 + j1 - j2 - j3 + mJ + mJ12)
(sqrt(1 + 2*J)*sqrt(1 + 2*J12)) W3j(j1, j2, J12; mj1, mj2, -mJ12)
W3j(J12, j3, J; mJ12, mj3, -mJ)
>> Rhs-sequence = Sum_[Basic[mj1, mj2, mj3, mJ23]] (-1)^(-J23 + j1 + j2 - j3 + mJ + mJ23)
(sqrt(1 + 2*J)*sqrt(1 + 2*J23)) W3j(j2, j3, J23; mj2, mj3, -mJ23)
W3j(j1, J23, J; mj1, mJ23, -mJ)

>> Apply sum rule for three W3j -- Sum(m4,m5,m6) ...
>> Apply sum rule for two W3j -- Sum(m1,m2) ...

(-1)^(-J - j1 - j2 - j3) (sqrt(1 + 2*J12)*sqrt(1 + 2*J23)) W6j{J, j3, J12; j2, j1, J23}

```

This script makes internal use of the coupling sequence *Csq* in order to specify the correct coupling order of both sides of the recoupling coefficient, and to re-write each side in terms of its Clebsch–Gordan expansion. The internal evaluation of the recoupling coefficient above therefore starts from an 8-fold summation over products of four Wigner 3-*j* symbols and, then, subsequently applies a sum rule for three Wigner 3-*j* as well as for two 3-*j* symbols to obtain the final result. From the coupling sequence, moreover, we can reverse the phase and make use of the symmetry of the Wigner 6-*j* symbols to finally arrive at the well-known expression [44]

$$\langle ((j_1, j_2) J_{12}, j_3) JM | (j_1, (j_2, j_3) J_{23}) JM \rangle = (-1)^{j_1+j_2+j_3+J} [J_{12}, J_{23}]^{1/2} \begin{Bmatrix} j_1 & j_2 & J_{12} \\ j_3 & J & J_{23} \end{Bmatrix}.$$

Obviously, therefore, the recoupling coefficient $\langle ((j_1, j_2) J_{12}, j_3) JM | (j_1, (j_2, j_3) J_{23}) JM \rangle$ just simplifies to a Wigner 6-*j* symbol, and without any dependence on the total projection *M*, i.e., upon the choice of the quantization axis. Showing this independence on *M* for the recoupling of three (or more) angular momenta is a typical textbook example and can be obtained also from symmetry arguments. More generally, the evaluation of recoupling coefficients usually leads to products of Wigner 6-*j* symbols, often with a summation over multiple indices, and which might be further simplified by applying different sum rules for the 6-*j* symbols.

4.2. Matrix Elements of Tensor Operators within a Coupled Basis

The interaction of many-particle systems is often described by tensor operators $[T^{(k_1)} \cdot U^{(k_2)}]_Q^{(K)}$ of rank *K*, where $T^{(k_1)}$ only operates on part 1 and $U^{(k_2)}$ on part 2 of some physical system. Such a partitioning into (sub)systems might refer to the space and spin coordinates of a single particle or, more generally, to any set of independent coordinates of two particles. For these tensor operators of rank *K*, the interaction matrix elements in a coupled basis

$$\langle \alpha(j_1, j_2) JM | X_Q^{(K)} | \alpha(j'_1, j'_2) J' M' \rangle = \langle \alpha(j_1, j_2) JM | [T^{(k_1)} \cdot U^{(k_2)}]_Q^{(K)} | \alpha(j'_1, j'_2) J' M' \rangle \quad (2)$$

can then be re-written in terms of the matrix elements of the underlying subsystems. In this notation, α refers to all (further) quantum numbers that are needed for a unique classification of the two-particle states. The task is to find the correct decomposition of the matrix element (2) in terms of its subsystems by using the standard coupling of tensor

