Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Physics Communications 153 (2003) 424–444
Computer Physics Communications
www.elsevier.com/locate/cpc

Maple procedures for the coupling of angular momenta. VII. Extended and accelerated computations ☆

S. Fritzsche [a,*], T. Inghoff [a], M. Tomaselli [b]

[a] *Fachbereich Physik, Universität Kassel, Heinrich-Plett-Str. 40, D-34132 Kassel, Germany*
[b] *Institut für Kernphysik, Universität Darmstadt, D-64289 Darmstadt, Germany*

Received 13 August 2002; accepted 13 September 2002

Abstract

During recent years, much attention in developing general-purpose, computer–algebra systems was focused not only on better symbolic algorithms but, to a very similar extent, also on fast numerical computations and improved tools for visualization. Behind this development, of course, the main idea is to provide the users with a single environment for the solution of their scientific or engineering tasks. In a revised version of the RACAH program, we follow this idea and provide a fast and much extended access to the standard quantities from the theory of angular momentum within the framework of MAPLE. In this revision, emphasis is paid to the efficient computation of the standard quantities by supporting both, the default software model as well as fast (hardware) floating-point computations. Moreover, RACAH is now organized and distributed as a MAPLE *module* which can be installed and utilized like any other module, including help pages and the use of internally recognized data structures. The present extension of the RACAH program may therefore enlarge the range of applications considerably towards problems from quantum optics, collision theory or even solid-state physics.

PACS: 3.65.F; 2.90.+p

NEW VERSION SUMMARY

Title of program: RACAH

Catalogue identifier: ADRW

Program Summary URL: http://cpc.cs.qub.ac.uk/summaries/ADRW

Program obtainable from: CPC Program Library, Queen's University of Belfast, N. Ireland

Licensing provisions: none

Computers for which the program is designed: all computers with a license of the computer algebra package MAPLE [1]

Installations: University of Kassel (Germany)

Operating systems under which the program has been tested: Linux 7.1+ and Windows2000

Program language used: MAPLE 7 and 8

Memory required to execute with typical data: 5–50 MB

☆ This program can be downloaded from the CPC Program Library under catalogue identifier: http://cpc.cs.qub.ac.uk/summaries/ADRW
* Corresponding author.
E-mail address: s.fritzsche@physik.uni-kassel.de (S. Fritzsche).

0010-4655/03/$ – see front matter
doi:10.1016/S0010-4655(03)00227-3

No. of bytes in distributed program, including test data, etc.: 2 429 654

Distribution format: tar gzip file

Keywords: Angular momentum, bipolar harmonic, Clebsch–Gordan coefficient, coefficient of fractional parentage, Condon–Shortley coefficient, Cruzan's algorithm, Gaunt coefficient, Racah's W-coefficient, reduced matrix element, rotation matrix, Sharp's symbol, spherical harmonic, spinor spherical harmonic, tensor spherical harmonic, tripolar harmonic, unit tensor, vector spherical harmonic, Wigner D-function, Wigner $n{-}j$ symbol.

Nature of the physical problem
The theories of angular momentum and spherical tensor operators provide a very powerful calculus for the study of (quantum) many-particle systems, sometimes known also as Racah's algebra. The efficient use of these theories, however, require—apart from the knowledge of a great deal of algebraic transformations and rules—a fast and reliable access to their standard quantities such as the Wigner $n{-}j$ symbols and Clebsch–Gordan coefficients, spherical harmonics of various kinds, the rotation matrices, and many others.

Method of solution
A set of MAPLE procedures has been developed and maintained over the last years which supports both, algebraic manipulations as well as fast computations of the standard expressions from the theory of angular momentum.

Restrictions onto the complexity of the problem
Of course, the full power of the angular momentum theory is given by a large set of (group-theoretical and often rather sophisticated) relations between its standard quantities, which may help simplify and reduce the "numerical costs" in the theoretical treatment of (quantum) many-particle systems dramatically. In fact, many details of such systems can be understood only if the proper algebraic relations are found and applied. In the present version of RACAH, however, symbolic manipulations are supported so far only for those expressions which include the Wigner $n{-}j$ symbols ($n \leqslant 9$), Clebsch–Gordan coefficients and/or spherical harmonics. For all other quantities, we currently just facilitate fast numerical computations, by making use also of MAPLE's recently implemented *hardware floating-point model*.

Unusual features of the program
The RACAH program provides an interactive environment which, apart from the standard symbols and functions of the theory of angular momentum, also supports the evaluation of recoupling coefficients, various coefficients and matrix elements from the atomic shell model as well as transformation matrices between different coupling schemes [2].

Typical running time
Although the program replies 'promptly' on most requests, the running time depends strongly on the complexity of the expressions.

References
[1] Maple is a registered trademark of Waterloo Maple Inc.
[2] S. Fritzsche, Comp. Phys. Commun. 103 (1997) 51;
G. Gaigalas, S. Fritzsche, B. Fricke, Comp. Phys. Commun. 135 (2001) 219;
S. Fritzsche, T. Inghoff, T. Bastug, M. Tomaselli, Comp. Phys. Commun. 139 (2001) 314.

LONG WRITE-UP

1. Introduction

The theory of angular momentum offers two crucial advantages for the treatment of quantum many-particle systems: (i) the definition of rather a small number of standard quantities and (ii) an elegant and very powerful calculus which help simplify and evaluate sophisticated expressions. Owing to these advantages, the techniques from this theory (sometimes known also as Racah algebra techniques [1]) have been utilized in a large number of applications and in quite different field of many-particle physics [2,3]. In the earlier design [4–6] of the RACAH program, however, we just focused on the second benefit so far, the algebraic transformation of *Racah expressions* [cf. Fig. 1 in Ref. [4]] which were found appropriate for symbolic manipulations. Until now, such Racah expressions may include the Wigner $n{-}j$ symbols, Clebsch–Gordan and recoupling coefficients as well as (various integrals over) the spherical harmonics. To obtain a simplification even for complex expressions, a large variety of sum and orthogonality rules were implemented earlier. Today, these developments from the last eight years about are utilized not only in the automatic derivation of (atomic) perturbation expansions, but also in the $LS \leftrightarrow jj$ transformation of symmetry-adapted functions [7] as well as for the evaluation of many-particle matrix elements within the atomic and nuclear shell model [8].

Less attention in the earlier design of the RACAH program has been paid to the efficient and reliable computation of the standard quantities from the angular momentum theory. Fast computations, of course, require the use of *hardware floating-point algorithms*, a line which has now been followed up also by several general-purpose, computer–algebra (CA) systems such as MAPLE or MATHEMATICA. The implementation of floating-point algorithms may help solve scientific and engineering tasks within the 'same computational environment', starting from a first analysis of the problem and by utilizing both, symbolic manipulations *and* numerical computations until the point is reached where the results need to be visualized and presented. For applications of the theory of angular momentum, the realization of this concept requires an efficient computation not only for the Wigner n–j symbols and spherical harmonics but also for a much larger set of other symbols and functions. A major interest in such a development concerns, for instance, various tensorial types of the spherical harmonics, the Wigner rotation matrices or special coefficients from the (atomic or nuclear) shell model. But although these standard quantities can be often introduced quite easily into some theoretical derivation, a reliable implementation—which is consistent in all of its definitions and phases—is usually much harder to find. Typical fields of application hereby may include high-energy physics, atomic and molecular structure and scattering theory or even electro-magnetic field computations for nano-type structures and mesoscopic systems.

With a larger range of applications, of course, a better incorporation of the RACAH program into the (underlying) MAPLE environment also became necessary and very desirable. This need is obvious as the whole program now contains more than 240 subprocedures which are hidden mainly behind the about 20 user-relevant (and public) commands. For a proper encapsulation and protection of the code, MAPLE now supports the use of *modules* which help incorporate additional packages into its overall framework, including new help pages, the use of internal data structures, and a much simpler distribution and installation of the code. In addition, the careful use of modules facilitates further improvements and modifications of the code without that the user interface need to be changed.

To follow the recent trend in the development of the general-purpose CA environments, here we present a revised version of the RACAH package. This package now provides a considerably enlarged set of (standard) functions from the theory of angular momentum which are distributed within two (MAPLE) modules, including their help pages and a manual for quick reference. In the next section, we first summarize all quantities which are now supported by the program as well as the use and implementation of floating-point algorithms for a few more important symbols. For their detailed definition and computation, however, we refer the reader to two appendices below. For most of the additional quantities, emphasis was paid first of all on a reliable computation while the knowledge about their (algebraic) properties and transformations can be likely considered only stepwise in the future. Section 3, then, describes the revised program structure and the distribution of the code. A short run time comparison in Section 4 finally demonstrates the acceleration in the computation of the Wigner 3–j and 6–j symbols and the spherical harmonics.

