

DBSR_BREIT (version 3)

The DBSR_BREIT program performs the angular integrations necessary to express the matrix elements of the Dirac-Breit Hamiltonian as a linear combination of radial integrals. The calculations of angular integrals follow the method based upon the representation of atomic configuration wave functions through the Slater determinants. Any amount of non-orthogonality between the orbitals may be present, leading to overlap factors in the matrix elements. The program can effectively reuse data obtained previously by creating a databank for angular coefficients. It can be used in different aspects of atomic-structure and scattering calculations.

1. Introduction

In the atomic-structure calculations of a many-electron system, an atomic wave function is expanded in terms of configuration-state functions, $\Phi(\gamma J)$, where γ refers to the configuration and coupling of shells, and J is the total angular momentum. The configuration-state function (CSF) is the sum of products of one-electron spin-orbitals. Common task is the evaluation of the matrix elements of the atomic Hamiltonian, $\langle \Phi(\gamma J) | H | \Phi(\gamma' J) \rangle$. This can be done in stages by first integrating over all angular and spin coordinates, leaving the radial integrals in an unevaluated form. Such a task is solved by the present code in the case of the Dirac-Breit Hamiltonian and non-orthogonal radial functions for one-electron spin-orbitals.

The evaluation of the pure angular momentum coefficients is a common problem in the atomic physics and a set of computational methods and programs were developed. In particular, for the *jj*-coupling, one of the most effective program is the ANCO (Gaigalas et al 2001, Gaigalas and Fritzsche 2002), based on techniques of second quantization, irreducible tensorial operators and quasispin. This program is designed for large-scale atomic structure calculations and its computational cost is relatively low. However, this program as well as other similar program are restricted to the orthogonal set of one-electron spin-orbitals. In this respect, the present DBSR_BREIT program may be considered as a complementary version of the ANCO program for the non-orthogonal case. It is intended to be used in atomic structure calculations, as well as the calculations of transition probabilities, the study of scattering processes or, in general, in all cases where the relaxation or term-dependence of atomic orbitals plays an important role and can be naturally included through the use of nonorthogonal orbitals.

Primarily, this concerns the study of transition probabilities, such as the shake-up probabilities or autoionization rates during the creation of inner-shell vacancies. In this case, the most efficient way is to determine the wave functions for initial and final states in separate, independent calculations, and then determine the relevant transition matrix elements. This naturally leads to the use of non-orthogonal orbitals. The non-orthogonality may be useful also for the calculation of the separate atomic states. The convergence of the many-configuration expansion may be much faster if the orbitals associated with different configurations and terms are

not necessarily required to be orthogonal. This allows us to include a large part of correlation in the first step of the determination of the basis CSFs and leads to much shorter expansions.

Another important application of the non-orthogonal technique may be expected in scattering calculations. In the close-coupling expansions for the continuum wave functions, it is commonly assumed that the radial functions for the scattered electron are orthogonal to the bound orbitals. The general formalism does not require orthogonality but it results in a considerable simplification of the calculations. In many cases, this don't lead to loss of generality, but results in an extremely large number of the so-called bound channels which are introduced to compensate the imposed orthogonality conditions on the continuum orbitals. This also can lead to the appearance of pseudo-resonances in the different cross-sections. The use of the non-orthogonality technique can overcome these problems and lead to the higher accuracy as was illustrated in many examples in the non-relativistic calculations with the BSR code (Zatsarinny 2006).

The present DBSR_BREIT program was designed as a part of the DBSR complex for treating the electron atom scattering in the B-spline R-matrix fully-relativistic approach based on the Dirac-Breit Hamiltonian. At the moment, the DBSR complex is in the process of preparation for submission to the CPC library. As a preliminary step towards this submission, the present DBSR_BREIT program is submitted separately as a code that can be used outside the DBSR complex for a variety of atomic applications. For this reason, the present submission also contain additional subroutines and utility programs which allow the user to employ the present technique in different tasks.