operators $[T^{(k_1)} \cdot U^{(k_2)}]_Q^{(K)} = \sum_{q_1 q_2} T_{q_1}^{(k_1)} U_{q_2}^{(k_2)} \langle k_1 q_1, k_2 q_2 | K Q \rangle$, and together with the Wigner–Eckart theorem

$$\langle jm | X_q^{(k)} | j' m' \rangle = \langle j' m', k q | jm \rangle \langle j || X^{(k)} || j' \rangle,$$

that separates the orientational dependence from the strength of interaction. If we first make a Clebsch–Gordan expansion on both sides of the matrix element (2) and use the two (simple) relations above, we can readily express the matrix element as

$$\begin{aligned} & \langle J' M', K Q | J M \rangle \langle \alpha(j_1, j_2) J || X^{(K)} || \alpha(j'_1, j'_2) J' \rangle \\ &= \sum_{\alpha''} \langle \alpha j_1 || T^{(k_1)} || \alpha'' j'_1 \rangle \langle \alpha'' j_2 || U^{(k_2)} || \alpha' j'_2 \rangle \\ & \quad \times \sum_{m_1 m_2, m'_1 m'_2, q_1 q_2} \langle j_1 m_1, j_2 m_2 | J M \rangle \langle j'_1 m'_1, j'_2 m'_2 | J' M' \rangle \langle k_1 q_1, k_2 q_2 | K Q \rangle \\ & \quad \times \langle j'_1 m'_1, k_1 q_1 | j_1 m_1 \rangle \langle j'_2 m'_2, k_2 q_2 | j_2 m_2 \rangle \end{aligned}$$

Here, again, the summation over α'' ensures the completeness of the decomposition. We can simplify the summation over all the magnetic quantum numbers by typing

```
# Left-hand side
wa = RacahAlgebra.ClebschGordan(Jp, Mp, K, Q, J, M)
(-1)^(Jp - K + M) sqrt(1 + 2*J) W3j(Jp, K, J; Mp, Q, -M)

# Right-hand side
wb = RacahExpression()
wb = RacahExpression(wb, summations=[m1, m2, m1p, m2p, q1, q2])
ClebschGordan(j1, m1, j2, m2, J, M) * ClebschGordan(j1p, m1p, j2p, m2p, Jp, Mp) *
ClebschGordan(k1, q1, k2, q2, K, Q) * ClebschGordan(j1p, m1p, k1, q1, j1, m1) *
ClebschGordan(j2p, m2p, k2, q2, j2, m2)
RacahAlgebra.evaluate(wb)

>> Apply sum rule for four W3j -- Sum(m1, m2, m3, m4) (-1)^(-m1-m2-m3-m4) ...
>> Apply sum rule for three W3j -- Sum(m4, m5, m6) ...
>> Apply sum rule for three W6j -- Sum(X) (-1)^2X [X] ...

(-1)^(J - M + j1 + j1p - j2 - j2p - k1 + k2)
(sqrt(1 + 2*K)*sqrt(1 + 2*J)*sqrt(1 + 2*j1)*sqrt(1 + 2*j2)*sqrt(1 + 2*Jp))
W3j(J, Jp, K; -M, Mp, Q) W9j{j2, J, j1; k2, K, k1; j2p, Jp, j1p}
```

and easily find, after some reordering of notations, the well-known relation

$$\langle \alpha J || X^{(K)} || \alpha' J' \rangle = \sum_{\alpha''} \langle \alpha j_1 || T^{(k_1)} || \alpha'' j'_1 \rangle \langle \alpha'' j_2 || U^{(k_2)} || \alpha' j'_2 \rangle [J, J', K]^{1/2} \begin{Bmatrix} j_1 & j'_1 & k_1 \\ j_2 & j'_2 & k_2 \\ J & J' & K \end{Bmatrix}.$$

For a scalar ($K = 0$) tensor operator $X^{(0)}$, moreover, the same procedure readily shows that the matrix elements are diagonal in J (and M)

$$\langle \alpha J || X^{(0)} || \alpha' J' \rangle = (-1)^{j_1 + j_2 + J} \sum_{\alpha''} \langle \alpha j_1 || T^{(k_1)} || \alpha'' j'_1 \rangle \langle \alpha'' j_2 || U^{(k_2)} || \alpha' j'_2 \rangle [J]^{1/2} \delta_{JJ'} \delta_{k_1 k_2} \begin{Bmatrix} j_1 & j_2 & J \\ j'_2 & j'_1 & K \end{Bmatrix},$$

and as expected for any scalar interaction.