2. Extensions to the RACAH program

2.1. Enlarged set of numerical procedures

In physics, a number of *standard quantities* are typically used in order to express most formulas which are related to the theory of angular momentum. Apart from the Wigner n–j symbols (whose need, originally, gave rise to the design and set-up of the RACAH package), for instance, these are the rotation matrices of various types, products and linear combinations of the spherical harmonics, reduced matrix elements, and many others. The algebraic manipulation of these quantities often help achieve (mathematical) simplifications of great elegance and, thus, to obtain insight also into the behavior of physical systems. For most of these quantities, however, the properties and relations among each other are unfortunately not (yet) available at a (*computer*–) *algebraic level*; a situation which is likely not to change much within the next few years. Therefore, in order to facilitate at least the efficient use and computation of these quantities, we incorporated several of them at a 'numerical level' into the present version of the RACAH program. Table 1 lists the presently implemented symbols and functions from the theory of angular momentum and the (corresponding) commands. These procedures provide not only a fast and interactive access to these entities—and, hence, may 'replace' many (old-fashioned) tabulations—but will also facilitate new applications, for instance, in the treatment of many-particle systems. The full algebraic support of

Table 1
List of symbols and functions which are presently supported by the RACAH program. The definition of these quantities mainly follows the monograph by Varshalovich et al. [3] about the theory of angular momentum

Symbol	Designation	RACAH procedure
$\begin{pmatrix} a & b & c \\ m_a & m_b & m_c \end{pmatrix}$	Wigner 3–j symbol	Racah_w3j(), Racah_w3j_range()
$\begin{Bmatrix} a & b & c \\ d & e & f \end{Bmatrix}$	Wigner 6–j symbol	Racah_w6j(), Racah_w6j_range()
$\begin{Bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{Bmatrix}$	Wigner 9–j symbol	Racah_w9j()
$\left\{\begin{array}{cccc\|c} a & b & c & d & \\ e & f & g & h & s \\ i & j & k & l & \end{array}\right\}$	Wigner 12–j symbol of kind $s = 1, 2$	Racah_w12j()
$\begin{Bmatrix} - & a_2 & a_3 & a_4 \\ b_1 & - & b_3 & b_4 \\ c_1 & c_2 & - & c_4 \\ d_1 & d_2 & d_3 & - \end{Bmatrix}$	Sharp's symbol [9]	Racah_w12j()
$\langle a\, m_a,\ b\, m_b \| c\, m_c \rangle$	Clebsch–Gordan coefficient	Racah_ClebschGordan()
$W(abcd; ef)$	Racah's W coefficient	Racah_Wcoefficient()
$d^j_{mm'}(\beta)$	Wigner $d^j_{mm'}(\beta)$ rotation matrix	Racah_dmatrix()
$D^j_{mm'}(\alpha, \beta, \gamma)$	Wigner's D-function	Racah_Dmatrix()
$U^j_{mm'}(\omega; \Theta, \Phi)$	Rotation matrix $U(\omega)$	Racah_Umatrix()
$Y_{lm}(\vartheta, \varphi)$	Spherical harmonic	Racah_Ylm()
$\{\mathbf{Y}_{l_1}(\vartheta_1, \varphi_1) \otimes \mathbf{Y}_{l_2}(\vartheta_2, \varphi_2)\}_{LM}$	Bipolar spherical harmonic	Racah_bipolarY()
$\{\mathbf{Y}_{l_1}(\vartheta_1, \varphi_1) \otimes \{\mathbf{Y}_{l_2}(\vartheta_2, \varphi_2) \otimes \mathbf{Y}_{l_3}(\vartheta_3, \varphi_3)\}_{l_{23}}\}_{LM}$	Tripolar spherical harmonic	Racah_tripolarY()
$Y^{ls}_{jm}(\vartheta, \varphi)$	Tensor spherical harmonic	Racah_tensorY()
$\Omega^l_{jm}(\vartheta, \varphi)$	Spinor spherical harmonic	Racah_spinorY()
$\mathbf{Y}^l_{jm}(\vartheta, \varphi)$	Vector spherical harmonic	Racah_vectorY()
$\langle l_a m_a \| l_b m_b \| l_c m_c \rangle$	Gaunt coefficient	Racah_Gaunt()
$C^k(l_a, m_a; l_b, m_b)$	Condon–Shortley coefficient	Racah_CondonShortley()

Table 1
(Continued)

Symbol	Designation	RACAH procedure
$\left(\gamma\alpha Q\Gamma \,\|\|\|\, a^{(q\gamma)} \,\|\|\|\, \gamma\alpha' Q'\Gamma'\right)$	Reduced coefficient of fractional parentage	Racah_rcfp()
$\left(l^N \alpha QLS \,\|\|\, l^{N-1}(\alpha' Q' L' S') l\right)$ or		
$\left(j^N \alpha QJ \,\|\|\, j^{N-1}(\alpha' Q' J') j\right)$	Coefficient of fractional parentage	Racah_cfp()
$\left(j^N \alpha J \,\|\|\, T^{(k)} \,\|\|\, j^N \alpha' J'\right)$	Reduced matrix element of the unit tensors $T^{(k)}$ in jj-coupling	Racah_reduced_T()
$\left(l^N \alpha LS \,\|\|\, U^{(k)} \,\|\|\, l^N \alpha' L' S'\right)$	Reduced matrix element of the unit tensors $U^{(k)}$ in LS-coupling	Racah_reduced_U()
$\left(l^N \alpha LS \,\|\|\, V^{(k1)} \,\|\|\, l^N \alpha' L' S'\right)$	Reduced matrix element of the unit tensors $V^{(k1)}$ in LS-coupling	Racah_reduced_V()
$\left(j\ \alpha QJ \,\|\|\|\, W^{(k_q k_j)} \,\|\|\|\, j\ \alpha' Q' J'\right)$	Reduced matrix element of the unit tensors $W^{(k_q k_j)}$ in jj-coupling	Racah_reduced_W()
$\left(l\alpha QLS \,\|\|\|\, W^{(k_q k_l k_s)} \,\|\|\|\, l\alpha' Q' L' S'\right)$	Reduced matrix element of the unit tensors $W^{(k_q k_l k_s)}$ in LS-coupling	Racah_reduced_W()

these quantities, however, can be provided only later step by step as further symbolic algorithms are developed and, in fact, the requirements will arise from the side of the user. Since the calculus of angular momentum has been developed to a very large and powerful framework during the last six decades, collaborations and help from other groups are always appreciated for this project.

Most of the quantities from Table 1 are likely to be known to the reader from the literature. In this write-up, therefore, we may leave out details from their definition and the large number of mathematical relations which they fulfill. For such details, we refer the reader to the classical text by Varshalovich et al. [3]. Appendix A, however, compiles some of the background in order to make the present implementation of the RACAH program useful for practical applications. An alphabetic list and description of all commands of the package, including those from Table 1, is provided with the program [cf. Section 3]. In the following, we only recall some basic facts and conventions, about which the user must be aware of, and where these quantities (may) arise in applications.

Wigner $3n\!-\!j$ symbols ($n = 1, 2$, and 3). The Wigner $3n\!-\!j$ symbols are all related to the transformation of angular momenta between different coupling schemes. For $n = 1, 2$, and 3, these symbols were taken as the basic data structures of the RACAH program and were defined explicitly in Ref. [4], Appendix A. The Wigner $3n\!-\!j$ symbols frequently arise in (almost) all applications of the theory of angular momentum.

$12\!-\!j$ symbols of first and second type. Sharp's symbol. Wigner $3n\!-\!j$ symbols of higher order ($n \geqslant 4$) are rarely used in applications as their complexity increases rapidly with n and as several *kinds* of these symbols appear [3]. There are two kinds of $12\!-\!j$ symbols, called the *first* and *second* kind or $12\!-\!j(1)$ and $12\!-\!j(2)$ symbols, respectively. These symbols are often written as

$$\left\{ \begin{array}{cccc|c} a_1 & a_2 & a_3 & a_4 & \\ b_1 & b_2 & b_3 & b_4 & s \\ c_1 & c_2 & c_3 & c_4 & \end{array} \right\},$$

where $s = 1, 2$ selects the *kind*. Instead of the $12\!-\!j$ symbol of *second* kind, *Sharp's symbol* [9] is sometimes used which possess a slightly higher symmetry [cf. Appendix A.2]. For $n = 4$ and 5, the properties of the $3n\!-\!j$ symbols are still a topic in modern research on group theory.

Clebsch–Gordan and Racah's W coefficients. The Clebsch–Gordan or vector coupling coefficients appear naturally as Fourier coefficients in the *re-coupling* of angular momenta; they are closely related to the Wigner 3–j symbols where, in the RACAH program, we use the Condon–Shortley pase convention [10]

$$\langle j_1 m_1, j_2 m_2 \mid j_3 m_3 \rangle = (-1)^{j_1 - j_2 + m_3} [j_3]^{1/2} \begin{pmatrix} j_1 & j_2 & j_3 \\ m_1 & m_2 & -m_3 \end{pmatrix}. \tag{1}$$

Also, Racah's W coefficients, $W(abed;cf) = (-1)^{a+b+d+e} \left\{ \begin{smallmatrix} a & b & c \\ d & e & f \end{smallmatrix} \right\}$, are basically equivalent to the Wigner 6–j symbols and were used only in the earlier literature on the recoupling of angular momenta; in the RACAH program, these coefficients are supported just for a quick reference or in order to *check* some given relation from the literature.

Wigner's D-function $D^j_{mm'}(\alpha, \beta, \gamma)$. The D-functions are often required for the transformation of wave functions under the rotation of coordinates

$$\left|\Psi_{jm'}(\vartheta', \varphi', \sigma')\right\rangle = \sum_m \left|\Psi_{jm}(\vartheta, \varphi, \sigma)\right\rangle D^j_{mm'}(\alpha, \beta, \gamma), \tag{2}$$

where ϑ, φ and ϑ', φ' are the polar angles in the initial and the rotated system, and $\sigma\,(\sigma')$ denote the corresponding spin variables. More general, these functions occur in the transformation of any irreducible spherical tensor of rank j. The D-functions are utilized, therefore, in all treatments of scattering processes (from elementary particles up to molecules and clusters), including capture, transfer, and emission processes. The D-functions fulfill a large number of (symmetry) properties [3] and are expressed most simply in terms of the Euler angles ($\alpha, \beta,$ and γ) and the Wigner rotation matrix $d^j_{mm'}(\beta)$, see below.

Rotation matrix $d^j_{mm'}(\beta)$. This matrix describes the rotation of any spherical tensor by the angle β around a given axis, say, the quantization axis of the system. Several explicit representations in $\sin(\beta/2)$ or $\cos(\beta/2)$, the hypergeometric function, or in terms of other polynomials exist for this *real* function; it will be used also to extent the data structures of the RACAH program for algebraic manipulations in the future.