2. Outline of the method

The present procedure is based on the expansion of the N -electron configuration wave function in terms of Slater determinants

$$\Phi(\gamma JM) = \sum_{\{u\}} (\{u\} | \gamma JM) |u_1 \dots u_N\rangle \quad (1)$$

where $(\{u\} | \gamma JM)$ denotes the expansion coefficient and the individual determinant $|u_1 \dots u_N\rangle$ has the form

$$|u_1 \dots u_N\rangle = \frac{1}{N!} \begin{vmatrix} \langle x_1 | u_1 \rangle & \langle x_1 | u_2 \rangle & \dots & \langle x_1 | u_N \rangle \\ \langle x_2 | u_1 \rangle & \langle x_2 | u_2 \rangle & \dots & \langle x_2 | u_N \rangle \\ \dots & \dots & \dots & \dots \\ \langle x_N | u_1 \rangle & \langle x_N | u_2 \rangle & \dots & \langle x_N | u_N \rangle \end{vmatrix} \quad (2)$$

Here $x=(\mathbf{r},\sigma)$ represent the position and spin of the individual electron and u stands for the one-electron quantum numbers, $|u\rangle = |nljm\rangle$ or $|u\rangle = |n\kappa m\rangle$, where m is the z-component of the total angular momentum j and κ is the relativistic quantum number defined as

$$\kappa = (l - j)(2j + 1) \quad \text{with} \quad \kappa(\kappa + 1) = l(l + 1). \quad (3)$$

Each of one-electron spin-orbitals in relativistic theory is a Dirac four-component spinor

$$\phi_{n\kappa m}(\mathbf{r}) = \frac{1}{r} \begin{pmatrix} P_{n\kappa}(r) \eta_{\kappa m}(\hat{\mathbf{r}}, \sigma) \\ iQ_{n\kappa}(r) \eta_{-\kappa m}(\hat{\mathbf{r}}, \sigma) \end{pmatrix}, \quad (4)$$

where the real and imaginary parts are referred as the large and small components, respectively, and $\eta_{\kappa m}$ is the spinor spherical harmonic

$$\eta_{\kappa m}(\hat{\mathbf{r}}, \sigma) = \sum_{m_\ell m_s} (\ell m_\ell \frac{1}{2} m_s | jm) Y_{\ell m_\ell}(\theta, \phi) \chi_{\frac{1}{2} m_s}(\sigma), \quad (5)$$

where $(\ell m_\ell \frac{1}{2} m_s | jm)$ are Clebsch-Gordon coefficients, $Y_{\ell m_\ell}(\theta, \phi)$ are spherical harmonics and $\chi_{\frac{1}{2} m_s}(\sigma)$ are the usual two-component Pauli spin functions. The separation of variables in representation (4) allows the Hamiltonian matrix element to be split so that angular integrals are calculated using Racah algebra and radial integrals by numerical quadrature.

Having decomposition (1), the determination of the matrix elements between CSFs is reduced to that of the matrix elements between separate Slater determinants, which, in turn, are reduced to one- or two-electron integrals between spin-orbitals in the $nljm$ -representation. In comparison to the existing codes based on the Racah technique, the resulting formulae explicitly include the magnetic quantum numbers and contain additional summations over Slater determinants and over all electrons but not shells. This increases considerably the number of operations required, and the speed of calculations is much slower. On the other hand, the final formulae are much simpler and more convenient for programming. But the main advantage is that the $nljm$ -

representation procedure facilitates the direct and rather simple extension to the case of non-orthogonal orbitals to be made in the most general way.

The main problem in the above scheme is to obtain the determinant representation of arbitrary atomic wave function, including the configurations with several open shells. An effective way to solve this problem has been proposed in the program ZAP_NO (Zatsarinny 1996) in the non-relativistic case. It is based on the preliminary creation of tables of expansion coefficients for the individual shells. The determination of the coefficients for the CSFs is then reduced to the vector coupling of shell terms. This procedure, extended to jj-coupling, is used in the present code, and we refer the reader for details to the description of program ZAP_NO.

2.1. The Dirac-Coulomb Hamiltonian

The multiconfiguration self-consistent-field calculations in relativistic atomic theory are based on the Dirac–Coulomb Hamiltonian for electrons in a central field $V_{\text{nuc}}(r)$ made by an infinitely heavy nucleus of charge Z :

$$H_{DC} = \sum_i h_D(r_i) + \sum_{i<j} \frac{1}{r_{ij}}, \quad (3)$$

where h_D is the Dirac one-particle operator

$$h_D = c \boldsymbol{\alpha} \cdot \mathbf{p} + (\boldsymbol{\beta} - 1)c^2 + V_{\text{nuc}}(r) \quad (4)$$

and the second term in (3) represents the Coulomb interaction of pairs of electrons. In (2), $\mathbf{p} = -i\nabla$ is the momentum operator of electron, c is the speed of light, and the electron rest mass has been subtracted. The components of the vector $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the usual Dirac matrices constructed from Pauli spin and unit matrixes.