4.3. Angular Reduction in Feynman–Goldstone Diagrams

In many-body perturbation theory or atomic QED, the angular reduction in diagrams often results in generalized recoupling coefficients with rather a large number of angular momenta as associated with the individual shells of otherwise equivalent electrons. For a

closed-shell atom, for example, (one of) the second-order correlation contributions can be written as [26,45]

$$D = \frac{1}{2} \sum_{abrs} \frac{\langle ab | \frac{1}{r_{12}} | rs \rangle \langle rs | \frac{1}{r_{12}} | ab \rangle}{\epsilon_a + \epsilon_b - \epsilon_r - \epsilon_s}, \quad (3)$$

and where $\frac{1}{r_{12}} \equiv \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|}$ refers to the instantaneous Coulomb repulsion among the electrons. Here, the matrix elements in the nominator are to be calculated in an uncoupled basis, and with one-particle energies $\epsilon_a = \epsilon_{n_a \ell_a j_a m_a}$ that do not depend on the magnetic quantum numbers. In the summation, however, the orbital indices $a \equiv (n_a \ell_a j_a m_a)$, $b \equiv (n_b \ell_b j_b m_b)$ run over all occupied one-particle states of the closed-shell atom or ion, and $r \equiv (n_r \ell_r j_r m_r), \dots$ over all those states that remain unoccupied in the Hartree–Fock determinant. By using the radial-angular decomposition of the Coulomb interaction matrix [17,46]

$$\left\langle ab \left| \frac{1}{r_{12}} \right| cd \right\rangle = \sum_{LM} (-1)^{L-M+j_a-m_a+j_b-m_b} \begin{pmatrix} j_a & L & j_c \\ -m_a & M & m_c \end{pmatrix} \begin{pmatrix} j_b & L & j_d \\ -m_b & -M & m_d \end{pmatrix} X^L(abcd)$$

we can then re-express the diagram (3) above as

$$D = \frac{1}{2} \sum_{n_a \ell_a j_a m_a, \dots, L_1 L_2} \frac{X^{L_2}(abrs) X^{L_1}(rsab)}{\epsilon_a + \epsilon_b - \epsilon_r - \epsilon_s} R$$

$$R = \sum_{m' s M_1 M_2} (-1)^S \begin{pmatrix} j_r & L_1 & j_a \\ -m_r & M_1 & m_a \end{pmatrix} \begin{pmatrix} j_s & L_1 & j_b \\ -m_s & M_1 & m_b \end{pmatrix} \begin{pmatrix} j_a & L_2 & j_r \\ -m_a & M_2 & m_r \end{pmatrix} \begin{pmatrix} j_b & L_2 & j_s \\ -m_b & M_2 & m_s \end{pmatrix},$$

and with $S = L_1 - M_1 + L_2 - M_2 + j_a - m_a + j_b - m_b + j_r - m_r + j_s - m_s$. Obviously, R has the form of a Racah expression with a six-fold summation over Wigner 3-j symbols and can be evaluated by following similar lines as in the previous example. We here omit the details of this symbolic evaluation by means of the JAC toolbox and just displays the result

$$D = \frac{1}{2} \sum_{n_a \ell_a j_a m_a, \dots, L} \frac{(-1)^{j_a+j_b-j_r-j_s}}{[L]} \frac{X^L(abrs) X^L(rsab)}{\epsilon_a + \epsilon_b - \epsilon_r - \epsilon_s} \delta(j_a, L, j_c) \delta(j_b, L, j_d).$$

In this expression, $X^L(abcd)$ refers to the so-called (reduced) Coulomb interaction strength. The same result is sometimes obtained by using graphical rules for the coupling of angular momenta. While, perhaps, such a graphical approach appears to be more elegant to the reader, it becomes tedious and prone to errors if the correct phase and weight factors need to be determined for complex diagrams. Similar tasks will occur in the future if the two-photon (QED) exchange diagrams are to be evaluated for open-shell atoms [47].