Rotation matrix $U^j_{mm'}(\omega; \Theta, \Phi)$. Instead of the Euler angles (α, β, γ), it is sometimes more convenient to describe rotations in terms of the rotation angle ω (around the axis of rotation) and the two polar angles Θ, Φ which determine the orientation of the rotation axis. For this choice of angles, the rotation operator is $\mathrm{e}^{-\mathrm{i}\omega\mathbf{n}\cdot\mathbf{j}}$ with $\mathbf{n} \equiv (\Theta, \Phi)$ and the matrix elements of this operator in terms of the variables ω, Θ, Φ give rise directly to

$$U^j_{mm'}(\omega; \Theta, \Phi) \equiv \left\langle jm \middle| \mathrm{e}^{-\mathrm{i}\omega\mathbf{n}\cdot\mathbf{j}} \middle| jm' \right\rangle. \tag{3}$$

An explicit representation of the rotation matrix $U^j_{mm'}(\omega; \Theta, \Phi)$ can be obtained from the D-functions [3] by using a proper transformation of the variables $(\omega; \Theta, \Phi) \to (\alpha, \beta, \gamma)$

$$U^j_{mm'}(\omega; \Theta, \Phi) = \mathrm{i}^{m-m'} \mathrm{e}^{-\mathrm{i}(m-m')\Phi} \left(\frac{1 - \mathrm{i} \tan\frac{\omega}{2} \cos\Theta}{\sqrt{1 + \tan^2\frac{\omega}{2} \cos^2\Theta}} \right)^{m+m'} d^j_{mm'}(\xi) \tag{4}$$

where the angle ξ is determined by

$$\sin\frac{\xi}{2} = \sin\frac{\omega}{2} \sin\Theta.$$

Spherical harmonics. The $Y_{lm}(\theta, \phi)$ functions form a complete and orthonormal set on the unit sphere, and are therefore widely used in classical and quantum physics. Not much need to be said here as their definition and implementation into the RACAH program have been presented recently [11]; the spherical harmonics frequently appear in the representation of wave functions for a wide range of physical systems, in the evaluation of the corresponding (quantum) matrix elements, and at many places elsewhere.

Bipolar and tripolar spherical harmonics. When defined as *irreducible tensors*, linear combinations of products of p spherical harmonics often form a convenient basis to represent (distribution) functions in physics which depend on p vector directions. For this purpose, for instance, the *bipolar spherical harmonics* are defined as the irreducible tensor product of two spherical harmonics with different arguments [3]

$$\left\{\mathbf{Y}_{l_1}(\vartheta_1,\varphi_1)\otimes\mathbf{Y}_{l_2}(\vartheta_2,\varphi_2)\right\}_{LM}=\sum_{m_1m_2}\langle l_1m_1,l_2m_2\mid LM\rangle Y_{l_1m_1}(\vartheta_1,\varphi_1)Y_{l_2m_2}(\vartheta_2,\varphi_2). \tag{5}$$

For different $l_1, l_2, L,$ and M, the bipolar harmonics form a complete and orthonormal set

$$\iint \mathrm{d}\Omega_1\,\mathrm{d}\Omega_2\left\{\mathbf{Y}_{l_1}(\Omega_1)\otimes\mathbf{Y}_{l_2}(\Omega_2)\right\}^*_{LM}\left\{\mathbf{Y}_{l'_1}(\Omega_1)\otimes\mathbf{Y}_{l'_2}(\Omega_2)\right\}_{L'M'}=\delta_{l_1l'_1}\delta_{l_2l'_2}\delta_{LL'}\delta_{MM'} \tag{6}$$

of functions which depend on two unit vectors, say, $\mathbf{n}_1$ and $\mathbf{n}_2$, respectively. A similar definition also applies for the *tripolar spherical harmonics*

$$\begin{aligned}&\left\{\mathbf{Y}_{l_1}(\vartheta_1,\varphi_1)\otimes\left\{\mathbf{Y}_{l_2}(\vartheta_2,\varphi_2)\otimes\mathbf{Y}_{l_3}(\vartheta_3,\varphi_3)\right\}_{l_{23}}\right\}_{LM}\\&\quad=\sum_{m_1,m_2,m_3,m_{23}}\langle l_1m_1,l_{23}m_{23}\mid LM\rangle\langle l_2m_2,l_3m_3\mid l_{23}m_{23}\rangle Y_{l_1m_1}(\vartheta_1,\varphi_1)Y_{l_2m_2}(\vartheta_2,\varphi_2)Y_{l_3m_3}(\vartheta_3,\varphi_3),\end{aligned} \tag{7}$$

where, however, different coupling sequences are possible and have to be taken into account in applications. Apart from the sequence $\mathbf{L}=\mathbf{l}_1+(\mathbf{l}_2+\mathbf{l}_3)_{l_{23}}$, we also support the coupling $\mathbf{L}=(\mathbf{l}_1+\mathbf{l}_2)_{l_{12}}+\mathbf{l}_3$ in the RACAH program; these two coupling sequences are related to each other by standard recoupling theory. The bipolar and tripolar harmonics obey simple transformation properties under the rotation and inversion of the coordinates. For products of two or more of these functions, which are given with the same arguments $\Omega_1,\Omega_2,\ldots,$ there exist Clebsch–Gordan expansions as for the spherical harmonics, including the Wigner n–j symbols of various types [cf. Appendix A.4].

Tensor spherical harmonics. Following the standard coupling of two angular momenta, the tensor spherical harmonics are constructed as a sum of products of the spherical harmonics $Y_{lm_l}(\vartheta,\varphi)$ (eigenfunctions of $\mathbf{l}^2$ and l_z) and the spin functions χ_{sm_s} (eigenfunctions of $\mathbf{s}^2$ and s_z)

$$Y^{ls}_{jm}(\vartheta,\varphi)\equiv\{\mathbf{Y}_l\otimes\chi_s\}_{jm}=\sum_{m_l,m_s}Y_{lm_l}(\vartheta,\varphi)\chi_{sm_s}\langle lm_l,sm_s\mid jm\rangle \tag{8}$$

so that an irreducible tensor of rank j is obtained. While the l quantum number always occurs as a (nonnegative) integer, the indices j and s are both either integers or half-integers. For given j and s, the (orbital) angular momentum l can take the values $j+s, j+s-1,\ldots,|j-s|$; the allowed values of m are $j, j-1,\ldots,-j$. Typically, the tensor spherical harmonics $Y^{ls}_{jm}(\vartheta,\varphi)$ are represented by a column matrix with $(2s+1)$ elements so that the summation over the spin variable is replaced by a matrix multiplication. Similar to the spherical harmonics, which form a complete set of functions on the unit sphere, the tensor spherical harmonics $Y^{ls}_{jm}(\vartheta,\varphi)$ form a complete and orthonormal set of functions for the expansion of rank s spinors with the domain of arguments $0\leqslant\vartheta\leqslant\pi$, $0\leqslant\varphi\leqslant 2\pi$.

According to different definitions of the spin functions such as in a Cartesian, spherical, or helicity basis representation, different components of the tensor spherical harmonics need to be distinguished. In the RACAH program, we currently support

(i) the *contravariant spherical components* from Eq. (8)

$$\left[Y^{ls}_{jm}(\vartheta,\varphi)\right]^{\sigma} = Y_{l,m-\sigma}(\vartheta,\varphi)\langle lm-\sigma, s\sigma \mid jm\rangle, \quad \sigma = s, s-1, \ldots, -s, \tag{9}$$

(ii) the *contravariant helicity components* [3]

$$\left[Y^{ls}_{jm}(\vartheta,\varphi)\right]^{\prime\lambda} = \left[\frac{2l+1}{4\pi}\right]^{1/2} D^{j}_{-\lambda-m}(0,\vartheta,\varphi)\langle l0, s\lambda \mid j\lambda\rangle. \tag{10}$$

There are two special cases of the tensor spherical harmonics which are often discussed separately.

Spinor spherical harmonics. For $s = 1/2$, the tensor harmonics (8) are also called spinor spherical harmonics

$$\Omega^{l}_{jm}(\vartheta,\varphi) \equiv Y^{l\frac{1}{2}}_{jm}(\vartheta,\varphi) \tag{11}$$

which are eigenfunctions of the operators $\mathbf{j}^2$, j_z, $\mathbf{l}^2$ and $\mathbf{s}^2$, where $\mathbf{s}$ is assumed to be the spin operator for $s = 1/2$. As for the tensor spherical harmonics, a number of different components need to be distinguished, including contravariant and covariant tensor components. Table 2 lists the components of the spinor and vector spherical harmonics which are presently supported by the RACAH program; they refer to different definitions of the spin basis functions as compiled in Appendix A.5.

Vector spherical harmonics. The other special case of the tensor spherical harmonics are those for spin $s = 1$, i.e. the vector spherical harmonics

$$\mathbf{Y}^{l}_{jm}(\vartheta,\varphi) \equiv Y^{l1}_{jm}(\vartheta,\varphi). \tag{12}$$

A large deal of representations, integrals, and algebraic relations are known for the vector spherical harmonics which play an crucial role, for instance, in the quantum theory of light and in the current (hot) topic of laser-matter interactions. Again, however, only a few components of the vector spherical harmonics with respect to different definitions of the spin basis functions are presently supported by the program (see Table 2).