The matrix elements are best evaluated using the techniques of tensor operator algebra, which requires the terms of the Hamiltonian operator be transformed first to the irreducible tensorial form. For Dirac-Coulomb Hamiltonian it leads to

$$H_{DC} = \sum_i h_D(r_i) + \sum_{i<j} \sum_k \frac{r_{<}^k}{r_{>}^{k+1}} (\mathbf{C}^{(k)}(i) \bullet \mathbf{C}^{(k)}(j)), \quad (5)$$

where $r_{<}$ and $r_{>}$ are the lesser and the greater of r_i and r_j and the modified spherical harmonic tensors have components

$$C_q^{(k)}(\theta, \phi) = \left[\frac{4\pi}{(2k+1)} \right]^{1/2} Y_q^k(\theta, \phi). \quad (6)$$

2.2. General expression for matrix elements

Matrix elements for functions (1) are reduced to the matrix elements between Slater determinants. The structure of determinant matrix elements in the general case of non-orthogonal orbitals has been derived by

Löwdin (1955). Consider a matrix elements between two Slater determinants, $|u_1 \dots u_N\rangle$ and $|v_1 \dots v_N\rangle$ of a Hermitean operator

$$\Omega = \Omega^{(0)} + \sum_i \Omega^{(1)}(i) + \sum_{i < j} \Omega^{(2)}(i, j) \quad (7)$$

which is symmetrical in the indices of the particles. Define the determinant $D_{uv} = \det \{ \langle u_i | v_j \rangle \}$ of matrix formed from all overlap integrals between the one-electron orbitals u_i and v_j , ($i, j=1, \dots, N$), and let $D_{uv}(\rho; \rho')$ and $D_{uv}(\rho, \sigma; \rho', \sigma')$ denote the first and second cofactors of the determinant D_{uv} . The Löwdin (1955) showed that the matrix elements between two Slater determinants can be express through the matrix element between individual orbitals as

$$\begin{aligned} \langle u_1 u_2 \dots u_N | \Omega | v_1 v_2 \dots v_N \rangle &= \Omega^{(0)} D_{uv} + \sum_{\rho, \rho'} (-1)^{\rho+\rho'} \langle u_\rho | \Omega^{(1)} | v_{\rho'} \rangle D_{uv}(\rho; \rho') \\ &+ \sum_{\rho < \sigma, \rho' < \sigma'} (-1)^{\rho+\sigma+\rho'+\sigma'} (1 - P_{\rho'\sigma'}) \langle u_\rho, u_\sigma | \Omega^{(2)} | v_{\rho'}, v_{\sigma'} \rangle D_{uv}(\rho, \sigma; \rho', \sigma'). \end{aligned} \quad (8)$$

Here $P_{\rho'\sigma'}$ denotes the transposition of two-spin orbitals $nljm$ labelled ρ' and σ' . Pairs (ρ, σ) and (ρ', σ') have the same parity and the same sum of magnetic quantum numbers, $m_\rho + m_\sigma = m_{\rho'} + m_{\sigma'}$. The overlap factors $D_{uv}(\rho; \rho')$ and $D_{uv}(\rho, \sigma; \rho', \sigma')$, along with many more extended summations over the spin orbitals, present the only point of departure with the case of orthogonal orbitals where overlap factors equal to 1 or 0, but it makes the calculations much more cumbersome.