5. Conclusions

While the power and techniques of Racah's algebra have been known for more than seven decades, it often leads to expressions that are quite cumbersome to simplify. Difficulties in this simplification arise not alone from the complexity of the underlying rules, which are hard to remember in practice, but especially from the number and symmetries of the Wigner n-j symbols, and of other characteristic functions or coefficients. To overcome these difficulties, a symbolic evaluation (simplification) of Racah algebra expressions has been worked out and demonstrated in this work.

To realize such a symbolic evaluations of expression from Racah's algebra, the Jena Atomic Calculator (JAC) has been expanded by the module RacahAlgebra that helps represent and manipulate the Wigner n-j symbols, spherical harmonics, Wigner rotation matrices and several other symbols from the theory of angular momentum. A simplification of such Racah expressions is obtained by following various strategies and, especially, by using a large number of sum rules. Further rules can be easily implemented by the user.

This (algebraic) evaluation of expressions complements the numerical atomic structure and collision computations of JAC and makes this toolbox suitable for emerging applications in many-particle theory and the simulation of complex behavior.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Racah, G. Theory of complex spectra I. *Phys. Rev.* **1941**, *61*, 186. [CrossRef]
2. Racah, G. Theory of complex spectra II. *Phys. Rev.* **1942**, *62*, 438. [CrossRef]
3. Racah, G. Theory of complex spectra III. *Phys. Rev.* **1943**, *63*, 367. [CrossRef]
4. Balashov, V.V.; Grum-Grzhimailo, A.N.; Kabachnik, N.M. *Polarization and Correlation Phenomena in Atomic Collisions: A Practical Theory Course*; Kluwer Academics/Plenum Press: New York, NY, USA, 2000.
5. Wigner, E.P. *Group Theory and its Application to the Quantum Mechanics of Atomic Spectra*; Academic Press: New York, NY, USA, 1959.
6. El-Baz, E.; Castel, B. *Graphical Methods of Spin Algebras in Atomic, Nuclear and Particle Physics*; Marcel Dekker: New York, NY, USA, 1972.
7. Biedenharn, L.C.; Louck, J.D. *The Racah-Wigner Algebra in Quantum Theory*; Cambridge University Press: Cambridge, UK, 1981.
8. Adams, B.G. *Algebraic Approach to Simple Quantum Systems*; Springer: Berlin/Heidelberg Germany, 1994.
9. Fritzsche, S.; Varga, S.; Geschke, D.; Fricke, B. Maple procedures for the coupling of angular momenta. II. Sum rule evaluation. *Comput. Phys. Commun.* **1998**, *111*, 167. [CrossRef]
10. Pope, C.N.; Shen, X.; Romans, L.J. W_∞ and the Racah-Wigner algebra. *Nucl. Phys. B* **1990**, *339*, 191. [CrossRef]
11. Shimizu, Y.; Yoshida, H.; Okada, K.; Muramatsu, Y.; Saito, N. Ultra-high resolution angle-resolved measurement of Auger emission from the photoexcited $1s^{-1} \rightarrow 3p$ state of Ne. *J. Phys.* **2000**, *B33*, L685.