Table 2
Contravariant and covariant components of the spinor and vector spherical harmonics. See Appendix A for the definition of the different spin basis functions

Spin basis	Component	Expression	Remarks
		Spinor spherical harmonics	
Spherical spin basis (A.17)	contravariant	$\left[\Omega^{l}_{jm}(\vartheta,\varphi)\right]^{\sigma} = \langle lm-\sigma, \frac{1}{2}\sigma \mid jm\rangle Y_{l,m-\sigma}(\vartheta,\varphi)$	$l = j \pm 1/2;\ \sigma = \pm 1/2$
	covariant	$\left[\Omega^{l}_{jm}(\vartheta,\varphi)\right]_{\sigma} = (-1)^{1/2-\sigma}\left[\Omega^{l}_{jm}(\vartheta,\varphi)\right]^{-\sigma}$	$l = j \pm 1/2;\ \sigma = \pm 1/2$
Spherical helicity basis (A.21)	contravariant	$\left[\Omega^{l}_{jm}(\vartheta,\varphi)\right]^{\prime\lambda} = \sqrt{\frac{2l+1}{4\pi}}\langle l0, \frac{1}{2}\lambda \mid j\lambda\rangle D^{j}_{-\lambda,-m}(0,\vartheta,\varphi)$	$l = j \pm 1/2;\ \lambda = \pm 1/2$
	covariant	$\left[\Omega^{l}_{jm}(\vartheta,\varphi)\right]^{\prime}_{\lambda} = (-1)^{1/2-\lambda}\left[\Omega^{l}_{jm}(\vartheta,\varphi)\right]^{\prime-\lambda}$	$l = j \pm 1/2;\ \lambda = \pm 1/2$
		Vector spherical harmonics	
Spherical spin basis (A.17)	contravariant	$\left[\mathbf{Y}^{l}_{jm}(\vartheta,\varphi)\right]^{\sigma} = \langle lm-\sigma, 1\sigma \mid jm\rangle Y_{l,m-\sigma}(\vartheta,\varphi)$	$l = j, j \pm 1;\ \sigma = 0, \pm 1$
	covariant	$\left[\mathbf{Y}^{l}_{jm}(\vartheta,\varphi)\right]_{\sigma} = (-1)^{\sigma}\left[\mathbf{Y}^{l}_{jm}(\vartheta,\varphi)\right]^{-\sigma}$	$l = j, j \pm 1;\ \sigma = 0, \pm 1$
Spherical helicity basis (A.21)	contravariant	$\left[\mathbf{Y}^{l}_{jm}(\vartheta,\varphi)\right]^{\prime\lambda} = \sqrt{\frac{2l+1}{4\pi}}\langle l0, 1\lambda \mid j\lambda\rangle D^{j}_{-\lambda,-m}(0,\vartheta,\varphi)$	$l = j, j \pm 1;\ \lambda = 0, \pm 1$
	covariant	$\left[\mathbf{Y}^{l}_{jm}(\vartheta,\varphi)\right]^{\prime}_{\lambda} = (-1)^{\lambda}\left[\mathbf{Y}^{l}_{jm}(\vartheta,\varphi)\right]^{\prime-\lambda}$	$l = j, j \pm 1;\ \lambda = 0, \pm 1$

Gaunt coefficients. Among the known set of integrals over the spherical harmonics, there are a few integrals which occur very frequently in the description of many-particle systems and to which a special notation has been assigned. The Gaunt coefficient, for instance, denotes the integral over a product of three spherical harmonics with the same argument $\Omega \equiv (\vartheta, \varphi)$

$$\langle l_a m_a | l_b m_b | l_c m_c \rangle \equiv \int \mathrm{d}\Omega \, Y^*_{l_a m_a}(\Omega) Y_{l_b m_b}(\Omega) Y_{l_c m_c}(\Omega). \tag{13}$$

In the study of atomic and molecular systems, the Gaunt coefficients naturally appear in the linearization of the product of two orbital angular momentum states

$$Y_{l_b m_b}(\Omega) Y_{l_c m_c}(\Omega) = \sum_{l_a, m_a} \langle l_a m_a | l_b m_b | l_c m_c \rangle Y_{l_a m_a}(\Omega)$$

as well as in the decomposition of reducible representations of the rotation group into irreducible representations.

Condon–Shortley coefficients. The Gaunt coefficients are closely related also to the Condon–Shortley coefficients $C^k(l_a, m_a; l_b, m_b)$ which are defined by the integral

$$C^k(l_a, m_a; l_b, m_b) = \left[\frac{4\pi}{2k+1} \right]^{1/2} \int \mathrm{d}\Omega \, Y^*_{l_a m_a}(\Omega) Y_{k, m_a - m_b}(\Omega) Y_{l_b m_b}(\Omega). \tag{14}$$

The Condon–Shortley coefficients are required in atomic and molecular Hartree–Fock calculations, for instance, in order to evaluate the electron–electron interaction matrix.

Coefficients of fractional parentage and reduced matrix elements. Several standard quantities from the atomic shell model such as the (reduced) coefficients of fractional parentage and the matrix elements of the unit tensors, both in LS- and jj-coupling, have been implemented before into the RACAH program [8]. They are augmented now also by the LS–jj transformation matrices and the corresponding transformation of symmetry-adapted configuration and atomic state functions [7].

Apart from the (reduced) coefficients of fractional parentage and the matrix elements of the unit tensors, all procedures from Table 1 have been implemented *in addition* to the previous version of the RACAH program [6]. These commands are designed in order to support *fast computations* or, at least, to provide a quick reference to a wider range of symbols and functions from the theory of angular momentum. All procedures, however, can be invoked with either symbolic or numerical arguments. Of course, the numerical evaluation of an expression is possible only if all arguments are of *numerical* type or, in the case of the Wigner n–j symbols, if they belong to a *special-value*. In all other cases, a number of *tests* are made on the consistency of the parameters but, otherwise, the commands return *unevaluated*. Then, a simplification may still occur later in a particular computation—if the arguments evaluate properly. In practice, most computations are traced back internally to the calculation of the Wigner $3n$–j symbols and the spherical harmonics.

2.2. Hardware floating-point computations

The incorporation of *numerical* procedures into the RACAH program follows a line which differs from our previous design. In most earlier versions, namely, emphasis was paid mainly to the algebraic transformation and simplification of Racah expressions [5] which (may) contain the Wigner n–j symbols, the spherical harmonics [11] and/or general recoupling coefficients [6]. However, a straightforward extension of this concept—aiming to incorporate also other quantities from the theory of angular momentum and spherical tensor operators at an *algebraic* level—appears as a very *elaborate* task which, in practice, may require a long-time effort. In this light, the *numerical support* of a wider class of symbols and functions can be considered as a *first step* which helps the user already now to tackle (more sophisticated) tasks within the RACAH framework. For similar reasons also, most modern CA environments have recently improved their support of hardware operations and their tools for visualization. Beside of its *software model*, for instance, which basically allows the 'exact' transformation and computation (including numbers of rather arbitrary lengths), MAPLE now supports the use of much faster floating-point operations (its so-called *hardware model*). For these two models, however, efficient algorithms are often quite different from each other and still need to be developed for most *non-standard* applications.

As mentioned above, the calculation of expressions from the theory of angular momentum can be traced back very often to the computation of the Wigner $3n$–j symbols and spherical harmonics. Therefore, special care has been taken to implement, in particular, the 3–j and 6–j symbols in an efficient way using their symmetries and recurrence relations [12]. In the program, however, the development of such *hardware floating-point* (hf) procedures are kept well separated from the previous code for symbolic manipulations. These hf procedures are called only if the parameters in the commands of Table 1 allow such floating-point operations without that a loss of accuracy can occur, i.e. that the requested number of `Digits` is appropriate for floating-point operations. In all other cases, the computations are carried out as before within MAPLE's software model, often by utilizing the basic definition of these quantities in terms of more elementary functions. For the hf procedures, we internally use the *name convention* Racah_compute_quantity_hf() where quantity refers to the name of the symbol (or procedure) as shown in Table 1.

Here, we will not explain the implementation of the individual symbols in detail. A central role in the calculation of many quantities is played by the recursive computation of the Wigner 3–j and 6–j symbols due to an algorithm of Schulten and Gordon [12] as well as by the careful choice of the expansion which is used to evaluate the spherical harmonics. Appendix B summarizes the recursion relations which are utilized for the Wigner 3–j and 6–j symbols; for the computation of the 9–j symbols, a proper summation over products of three 6–j's is applied internally. While, however, the implementation of 'symbolic transformations' often require a very clear and formal program structure, a higher efficiency is sometimes obtained in numerical computations by making use of a (not-so-obvious) re-arrangement of data and/or operations. Moreover, some of the numerical algorithms are suitable only for a restricted range of parameters which are hard to recognize from a given implementation. Our strict separation of the *software* and *hardware* procedures of the RACAH program will therefore help to append further numerical algorithms without that the symbolic part of the code is disturbed. The explicit development and incorporation of such (additional) algorithms will however depend on the *feedback* which we will obtain from the user community.

3. Further modifications and distribution of the code

Recent MAPLE releases have provided a number of syntax extensions (and modifications) as appropriate for a modern language. Of course, the central idea behind these improvements is to facilitate the program design and maintenance. Two important syntax extensions from the recent years concern, for example, the automatic *type checking* of the (incoming) arguments and a simple incorporation of *user-defined types*. The use of these two extensions have simplified the communication and data transfer within the RACAH program considerably. Moreover, the `type` check of user-defined data structures can now be handled by MAPLE's standard syntax such as `type(a,keyword)`, where a denotes any variable or expression, and `true` or `false` is returned in dependence whether a meets the internal representation of the *keyword-type* or not. Table 3 lists the *type definitions* which have been appended by the RACAH program to the inherent types of MAPLE. Other syntax modifications such as the (new) constructs for ... do ... end do or if ... elif ... else ... endif, and others are now also exploited within RACAH and helped improve the readability and re-use of the code.

Another great benefit for the implementation of large software packages in MAPLE arises from the proper use of *modules* which help to encapsulate, to maintain, and to install the code. Moreover, the use of modules facilitates the *hiding* of internal data and program structures since all commands, which are provided to the user, must be *exported* explicitly. We make use of this feature also for the RACAH program which is provided now in terms of

Table 3
New `type` definitions for the RACAH program. A boolean value of either `true` or `false` is returned

`type(a,...)`	Returns `true` if a belongs to or is a ...
`type(a,halfinteger)`	$\ldots, -1/2, 0, 1/2, 1, 3/2, \ldots$
`type(a,halfnonposint)`	$\ldots, -3/2, -1, -1/2, 0$
`type(a,halfnonnegint)`	$0, 1/2, 1, 3/2, \ldots$
`type(a,halfnegint)`	$\ldots, -3/2, -1, -1/2$
`type(a,halfposint)`	$1/2, 1, 3/2, \ldots$
`type(a,Racahexpr)`	Racah expression, see Ref. [4]
`type(a,Racahsum)`	Racah sum, i.e. a sum of Racah expressions
`type(a,tdelta)`	Kronecker δ_{ab}, a triangular $\delta(a,b,c)$ or a Dirac function $\delta(x-x')$
`type(a,delta)`	Kronecker δ_{ab}
`type(a,triangle)`	triangular $\delta(a,b,c)$
`type(a,dirac)`	Dirac function $\delta(x-x')$
`type(a,wnj)`	Wigner n–j symbol ($n = 3$, 6, or 9)
`type(a,w3j)`	Wigner 3–j symbol
`type(a,w6j)`	Wigner 6–j symbol
`type(a,w9j)`	Wigner 9–j symbol
`type(a,Ylm)`	spherical harmonic

two modules: While (i) the Racah module comprises all procedures for the symbolic and numerical treatment of Racah expressions [4–6], the (ii) Jucys module contains the quantities from the atomic shell model [8] as well as the LS–jj transformation matrices and procedures [7]. Both modules can be invoked simply by with(Racah) and with(Jucys) and, altogether, contain more than 240 procedures or about 50,000 lines of code and data. To make use of Jucys' functionality requires, however, that the Racah module has been loaded before.