The above formulas, in connection with irreducible tensor algebra, allows us to reduce the configuration matrix elements to the one- or two-electron matrix elements between spin-orbitals $nljm$ multiplied by overlap factors. In case of the one-electron operator (4), the integration over angular variables is trivial and reduces to the δ -symbols for involved quantum numbers:

$$\langle u_\rho | h_D(r) | v_{\rho'} \rangle = \delta(m_\rho, m_{\rho'}) \delta(\kappa_\rho, \kappa_{\rho'}) I(\rho, \rho'), \quad (9)$$

where the one-electron radial integral is expressed through the large and small radial components $P(r)$ and $Q(r)$ as

$$I(\rho, \rho') = \int_0^\infty dr \left[c Q_\rho \left(\frac{d}{dr} + \frac{\kappa_\rho}{r} \right) P_{\rho'} + c P_\rho \left(-\frac{d}{dr} + \frac{\kappa_\rho}{r} \right) Q_{\rho'} - 2c^2 Q_\rho Q_{\rho'} + V_{\text{nuc}}(r) (P_\rho P_{\rho'} + Q_\rho Q_{\rho'}) \right] \quad (10)$$

To obtain the expression for Coulomb interaction, we first recall the definition of scalar product to obtain

$$\begin{aligned} \langle j_1 m_1 j_2 m_2 | \mathbf{C}^{(k)}(1) \bullet \mathbf{C}^{(k)}(2) | j_1' m_1' j_2' m_2' \rangle \\ = \sum_q (-1)^q \langle j_1 m_1 | C_q^{(k)}(1) | j_1' m_1' \rangle \langle j_2 m_2 | C_{-q}^{(k)}(2) | j_2' m_2' \rangle. \end{aligned} \quad (11)$$

From the Wigner-Eckart theorem we have

$$\langle jm | C_q^{(k)} | j' m' \rangle = (-1)^{j-m} \begin{pmatrix} j & k & j' \\ -m & q & m' \end{pmatrix} \langle j \| \mathbf{C}^{(k)} \| j' \rangle, \quad (12)$$

where the reduced matrix element $\langle j \| \mathbf{C}^{(k)} \| j' \rangle$ is found to be (Grant 2008)

$$\langle j \| \mathbf{C}^{(k)} \| j' \rangle = (-1)^{j+1/2} \sqrt{(2j+1)(2j'+1)} \begin{pmatrix} j & k & j' \\ \frac{1}{2} & 0 & -\frac{1}{2} \end{pmatrix}. \quad (13)$$

Combining the above formulas, the matrix element for Coulomb interaction between individual spin-orbitals are reduced to

$$\begin{aligned} \langle u_\rho, u_\sigma | \frac{1}{r_{ij}} | v_{\rho'}, v_{\sigma'} \rangle &= \sum_{kq} (-1)^{q+j_\rho-m_\rho+j_\sigma-m_\sigma} \begin{pmatrix} j_\rho & k & j_{\rho'} \\ -m_\rho & q & m_{\rho'} \end{pmatrix} \begin{pmatrix} j_\sigma & k & j_{\sigma'} \\ -m_\sigma & -q & m_{\sigma'} \end{pmatrix} \langle j_\rho \| \mathbf{C}^{(k)} \| j_{\rho'} \rangle \langle j_\sigma \| \mathbf{C}^{(k)} \| j_{\sigma'} \rangle \\ &\times \Pi^e(l_\rho, l_{\rho'}, k) \Pi^e(l_\sigma, l_{\sigma'}, k) R^k(\rho\sigma\rho'\sigma') \end{aligned} \quad (14)$$

where the even-parity coefficient

$$\Pi^e(l_\rho, l_{\rho'}, k) = \begin{cases} 1 & \text{if } l_\rho + l_{\rho'} + k \text{ even,} \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

reflects the selection rules for the Coulomb interaction. The relativistic Slater integral $R^k(abcd)$ are defined as

$$R^k(\rho\sigma\rho'\sigma') = \int_0^\infty \int_0^\infty [P_\rho(r_1)P_{\rho'}(r_1) + Q_\rho(r_1)Q_{\rho'}(r_1)] \frac{r_<^k}{r_>^{k+1}} [P_\sigma(r_2)P_{\sigma'}(r_2) + Q_\sigma(r_2)Q_{\sigma'}(r_2)] dr_1 dr_2 \quad (16)$$