12. Kabachnik, N.M.; Fritzsche, S.; Grum-Grzhimailo, A.N.; Meyer, M.; Ueda, K. Coherence and correlations in photoinduced Auger and fluorescence cascades in atoms. *Phys. Rep.* **2007**, *451*, 155. [CrossRef]
13. Gaigalas, G.; Fritzsche, S.; Rudzikas, Z. Reduced coefficients of fractional parentage and matrix elements of the tensor $W^{(k_q k_j)}$ in jj -coupling. *At. Data Nucl. Data Tables* **2000**, *76*, 235. [CrossRef]
14. Zare, R.N. *Angular Momentum: Understanding Spatial Aspects in Chemistry and Physics*; John Wiley & Sons: New York, NY, USA, 1988.
15. Müller, N.; Khalil, T.; Pohl, M.; Uphues, T.; Polcik, M.; Rader, O.; Heigl, F.; Starke, K.; Fritzsche, S.; Kabachnik, N.M.; et al. Interference of spin states in resonant photoemission induced by circularly polarized light from magnetized Gd. *Phys. Rev.* **2006**, *B74*, 161401(R). [CrossRef]
16. Sitenko, A.; Tartakovskii, V. *Theory of Nucleus: Nuclear Structure and Nuclear Interaction*; Springer Science+Business Media, B.V.: Dordrecht, UK, 1997.
17. Grant, I.P. *Relativistic Quantum Theory of Atoms and Molecules: Theory and Computation*; Springer: New York, NY, USA, 2007.
18. Varshalovich, D.A.; Moskalev, A.N.; Khersonskii, V.K. *Quantum Theory of Angular Momentum*; World Scientific: Singapore, 1988.
19. Gaigalas, G.; Fritzsche, S.; Grant, I.P. Calculation of angular coefficients in jj -coupling. *Comput. Phys. Commun.* **2001**, *139*, 263. [CrossRef]
20. Fritzsche, S.; Froese Fischer, C.; Gaigalas, G. A program for relativistic configuration interaction calculations. *Comput. Phys. Commun.* **2002**, *148*, 103. [CrossRef]
21. Fritzsche, S. Application of point-group symmetries in chemistry and physics: A computer-algebraic approach. *Int. J. Quant. Chem.* **2006**, *106*, 98. [CrossRef]
22. Soguel, R.N.; Volotka, A.V.; Tryapitsyna, E.V.; Glazov, D.A.; Kosheleva, V.P.; Fritzsche, S. Redefined vacuum approach and gauge-invariant subsets in two-photon-exchange diagrams for closed-shells system with a valence electron. *Phys. Rev.* **2020**, *A103*, 042818.
23. Raychev, P. Quantum Groups: Application to Nuclear and Molecular Spectroscopy. *Adv. Quant. Chem.* **1995**, *26*, 239.
24. Fritzsche, S.; Grum-Grzhimailo, A.N.; Gryzlova, E.V.; Kabachnik, N.M. Sequential two-photon double ionization of Kr atoms. *J. Phys.* **2009**, *B42*, 145602. [CrossRef]
25. Regge, T. Symmetry properties of Clebsch-Gordan coefficients. *Nuovo C.* **1958**, *10*, 544. [CrossRef]
26. Lindgren, I.; Morrison, J. *Atomic Many-Body Theory*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 1986.
27. Fritzsche, S. A fresh computational approach to atomic structures, processes and cascades. *Comp. Phys. Commun.* **2019**, *240*, 1. [CrossRef]
28. Fritzsche, S. JAC: User Guide, Compendium & Theoretical Background. Available online: <https://github.com/OpenJAC/JAC.jl> (accessed on 1 June 2021).