The extension of RACAH towards applications from the atomic shell model [7,8] required to incorporate a large number of *data* for the classification of the subshell states. In the previous versions, these data needed to be re-initialized explicitly during the execution of the program, in dependence of the particular application. With the (new) *module* structure of MAPLE, a faster access onto this quantities has now be achieved. These data for the classification and transformation of atomic shell states are all contained in the Jucys module. With the adaptation of the RACAH program to the present standard, we also included a number of new *help pages* which, by a proper installation, are incorporated directly into MAPLE's help facilities. Moreover, a brief list of all (exported) procedures of the individual modules can be obtained by means of Racah_help(Racah) or Racah_help(Jucys), respectively.

The design and set-up of modules also facilitates the distribution of the code. As before, the whole package is distributed by the tar file `Racah2002.tar` from which the `Racah2002` root directory is (re-)generated by the command `tar -xvf Racah2002.tar`. This root contains the source code libraries (for MAPLE 7 and 8), a `Read.me` for the installation of the program as well as the document `Racah-commands.ps`. This document provides the definition of all *data structures* of the RACAH program as well as an alphabetic list of all user relevant (and exported) commands. The `Racah2002` root also contains an example of a `.mapleinit` file which can easily be modified and incorporated into the user's `home`. Making use of such a `.mapleinit` file, then, the two modules Racah and Jucys should be available like any other module of MAPLE.

4. Run-time comparison between the different computational models

In many applications, the *reliable* but also *fast* access to the symbols and functions from Section 2 appears to be of quite similar importance. Therefore, in order to demonstrate the acceleration in the computations due to the use of the particularly designed numerical algorithms and hardware floating-point operations, here we provide a brief run-time comparison between the *software* and *hardware* procedures of the RACAH program. This comparison shows the capabilities of the presently revised version and may help the user in making the decision, whether a given problem can be solved entirely within the framework of RACAH or if an interface between the symbolic evaluation and the (final) numerical computations need to be developed.

In practice, of course, the overall gain in the efficiency of a computation by means of hardware-adapted algorithms will usually depend on the particular requirements of the given task. If one considers the wide range of possible parameters and applications, therefore, a single *gain factor* is hardly to be obtained for the various cases. However, to provide a good fingerprint on the acceleration due to the use of numerical procedures, we may define and calculate a few representative Racah expression—utilizing both, RACAH's software and hardware procedures. Since most quantities from Table 1 are traced back internally to the computation of the Wigner $n\!-\!j$ symbols and spherical harmonics, the following two (finite) summations

$$\sum_{a=0}^{l_{\max}} \sum_{c=|a-5|}^{a+5} \begin{pmatrix} a & 5 & c \\ -c & 1 & c-1 \end{pmatrix}, \tag{15}$$

$$\sum_{a=0}^{l_{\max}} \sum_{c=|a-5|}^{a+5} \sum_{d=|c-2|}^{c+2} \sum_{f=|a-2|}^{a+2} \begin{Bmatrix} a & 5 & c \\ d & 2 & f \end{Bmatrix} \tag{16}$$

are defined for the Wigner $3\!-\!j$ and $6\!-\!j$ symbols, respectively, which only depend on a single parameter ($l_{\max} \geqslant 0$) and which, in dependence on $l_{\max}$, are quite representative for rather a wide range of parameters. For the spherical harmonics, similarly, we define the (finite) sum

Table 4
Run-time comparison between RACAH's software and hardware procedures for the finite summations (15)–(17). A comparison is made in dependence of the parameter $l_{\max}$ which is representative for a wide range of angular momentum quantum numbers. N is the number of terms in the corresponding summation and the (run) times are given in seconds. All computations have been carried out by means of a 1000 MHz Pentium III processor. The values of these (rather arbitrary) expressions are also given in order to facilitate a test of the installation

Expression	$l_{\max}$	N	Time	Value	Comments
Expr. (15)	10	91	$\ll 1$	0.6399280116	hardware model, `Digits = 12`
			<1		software model, `Digits = 12`
			<1		software model, `Digits = 24`
	200	2181	–	[a]	soft- and hardware model, `Digits = 12`
			6	9.354589477	software model, `Digits = 24`
Expr. (16)	10	2175	1	1.418802038	hardware model, `Digits = 12`
			5		software model, `Digits = 12`
			7		software model, `Digits = 24`
	50	13175	6	2.239234279	hardware model, `Digits = 12`
			44		software model, `Digits = 12`
			53		software model, `Digits = 24`
Expr. (17)	10	121	$\ll 1$	{7.218872080	hardware model, `Digits = 12`
			1	–5.816808893i}	software model, `Digits = 12`
			1		software model, `Digits = 24`
	30	961	<1	{–12.04709382	hardware model, `Digits = 12`
			7	–29.09665235i}	software model, `Digits = 12`
			9		software model, `Digits = 24`

[a] Computation not possible for the given number of `Digits` due to the large factorials which arise in the evaluation.

$$\sum_{l=0}^{l_{\max}} \sum_{m=-l}^{l} Y_{lm}(\vartheta_0, \varphi_0) \tag{17}$$

which we evaluate below for $\vartheta_0 = 0.166$ and $\varphi_0 = 0.675$, the approximate (geographical) coordinates of our university in Kassel. For these three expressions (15)–(17), Table 4 compares the run-time requirements of the software and hardware computations for different values of $l_{\max}$, i.e. different ranges of parameters, and for a different number of (valid) `Digits`. In this table, all computations were carried out by means of a 1000 MHz Pentium III processor. Moreover, we also display the (rather arbitrary) values of the corresponding sums in order to provide the user with a simple test of his or her installation. Overall, a gain of about a factor $5\ldots 8$ is obtained if the given parameters and the required accuracy of the computations allow for the use of RACAH's hardware procedures. A similar gain can be expected also for most other quantities from Table 1 which, internally, refer to the Wigner $3n$–j symbols and spherical harmonics.

In conclusion, a new and quite extended version of the RACAH program is provided which incorporates a large set of new symbols and functions from the theory of angular momentum. Since the full support of (all known) symbolic transformations will require a long-term effort, emphasize has first been paid to the fast and reliable computation of these symbols. Apart from MAPLE's (standard) software model, we now also support much faster *hardware computations*—if this happens to be possible for the given set of parameters. For a further acceleration of these computations, moreover, efficient *numerical algorithms* have been implemented for the Wigner 3–j and 6–j symbols, the spherical harmonics as well as the Gaunt coefficients. We hope and expect that with this *hybrid* solution, which combines software and hardware algorithms from the theory of angular momentum, RACAH's range of applications will be considerably enlarged within the near future.

Acknowledgement

We like to thank G. Gaigalas for his help in the development and test of the code.

Appendix A. Definition of standard coefficients and mathematical relations

For later reference, here we compile the definition and a few important relations for (most of) the quantities from Table 1. These relations are used internally for the algebraic rearrangement of expressions as well as for the numerical computations within MAPLE's *software and/or hardware floating-point model*. They also provide the user with the rules and phase or normalization conventions which we follow within the RACAH program. For the Wigner 3–j and 6–j symbols, moreover, a set of *efficient* but usually more sophisticated algorithms have been implemented for the further acceleration of the *hardware* computations; a brief outline of these numerical algorithms is given below in Appendix B.

A.1. Wigner $3n$–j symbols ($n = 1, 2$, and 3) and Clebsch–Gordan coefficients

An explicit representation of the Wigner 3–j and 6–j symbols in terms of sums of factorials has been given in Ref. [4], Appendix A. These expression are used internally for all computations of the Wigner $3n$–j symbols ($n = 1, 2$, and 3) within MAPLE's software model whereby the 9–j symbols are obtained from a finite sum over products of three 6–j symbols [cf. Ref. [4], Eq. (17)]. Beside of numerical computations, however, the RACAH program also knows a large number of sum rules for the Wigner $3n$–j symbols which can be utilized for algebraic manipulations and simplifications of typical Racah expressions [5]. Moreover, the program recognizes a variety of special symbols for the Wigner 3–j and 6–j symbols, if these coefficients cannot be evaluated numerically. In order to support fast hardware floating-point computations for the Wigner 3–j and 6–j symbols, a recursive procedure due to Schulten and Gordon [12] is implemented which is summarized in Appendix B.1 below. Recent progress has been made in this field also by Roothaan and Lai [13] who applied a new algebraic method due to Labarthe [14] to derive explicit formulas for the computation of the $3n$–j symbols.

No explicit representations are known internally for the Clebsch–Gordan coefficients. As mentioned in Section 2, these coefficients are first always *reduced* to the Wigner 3–j symbols, before any computation or manipulation is carried out using the phase convention of Condon and Shortley [10].