2.3. The Breit interaction

The Breit interaction describes the relativistic corrections to the electronic motion due to magnetic and retardation effects (Grant 2007). An effective operator for the Breit interaction can be derived from quantum electro-dynamics (QED) in perturbation theory with respect to the number of virtually exchanged photons; in Coulomb gauge, for instance, the (transverse) Breit interaction is

$$b_{12} = \frac{\boldsymbol{\alpha}_1 \cdot \boldsymbol{\alpha}_2}{r_{12}} \cos(\omega r_{12}) + (\boldsymbol{\alpha}_1 \cdot \nabla_1)(\boldsymbol{\alpha}_2 \cdot \nabla_2) \frac{\cos(\omega r_{12}) - 1}{\omega^2 r_{12}} \quad (17)$$

where $\omega = |\varepsilon_1 - \varepsilon_1|/c$ describes the difference in the one-particle energies. From this expression, also, the *frequency-independent* (and original) Breit operator

$$b_{12}^0 = -\frac{1}{2r_{12}} \left[\boldsymbol{\alpha}_1 \cdot \boldsymbol{\alpha}_2 + \frac{(\boldsymbol{\alpha}_1 \cdot \mathbf{r}_{12})(\boldsymbol{\alpha}_2 \cdot \mathbf{r}_{12})}{r_{12}^2} \right] \quad (18)$$

is obtained in the long-wavelength approximation, $\omega \rightarrow 0$. This zero-frequency approximation to the full transverse interaction neglects all contribution $\sim \alpha^4 Z^3$ (and of higher order in αZ) but is well suited for most computations of many-electron atoms and ions since the explicit frequency-dependence of expression (17)

usually gives rise to only (very) small corrections. Below, we will therefore use the term *Breit interaction* for the frequency-independent expression (18) and we will treat it as an part of the electron–electron interaction

$$V_{12} = \frac{1}{r_{12}} + b_{12}^0. \quad (19)$$

In the DBSR programs, the inclusion of the Breit interaction is optional and due to the big computational efforts, this interaction is usually omitted in the set-up of the Hamiltonian matrix when its influence is expected to be small.

The correct form of the Breit interaction for use in multi-configuration calculations was discussed by Grant and Pyper (1976). Utilizing a representation of these operators in terms of spherical tensors, they provided the angular momentum reduction of relevant matrix elements to a sum of radial integrals. Final expression for the Breit (b_{12}^0) matrix elements can be expressed though the integrals

$$S^k(\rho\sigma\rho'\sigma') = \int_0^\infty \int_0^\infty P_\rho(r_1) Q_\sigma(r_1) \frac{r_1^k}{r_2^{k+1}} \varepsilon(r_2 - r_1) P_{\rho'}(r_2) Q_{\sigma'}(r_2) dr_1 dr_2 \quad (20)$$

where

$$\varepsilon(r_2 - r_1) = \begin{cases} 1 & \text{if } r_2 > r_1 \\ 0 & \text{if } r_1 > r_2 \end{cases}. \quad (21)$$

Then the matrix element for the Breit interaction in the uncoupled representation is given by

$$\begin{aligned} \langle u_\rho, u_\sigma | b_{12}^0 | v_{\rho'}, v_{\sigma'} \rangle &= \sum_{kq} (-1)^{q+j_\rho-m_\rho+j_\sigma-m_\sigma} \begin{pmatrix} j_\rho & k & j_{\rho'} \\ -m_\rho & q & m_{\rho'} \end{pmatrix} \begin{pmatrix} j_\sigma & k & j_{\sigma'} \\ -m_\sigma & q & m_{\sigma'} \end{pmatrix} \langle j_\rho \| \mathbf{C}^{(k)} \| j_{\rho'} \rangle \langle j_\sigma \| \mathbf{C}^{(k)} \| j_{\sigma'} \rangle \\ &\times \sum_{v=k-1}^{v=k+1} \Pi^o(l_\rho, l_{\rho'}, v) \Pi^o(l_\sigma, l_{\sigma'}, v) \sum_{\mu=1}^8 s_\mu^{vk}(\rho\sigma\rho'\sigma') S_\mu^v(\rho\sigma\rho'\sigma') \end{aligned}$$

where

$$\Pi^o(l, l', k) = 1 - \Pi^e(l, l', k), \quad (22)$$

and explicit formulae for the coefficients s_μ^{vk} and corresponding permutation of orbitals in the S_μ^v integrals are given in Table 1. Note the similar structure of matrix elements for Coulomb and Breit interactions, except the last terms given in the second line.

Table 1. The coefficients s_{μ}^{vk} .