29. Fritzsche, S. Large-scale accurate structure calculations for open-shell atoms and ions. *Phys. Scr.* **2002**, *T100*, 37. [CrossRef]

30. Cheal, B.; Cocolios, T.E.; Fritzsche, S. Laser spectroscopy of radioactive isotopes: Role and limitations of accurate isotope-shift calculations. *Phys. Rev.* **2012**, *A86*, 042501. [CrossRef]
31. Fritzsche, S.; Surzhykov, A. Approximate atomic Green functions. *Molecules* **2021**, *26*, 2660. [CrossRef]
32. Fritzsche, S.; Palmeri, P.; Schippers, S. Atomic cascade computations. *Symmetry* **2021**, *13*, 520. [CrossRef]
33. Beerwerth, R.; Buhr, T.; Perry-Sassmannshausen, A.; Stock, S.O.; Bari, S.; Holste, K.; Kilcoyne, A.L.D.; Reinwardt, S.; Ricz, S.; Savin, D.W. Near L-edge single and multiple photoionization of triply charged iron ions. *Astrophys. J.* **2019**, *887*, 189. [CrossRef]
34. Schippers, S.; Beerwerth, R.; Bari, S.; Buhr, T.; Holste, K.; Kilcoyne, A.L.D.; Perry-Sassmannshausen, A.; Phaneuf, R.A.; Reinwardt, S.; Savin, D.W. Near L-edge single and multiple photoionization of doubly charged iron ions. *Astrophys. J.* **2021**, *908*, 52. [CrossRef]
35. SymEngine.jl. Available online: <https://github.com/symengine/SymEngine.jl> (accessed on 1 June 2021).
36. Fritzsche, S. Maple procedures for the coupling of angular momenta. I. Data structures and numerical evaluation. *Comp. Phys. Commun.* **1997**, *103*, 51. [CrossRef]
37. Fritzsche, S.; Inghoff, T.; Bastug, T.; Tomaselli, M. Maple procedures for the coupling of angular momenta. V. Recoupling coefficients. *Comput. Phys. Commun.* **2001**, *139*, 314. [CrossRef]
38. Pagarán, J.; Fritzsche, S.; Gaigalas, G. Maple procedures for the coupling of angular momenta. IX. Wigner D-functions and rotation matrices. *Comput. Phys. Commun.* **2006**, *174*, 616. [CrossRef]
39. JULIA Comes with a Full-Featured Interactive and Command-Line REPL (Read-Eval-Print Loop) That Is Built into the Executable of the Language. Available online: <https://docs.julialang.org/en/v1/stdlib/REPL/> (accessed on 1 June 2021).
40. Louck, J.D. New recursion relation for the Clebsch-Gordan coefficients. *Phys. Rev.* **1958**, *110*, 815. [CrossRef]
41. Rotenberg, M.; Bivins, R.; Metropolis, N.; Wooten, J.K., Jr. *The 3-j and 6-j Symbols*; The Technology Press Cambridge: Cambridge, MA, USA, 1959.
42. Judd, B.R. *Operator Techniques in Atomic Spectroscopy*; McGraw-Hill: New York, NY, USA, 1963.
43. Gaigalas, G.; Zalandauskas, T.; Fritzsche, S. Spectroscopic *LSJ* notation for atomic levels as obtained from relativistic calculations. *Comput. Phys. Commun.* **2004**, *157*, 239. [CrossRef]
44. Rose, M.E. *Elementary Theory of Angular Momentum*; Dover Publications Inc.: New York, NY, USA, 1995.
45. Fritzsche, S.; Mani, B.K.; Angom, D. A computer—Algebraic approach to the derivation of Feynman–Goldstone perturbation expansions for open-shell atoms and molecules. *Adv. Quant. Chem.* **2008**, *53*, 177.
46. Fritzsche, S. The RATIP program for relativistic calculations of atomic transition, ionization and recombination properties. *Comp. Phys. Commun.* **2012**, *183*, 1525. [CrossRef]
47. Soguel, R.N.; Volotka, A.V.; Glazov, D.A.; Fritzsche, S. Many-electron QED with redefined vacuum approach. *Symmetry* **2021**, *13*, 1014. [CrossRef]