A.2. Wigner 12–j symbols

The Wigner 12–j symbols are related to the unitary transformation of five or more (coupled) angular momenta between various coupling schemes. There exist 12–j symbols of two different kinds, called the *first* and *second* kind, which are usually expressed in terms of the Wigner 6–j symbols. For the *first* kind, the 12–$j(1)$ symbols, such an expansion reads [3]

$$\left\{\begin{matrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \end{matrix}\,\middle|\, 1\right\} = \sum_x (-1)^{S-x}[x] \begin{Bmatrix} a_1 & a_2 & b_1 \\ c_2 & c_1 & x \end{Bmatrix} \begin{Bmatrix} a_2 & a_3 & b_2 \\ c_3 & c_2 & x \end{Bmatrix} \times \begin{Bmatrix} a_3 & a_4 & b_3 \\ c_4 & c_3 & x \end{Bmatrix} \begin{Bmatrix} a_4 & c_1 & b_4 \\ a_1 & c_4 & x \end{Bmatrix} \tag{A.1}$$

with $S = \sum_{i=1}^{4}(a_i + b_i + c_i)$, while the 12–$j(2)$ symbols of the *second* kind are simpler expressed in terms of Sharp's symbol [9]

$$\left\{\begin{matrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \end{matrix}\,\middle|\, 2\right\} = (-1)^{a_1+a_2+a_3+a_4-c_1-c_2-c_3-c_4} \begin{Bmatrix} - & a_4 & a_1 & b_4 \\ c_2 & - & b_1 & c_1 \\ c_3 & b_3 & - & c_4 \\ b_2 & a_3 & a_2 & - \end{Bmatrix} \tag{A.2}$$

which obeys a slightly higher symmetry. A representation of Sharp's symbol in terms of the Wigner 6–j's is given by [3]

$$\begin{Bmatrix} - & a_2 & a_3 & a_4 \\ b_1 & - & b_3 & b_4 \\ c_1 & c_2 & - & c_4 \\ d_1 & d_2 & d_3 & - \end{Bmatrix} = (-1)^{b_3-a_4-d_1+c_2} \sum_x [x] \begin{Bmatrix} a_3 & b_4 & x \\ b_1 & d_3 & b_3 \end{Bmatrix} \begin{Bmatrix} a_3 & b_4 & x \\ c_4 & a_2 & a_4 \end{Bmatrix} \times \begin{Bmatrix} b_1 & d_3 & x \\ d_2 & c_1 & d_1 \end{Bmatrix} \begin{Bmatrix} c_4 & a_2 & x \\ d_2 & c_1 & c_2 \end{Bmatrix}. \tag{A.3}$$

In the RACAH program, Eqs. (A.1)–(A.3) are utilized by the (new) command Racah_w12j() for the numerical computation of the 12–j symbols (of both kinds) as well as of Sharp's symbol. They are applied both, in the software model as well as for hardware floating-point computations, exploiting the fast algorithm for the Wigner 6–j symbols. Further algebraic properties of these symbols are not known to the program.

A.3. Wigner D-functions and rotation matrices

The Wigner D-functions $D^j_{mm'}(\alpha, \beta, \gamma)$ are usually defined as the matrix elements of the rotation operator $\widehat{D}(\alpha, \beta, \gamma)$ in the jm-representation

$$\langle jm | \widehat{D}(\alpha, \beta, \gamma) | j'm' \rangle = \delta_{jj'} D^j_{mm'}(\alpha, \beta, \gamma), \tag{A.4}$$

see Varshalovich et al. [3], Chapter 4. In this definition, $\alpha, \beta,$ and γ are the Euler angles which specify the rotation of either the physical system or, often more conveniently, the coordinates. The Wigner D-function also help realize the transformation of irreducible tensors of rank j, i.e. the $(2j+1)$-dimensional matrices $D^{j}_{mm'}(\alpha, \beta, \gamma)$ in m and m' are unimodular

$$\det\left|D^{j}_{mm'}(\alpha, \beta, \gamma)\right| = +1. \tag{A.5}$$

Usually, the Wigner D-functions are represented as a product of three functions

$$D^{j}_{mm'}(\alpha, \beta, \gamma) = \mathrm{e}^{-\mathrm{i}m\alpha} d^{j}_{mm'}(\beta) \mathrm{e}^{-\mathrm{i}m'\gamma} \tag{A.6}$$

which depend each on just a single Euler angle; the function $d^{j}_{mm'}(\beta)$ is taken to be real and is often called Wigner's rotation matrix. In the RACAH program, the Wigner D-functions are always treated by means of expression (A.6) and the corresponding rotation matrix $d^{j}_{mm'}(\beta)$.

For the rotation matrices, a large number of explicit representations in terms of different functions or derivatives as well as several integral representations are listed by Varshalovich et al. [3], Section 4.3. From these representations, we use the four expressions

$$\begin{aligned} d^{j}_{mm'}(\beta) &= (-1)^{j-m'}\big[(j+m)!(j-m)!(j+m')!(j-m')!\big]^{1/2} \\ &\quad \times \sum_{k} (-1)^{k} \frac{(\cos\frac{\beta}{2})^{m+m'+2k}(\sin\frac{\beta}{2})^{2j-m-m'-2k}}{k!(j-m-k)!(j-m'-k)!(m+m'+k)!}, \end{aligned} \tag{A.7}$$

$$\begin{aligned} d^{j}_{mm'}(\beta) &= (-1)^{j+m}\big[(j+m)!(j-m)!(j+m')!(j-m')!\big]^{1/2} \\ &\quad \times \sum_{k} (-1)^{k} \frac{(\cos\frac{\beta}{2})^{2k-m-m'}(\sin\frac{\beta}{2})^{2j+m+m'-2k}}{k!(j+m-k)!(j+m'-k)!(k-m-m')!}, \end{aligned} \tag{A.8}$$

$$\begin{aligned} d^{j}_{mm'}(\beta) &= \big[(j+m)!(j-m)!(j+m')!(j-m')!\big]^{1/2} \\ &\quad \times \sum_{k} (-1)^{k} \frac{(\cos\frac{\beta}{2})^{2j-2k+m-m'}(\sin\frac{\beta}{2})^{2k-m+m'}}{k!(j+m-k)!(j-m'-k)!(m'-m+k)!}, \end{aligned} \tag{A.9}$$

$$\begin{aligned} d^{j}_{mm'}(\beta) &= (-1)^{m-m'}\big[(j+m)!(j-m)!(j+m')!(j-m')!\big]^{1/2} \\ &\quad \times \sum_{k} (-1)^{k} \frac{(\cos\frac{\beta}{2})^{2j-2k-m+m'}(\sin\frac{\beta}{2})^{2k+m-m'}}{k!(j-m-k)!(j+m'-k)!(m-m'+k)!} \end{aligned} \tag{A.10}$$

for all numerical computations, in dependence of the factorials which occur in the denominators of (A.7)–(A.10). Both, MAPLE's software and hardware floating-point computations are supported along these lines.

In addition to the Wigner $D^{j}_{mm'}(\alpha, \beta, \gamma)$ and $d^{j}_{mm'}(\beta)$ functions, which are expressed in terms of the Euler angles, only the rotation matrix $U^{j}_{mm'}(\omega; \Theta, \Phi)$ [cf. Section 2] are also supported by the program. Several other variants of the rotation matrix are known but are not incorporated (yet) into the program.

A.4. Spherical harmonics; bipolar and tripolar harmonics

For the last two years [11], the symmetries and properties of the spherical harmonics $Y_{lm}(\vartheta, \varphi)$ are now also known to the RACAH program and are exploited for symbolic manipulations of Racah expressions. By using the properties of the spherical harmonics, however, products of two or three of these functions can be *combined* also to represent irreducible tensors of two (three) solid angles. Eqs. (5) and (7) give the definition of the *bipolar* and *tripolar* spherical harmonics as irreducible tensors of rank L. For functions which depend on two (three) unit vectors, say $\mathbf{n}_1$, $\mathbf{n}_2$, (and $\mathbf{n}_3$,) the bipolar (tripolar) spherical harmonics with different $l_1, l_2, (l_3,) L, M$ form a complete and orthonormal set. For the bipolar spherical harmonics, for instance, the orthogonality and normalization is

$$\iint \mathrm{d}\Omega_1\,\mathrm{d}\Omega_2\,\{\mathbf{Y}_{l_1}(\Omega_1)\otimes\mathbf{Y}_{l_2}(\Omega_2)\}^*_{LM}\{\mathbf{Y}_{l'_1}(\Omega_1)\otimes\mathbf{Y}_{l'_2}(\Omega_2)\}_{L'M'} = \delta_{l_1l'_1}\delta_{l_2l'_2}\delta_{LL'}\delta_{MM'}, \tag{A.11}$$

while the corresponding completeness condition takes the form

$$\sum_{l_1l_2LM}\{\mathbf{Y}_{l_1}(\Omega'_1)\otimes\mathbf{Y}_{l_2}(\Omega'_2)\}^*_{LM}\{\mathbf{Y}_{l_1}(\Omega_1)\otimes\mathbf{Y}_{l_2}(\Omega_2)\}^*_{LM} = \delta(\Omega_1-\Omega'_1)\delta(\Omega_2-\Omega'_2) \tag{A.12}$$

with

$$\Omega\equiv\{\vartheta,\varphi\},\qquad \int\mathrm{d}\Omega\equiv\int_0^{2\pi}\mathrm{d}\varphi\int_0^{\pi}\mathrm{d}\vartheta\,\sin\vartheta,$$
$$\delta(\Omega-\Omega')\equiv\delta(\cos\vartheta-\cos\vartheta')\delta(\varphi-\varphi').$$

Analogue relations also apply to the tripolar spherical harmonics. Moreover, for products of two (or more) bipolar or tripolar harmonics with the same arguments Ω_1, Ω_2 (and Ω_3), Clebsch–Gordan expansions in terms of these functions and the Wigner n–j symbols are known and utilized in different areas of physics; for the bipolar spherical harmonics, for instance, one of the Clebsch–Gordan expansions reads as [3]

$$\begin{aligned}&\{\mathbf{Y}_{l'_1}(\Omega_1)\otimes\mathbf{Y}_{l'_2}(\Omega_2)\}_{L'M'}\{\mathbf{Y}_{l''_1}(\Omega_1)\otimes\mathbf{Y}_{l''_2}(\Omega_2)\}_{L''M''}\\&\quad=\sum_{LM}\langle L'M',L''M''|LM\rangle\sum_{l_2l_2}B(l'_1l'_2L';l''_1l''_2L'';l_1l_2L)\{\mathbf{Y}_{l_1}(\Omega_1)\otimes\mathbf{Y}_{l_2}(\Omega_2)\}_{LM},\end{aligned} \tag{A.13}$$

where

$$B(l'_1l'_2L';l''_1l''_2L'';l_1l_2L)=\frac{[l'_1,l'_2,l''_1,l''_2,L',L'',]^{1/2}}{4\pi}\langle l'_10,l''_10|l_10\rangle\langle l'_20,l''_20|l_20\rangle\begin{Bmatrix}l'_1 & l''_1 & l_1\\ l'_2 & l''_2 & l_2\\ L' & L'' & L\end{Bmatrix} \tag{A.14}$$

and $[a,b,\ldots]=(2a+1)(2b+1)\ldots$. A number of other Clebsch–Gordan expansions are also found in the literature. Until now, however, only the computation of the bipolar and tripolar harmonics is supported by the RACAH program; for fast numerical computations, the corresponding algorithms of the spherical harmonics are used.

A.5. Spin functions and tensor spherical harmonics

In quantum physics, the spin functions are used to describe the polarization properties of particles and composite systems with definite spin, i.e. a definite intrinsic angular momentum [3]. Usually, the spin functions are treated as functions of some discrete variable σ which, for instance, may represent the spin projection onto the z-axis. For a given spin s, these functions are commonly written as column matrices

$$\chi=\begin{pmatrix}\chi(s)\\ \chi(s-1)\\ \vdots\\ \chi(-s)\end{pmatrix} \tag{A.15}$$

with $(2s+1)$ elements which represent the values of the spin function $\chi(\sigma)$ for the corresponding value of σ. Accordingly, the Hermitian conjugate function $\chi^\dagger$ takes the form of a row matrix

$$\chi^\dagger = \big(\chi^*(s), \chi^*(s-1), \ldots, \chi^*(-s)\big). \tag{A.16}$$

Several definitions of the spin functions are known from the literature, in dependence of the choice of the quantization axis and the context of application. With space-fixed coordinates, one often uses (i) the (*spherical*) *spin basis functions*, while (ii) the (*spherical*) *helicity basis functions* are applied for particles with a well-defined linear momentum.

The spherical spin basis functions are eigenfunctions of

$$\mathbf{s}^2 \chi_{sm} = s(s+1)\hbar^2 \chi_{sm}, \qquad s_z \chi_{sm} = m\hbar \chi_{sm}, \tag{A.17}$$

and, thus, have the *contravariant components*

$$[\chi_{sm}]^\sigma = \delta_{\sigma m}. \tag{A.18}$$

For $s = 1/2$ particles, in particular, the spin basis functions χ_{sm} $(m = \pm 1/2)$ are usually written as

$$\chi_{\frac{1}{2}\frac{1}{2}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \qquad \chi_{\frac{1}{2}-\frac{1}{2}} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \tag{A.19}$$

while, for $s = 1$ particles, these functions χ_{sm} $(m = \pm 1, 0)$ are given by

$$\chi_{11} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \qquad \chi_{10} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \qquad \chi_{1-1} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \tag{A.20}$$

In general, these sets of $2s + 1$ basis functions χ_{sm} $(m = s, s-1, \ldots, -s)$ form a complete and orthonormal basis in the according spin space.

If the linear momentum $\mathbf{p}$ of a particle is taken as quantization axis, the spin projection λ along this axis is called the *helicity*. Again, the helicity $\lambda = s, s-1, \ldots, -s$ takes $2s + 1$ possible values; however, since the helicity basis functions are eigenfunctions of $\mathbf{s}^2$ and $\mathbf{s} \cdot \mathbf{n} \equiv \mathbf{s} \cdot \mathbf{p}/p$

$$\mathbf{s} \cdot \mathbf{n} \chi_{s\lambda}(\vartheta, \varphi) = \lambda\hbar \chi_{s\lambda}(\vartheta, \varphi), \tag{A.21}$$

these functions depend on the polar angles ϑ and φ (of the momentum vector $\mathbf{p}$ or $\mathbf{n}$, respectively).

The helicity basis function $\chi_{s\lambda}(\vartheta, \varphi)$ can be generated from the spin basis functions (A.17) by means of a rotation which turns the (space-fixed) z-axis parallel to $\mathbf{n}(\vartheta, \varphi)$ [3]

$$\chi_{s\lambda}(\vartheta, \varphi) = \sum_m D^s_{m\lambda}(\varphi, \vartheta, 0) \chi_{sm}, \tag{A.22}$$

and vice versa

$$\chi_{sm} = \sum_\lambda D^s_{-\lambda-m}(0, \vartheta, \varphi) \chi_{s\lambda}(\vartheta, \varphi), \tag{A.23}$$

and, hence, the contravariant components of the helicity basis functions are

$$\big[\chi_{s\lambda}(\vartheta, \varphi)\big]^\sigma = D^s_{\sigma\lambda}(\varphi, \vartheta, 0). \tag{A.24}$$

For spin-$\frac{1}{2}$ particles, the explicit form of the helicity basis functions is given by

$$\chi_{\frac{1}{2}\ \frac{1}{2}}(\vartheta, \varphi) = \begin{pmatrix} \cos\frac{\vartheta}{2} e^{-i\varphi/2} \\ \sin\frac{\vartheta}{2} e^{i\varphi/2} \end{pmatrix}, \qquad \chi_{\frac{1}{2}\ -\frac{1}{2}}(\vartheta, \varphi) = \begin{pmatrix} -\sin\frac{\vartheta}{2} e^{-i\varphi/2} \\ \cos\frac{\vartheta}{2} e^{i\varphi/2} \end{pmatrix}. \tag{A.25}$$

Using the definition of the basis functions (A.17) or (A.21), one often wishes to construct tensor spherical harmonics

$$Y_{jm}^{ls}(\vartheta,\varphi) \equiv \{\mathbf{Y}_l \otimes \chi_s\}_{jm} = \sum_{m_l,m_s} Y_{lm_l}(\vartheta,\varphi)\chi_{sm_s}\langle lm_l, sm_s \mid jm\rangle \tag{A.26}$$

which are obtained from the coupling of the orbital angular momentum $\mathbf{l}$ and spin $\mathbf{s}$ to a total angular momentum $\mathbf{j} = \mathbf{l} + \mathbf{s}$ and, thus, are eigenfunctions of $\mathbf{l}^2, \mathbf{s}^2, \mathbf{j}^2, j_z$. Obviously, the $2j+1$ functions $Y_{jm}^{ls}(\vartheta,\varphi)$ with $m = j, j-1, \ldots, -j$ form a irreducible tensor of rank j; moreover, the tensor spherical harmonics $Y_{jm}^{ls}(\vartheta,\varphi)$ also provide a complete and orthonormal set of functions for the expansion of rank s tensor functions with the domain of arguments $0 \leqslant \theta \leqslant \pi$, $0 \leqslant \phi \leqslant 2\pi$. In the RACAH program, we currently support the (numerical) computation of the contravariant components with respect to the spherical spin basis functions (A.17) and helicity basis functions (A.21) for arbitrary spin s. In addition, special procedures are provided for the more frequently used spinor ($s = 1/2$) and vector ($s = 1$) spherical harmonics; see Table 2 and the following Section A.6. Apart from the (contravariant) components no further properties such as the transformation coefficients of the spin functions, the spin density matrices, or others are (yet) supported by the program.

A.6. Spinor and vector spherical harmonics

For $s = 1/2$ and $s = 1$, special symbols and notations are used for the tensor spherical harmonics (A.26)

$$\Omega_{jm}^{l}(\vartheta,\varphi) \equiv Y_{jm}^{l\frac{1}{2}}(\vartheta,\varphi), \tag{A.27}$$

$$\mathbf{Y}_{jm}^{l}(\vartheta,\varphi) \equiv Y_{jm}^{l1}(\vartheta,\varphi) \tag{A.28}$$

which are called the spinor and vector spherical harmonics in these cases. They are typically expressed by 2×1 (and 3×1) column matrices which help replace the summation over the spin variable by a matrix multiplication. For the spinor and vector spherical harmonics, the program supports both, the contravariant as well as covariant components in the (spherical) spin and helicity basis. Table 2 lists the definition of the various components as utilized by the RACAH program.

Appendix B. Fast numerical algorithms

Algorithms for fast numerical computations are known for various quantities from the literature. They often benefit from particular restrictions which apply to certain classes of problems; other algorithms sometimes uses *predefined* arrays of numbers from which the results are obtained more easily than from a direct computation. In the RACAH program, we presently support only the fast recursive computation of (a whole set of) the Wigner 3–j and 6–j symbols, both in the software as well as hardware floating-point model. These recursive relations are then also used for the computation of the Gaunt coefficients.

B.1. Schulten–Gordon recurrence for the Wigner 3–j and 6–j symbols

A fast recursive computation of the Wigner 3–j and 6–j symbols and, in particular for large values of the arguments or for complete sets of such coefficients, were first implemented in a FORTRAN code by Schulten and Gordon [12]. In this previous implementation, use was made of the two (*one-step*) recursion relations for the j- and m-values in the Wigner 3–j symbols

$$aA(a+1)\begin{pmatrix} a+1 & b & c \\ m_a & m_b & m_c \end{pmatrix} + B(a)\begin{pmatrix} a & b & c \\ m_a & m_b & m_c \end{pmatrix} + (a+1)A(a)\begin{pmatrix} a-1 & b & c \\ m_a & m_b & m_c \end{pmatrix} = 0 \tag{B.1}$$

with

$$\begin{aligned} A(a) &= \sqrt{a^2-(b-c)^2}\,\sqrt{(b+c+1)^2-a^2}\,\sqrt{a^2-m_a^2},\\ B(a) &= -(2a+1)\big[b(b+1)m_a - c(c+1)m_a - a(a+1)(m_c-m_b)\big], \end{aligned} \tag{B.2}$$

and

$$C(m_a+1)\begin{pmatrix} a & b & c\\ m_a+1 & m_b-1 & m_c\end{pmatrix} + D(m_a)\begin{pmatrix} a & b & c\\ m_a & m_b & m_c\end{pmatrix} + C(m_a)\begin{pmatrix} a & b & c\\ m_a-1 & m_b+1 & m_c\end{pmatrix} = 0 \tag{B.3}$$

with

$$\begin{aligned} C(m_a) &= \sqrt{(a-m_a+1)(a+m_a)(b+m_b+1)(b-m_b)},\\ D(m_a) &= a(a+1)+b(b+1)-c(c+1)+2m_a m_b. \end{aligned} \tag{B.4}$$

These linear three-term recursions (B.1) and (B.3) reduce to just two terms for the minimal and maximal values

$$\begin{aligned} a_{\min} &= |b-c|, \qquad a_{\max} = b+c,\\ m_{a_{\min}} &= |-a|, \qquad m_{a_{\max}} = a \end{aligned} \tag{B.5}$$

and, hence, can be utilized for and *downward* or *upward* recursion. In the RACAH program, we apply Eq. (B.1) and a downward recursion for the *recursive* computation of single Wigner 3–j symbols, by starting from the special value

$$\begin{aligned} \begin{pmatrix} b+c & b & c\\ m_a & m_b & m_c\end{pmatrix} &= (-1)^{-b+c-m_b-m_c}[b+c]^{-1/2}\\ &\quad\times\left[\frac{(2b)!(2c)!(b+c+m_b+m_c)!(b+c-m_b-m_c)!}{(2b+2c)!(b+m_b)!(b-m_b)!(c+m_c)!(c-m_c)!}\right]^{1/2} \end{aligned} \tag{B.6}$$

and by using a re-arrangement of the arguments so that $a = \max(a,b,c)$ is always achieved.

A similar (*one-step*) recursion relation also applies for the Wigner 6–j symbols [12]

$$aE(a+1)\begin{pmatrix} a+1 & b & c\\ d & e & f\end{pmatrix} + F(a)\begin{pmatrix} a & b & c\\ d & e & f\end{pmatrix} + (a+1)E(a)\begin{pmatrix} a-1 & b & c\\ d & e & f\end{pmatrix} = 0 \tag{B.7}$$

with

$$\begin{aligned} E(a) &= \sqrt{a^2-(b-c)^2}\,\sqrt{(b+c+1)^2-a^2}\,\sqrt{a^2-(e-f)^2}\,\sqrt{(e+f+1)^2-a^2},\\ F(a) &= (2a+1)\big\{a(a+1)\big[-a(a+1)+b(b+1)+c(c+1)\big]\\ &\quad + e(e+1)\big[a(a+1)+b(b+1)-c(c+1)\big]\\ &\quad + f(f+1)\big[a(a+1)-b(b+1)+c(c+1)\big] - 2a(a+1)d(d+1)\big\} \end{aligned} \tag{B.8}$$

and with *non-zero* a-values in the range

$$a_{\min} = \max\big(|b-c|, |e-f|\big) \quad \ldots \quad a_{\max} = \min(b+c, e+f). \tag{B.9}$$

Again, this recursion equation (B.7) is applied for the *recursive* computation of single Wigner 6–j symbols by arranging the arguments to have $a = \max(a,b,c,d,e,f)$ and $a_{\max} = b+c$, and by using the special value

$$\begin{aligned} \begin{Bmatrix} b+c & b & c\\ d & e & f\end{Bmatrix} &= (-1)^{b+c+e+f}\left[\frac{(2b)!(2c)!(b+c+e+f+1)!(b+c-e+f)!}{(2b+2c+1)!(-b-c+e+f)!(b+f-d)!}\right.\\ &\quad\times\left.\frac{b+c+e-f)!(-b+f+d)!(-c+e+d)!}{(b-f+d)!(b+f+d+1)!(c+e-d)!(c-e+d)!(c+e+d+1)!}\right]. \end{aligned} \tag{B.10}$$

Apart from the (recursive) computation of single Wigner 3–j and 6–j symbols, the recursion equations (B.1), (B.3), and (B.7) above can be also utilized to generate a whole range of coefficients such as

$$\begin{pmatrix} a_{\min}..a_{\max} & b & c \\ m_a & m_b & m_c \end{pmatrix} = \begin{pmatrix} a_{\min} & b & c \\ m_a & m_b & m_c \end{pmatrix}, \begin{pmatrix} a_{\min}+1 & b & c \\ m_a & m_b & m_c \end{pmatrix}, \ldots, \begin{pmatrix} a_{\max} & b & c \\ m_a & m_b & m_c \end{pmatrix}, \quad \text{(B.11)}$$

$$\begin{pmatrix} a & b & c \\ -a..a & . & m_c \end{pmatrix} = \begin{pmatrix} a & b & c \\ -a & a-m_c & m_c \end{pmatrix}, \begin{pmatrix} a & b & c \\ -a+1 & a-1-m_c & m_c \end{pmatrix}, \ldots, \begin{pmatrix} a & b & c \\ a & -a-m_c & m_c \end{pmatrix}, \quad \text{(B.12)}$$

for the Wigner 3–j symbols and, similarly,

$$\begin{Bmatrix} a_{\min}..a_{\max} & b & c \\ d & e & f \end{Bmatrix} = \begin{Bmatrix} a_{\min} & b & c \\ d & e & f \end{Bmatrix}, \begin{Bmatrix} a_{\min}+1 & b & c \\ d & e & f \end{Bmatrix}, \ldots, \begin{Bmatrix} a_{\max} & b & c \\ d & e & f \end{Bmatrix} \quad \text{(B.13)}$$

for the 6–j symbols. For such a *range of coefficients*, however, the recursions must be carried out both, downward and upward from the two non-classical domains and towards the intermediate region of large coupling coefficients, as pointed out by Schulten and Gordon [12]. In this case, moreover, the recursion equations are better applied to determine the Wigner 3–j and 6–j symbols only up to a constant factor which is finally obtained from the following unitary properties and phase conventions [12]

$$\sum_a [a] \begin{pmatrix} a & b & c \\ m_a & m_b & m_c \end{pmatrix}^2 = 1, \qquad \operatorname{sign}\left[\begin{pmatrix} a_{\max} & b & c \\ m_a & m_b & m_c \end{pmatrix}\right] = (-1)^{b-c-ma}, \quad \text{(B.14)}$$

$$\sum_a [a] \begin{pmatrix} a & b & c \\ m_a & m_b & -m_a-m_b \end{pmatrix}^2 = 1, \qquad \operatorname{sign}\left[\begin{pmatrix} a & b & c \\ a & m_b & -a-m_b \end{pmatrix}\right] = (-1)^{2a+b-m_b}, \quad \text{(B.15)}$$

$$\sum_a [a,d] \begin{Bmatrix} a & b & c \\ d & e & f \end{Bmatrix}^2 = 1, \qquad \operatorname{sign}\left[\begin{Bmatrix} a_{\max} & b & c \\ d & e & f \end{Bmatrix}\right] = (-1)^{b+c+e+f}. \quad \text{(B.16)}$$

A *two-side* recursion is generally necessary in order to ensure a numerically stable procedure since the coupling coefficients may oscillate rapidly in the intermediate region of the quantum numbers but decay exponentially to zero towards their boundaries.

In the RACAH program, a recursive computation of single Wigner 3–j and 6–j symbols is implemented, if the keyword *recursive* is given explicitly; moreover, such a recursive procedure is also utilized, if $a+b+c \geqslant 30$ for the Wigner 3–j symbols and $a+b+c+d+e+f \geqslant 40$ for the 6–j symbols, respectively. In the present implementation, moreover, we always utilize—if possible—MAPLE's hardware floating-point model for the computation of these symbols. The recursion relations (B.1), (B.3), and (B.7) are also applied if a (complete) range of coefficients such as (B.11)–(B.13) is to be calculated by a call to Racah_w3j_range() or Racah_w6j_range(), respectively. For all further details, see the user manual in the document Racah-command.ps which is distributed with the program.

B.2. Cruzan's algorithm for the Gaunt coefficients

The efficient computation of the Gaunt coefficients, i.e. the integrals over products of three spherical harmonics, has been recently reviewed and improved by Sebilleau [15]. From his analysis of various algorithms, he suggested to start from the representation of the Gaunt coefficients in terms of the Wigner 3–j symbols

$$\langle l_a m_a | l_b m_b | l_c m_c \rangle = (-1)^{m_a} \left(\frac{[l_a, l_b, l_c]}{4\pi} \right)^{1/2} \begin{pmatrix} l_a & l_b & l_c \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} l_a & l_b & l_c \\ -m_a & m_b & m_c \end{pmatrix}, \quad \text{(B.17)}$$

which was first introduced by Cruzan [16], and to use the recurrence procedures from Appendix B.1 The use of these recursions is supported by the command Racah_Gaunt() if an additional keyword *recursive* is given explicitly or, as default, if $a + b + c \geqslant 30$.

References

[1] G. Racah, Phys. Rev. 61 (1941) 186; Phys. Rev. 62 (1942) 438; Phys. Rev. 63 (1943) 367.
[2] E.P. Wigner, Group Theory and its Application to the Quantum Mechanics of Atomic Spectra, Academic Press, New York, 1959.
[3] D.A. Varshalovich, A.N. Moskalev, V.K. Khersonskii, Quantum Theory of Angular Momentum, World Scientific, Singapore a.o., 1988.
[4] S. Fritzsche, Comp. Phys. Commun. 103 (1997) 51.
[5] S. Fritzsche, S. Varga, D. Geschke, B. Fricke, Comp. Phys. Commun. 111 (1998) 167.
[6] S. Fritzsche, T. Inghoff, T. Bastug, M. Tomaselli, Comp. Phys. Commun. 139 (2001) 314.
[7] G. Gaigalas, S. Fritzsche, Comp. Phys. Commun. 149 (2002) 39.
[8] G. Gaigalas, S. Fritzsche, B. Fricke, Comp. Phys. Commun. 135 (2001) 219.
[9] W.T. Sharp, Chalk River Report TPJ-81, Ontario, 1955.
[10] E.U. Condon, Q.W. Shortley, The Theory of Atomic Spectra, Cambridge University Press, Cambridge, 1935.
[11] T. Inghoff, S. Fritzsche, B. Fricke, Comp. Phys. Commun. 139 (2001) 297.
[12] K. Schulten, R.G. Gordon, Comp. Phys. Commun. 11 (1976) 269.
[13] C.C.J. Roothaan, S.T. Lai, Int. J. Quantum Chem. 63 (1997) 57.
[14] J.J. Labarthe, J. Math. Phys. 27 (1986) 2964.
[15] D. Sebilleau, J. Phys. A 31 (1998) 7157.
[16] O.R. Cruzan, Quart. Appl. Math. 20 (1962) 33.