μ	S_{μ}^v	$v = k - 1$	$v = k$	$v = k + 1$
1	$(\rho\sigma\rho'\sigma')$	$(k+K)(b'+c'K')$	$-(\kappa_a+\kappa_c)(\kappa_b+\kappa_d)/k/(k+1)$	$(b+cK)(K'-k-1)$
2	$(\sigma\rho\sigma'\rho')$	$(k+K')(b'+c'K)$	- / -	$(b+cK')(K-k-1)$
3	$(\rho'\sigma'\rho\sigma)$	$(k-K)(b'-c'K')$	- / -	$(b-cK)(-K'-k-1)$
4	$(\sigma'\rho'\sigma\rho)$	$(k-K')(b'-c'K)$	- / -	$(b-cK')(-K-k-1)$
5	$(\rho\sigma'\rho'\sigma)$	$-(k+K)(b'-c'K')$	- / -	$-(b+cK)(-K'-k-1)$
6	$(\sigma'\rho\sigma\rho')$	$-(k-K')(b'+c'K)$	- / -	$-(b-cK')(K-k-1)$
7	$(\rho'\sigma\rho\sigma')$	$-(k-K)(b'+c'K')$	- / -	$-(b-cK)(K'-k-1)$
8	$(\sigma\rho'\sigma'\rho)$	$-(k+K')(b'-c'K)$	- / -	$-(b+cK')(-K-k-1)$
$K=\kappa_c-\kappa_a$		$b=(v-1)/2/(2v+1)$		$c=(v+2)/2/v/(2v+1), v \geq 1$
$K'=\kappa_d-\kappa_b$		$b'=(v+2)/2/(2v+1)$		$c'=- (v-1)/(2v+1)/(2v+2), v \geq 0$

2.4. General remarks

Base on the above considerations Slater determinants, Eq.(7) and (8), the matrix element between two atomic states in the most general case of fully non-orthogonal orbitals has the form

$$\langle \Phi(\{nl\}\gamma J) | \mathbf{O} | \Phi'(\{n'l'\}\gamma' J') \rangle = \sum a_k O^k(i, j; i', j') \times D(\{nl\}; \{n'l'\}), \quad (23)$$

where a_k are the numerical coefficients, which depend only on the angular symmetry of the configuration state functions involved, O^k is the corresponding radial integral dependent on the operator \mathbf{O} under consideration, and $D(\{nl\}, \{n'l'\})$ is the overlap factor, which depends only on the sets of radial orbitals used in the construction of the wavefunctions Φ and Φ' , respectively. Hereafter, the notation *angular symmetry* (AS) is used to denote the full set of angular momenta ($\{k\}\gamma J(J')$), which includes the momenta of one-electron orbitals $\{k\}$, the terms of subshells $\alpha_i J_i$, and the intermediate terms J'_i as well as the total momenta J . The symbol γ defines the coupling of momenta of all occupied subshells according to the given scheme

$$((((core, J_1)J'_1, J_2)J'_2, J_3)J'_3 \dots)J. \quad (24)$$

It should be emphasized that the coefficients a_k do not depend directly on the set of the principal quantum numbers $\{n\}$ and $\{n'\}$. The overlap factor $D(\{nk\}, \{n'k'\})$ is generally a product of determinants corresponding to the matrices of one-electron overlap integrals $\langle nk|n'k' \rangle$; it depends only on the orthogonality conditions imposed on the one-electron radial functions. For example, in the special case of orthogonal radial orbitals, the overlap factors are trivially reduced to 1 or 0, and we obtain the weighted sum of radial integrals R^k , which is ordinarily

used in atomic structure calculations. For partial orthogonality conditions, some D -factors disappear while others are simplified. Thus, once we obtain the matrix element in the case of fully non-orthogonal orbitals, we can also obtain the matrix elements for all other configurations with the same AS and for all other orthogonal conditions, only by analyzing the overlap factors $D(\{nk\},\{n'k'\})$ and replacing the set $\{k,k'\}$ by the new one. Such a scheme is realized in the present code. In large-scale calculations, it can lead to a considerable reduction of the execution time, because in practical calculations the set of configurations contains many configurations with the same angular symmetry. Note that for scalar operators, the matrix elements are independent of the total magnetic quantum number M , and it is most effective to choose $M=J$, a choice that results in the minimum determinant expansions.

On the other hand, rather than record the angular coefficients a_k for a given set of CSFs, we can save only the expansions for the matrix between ASs with full non-orthogonality conditions. This data may be considered as a *databank* for angular coefficients a_k . They allow us to obtain the expansions of matrix elements for any set of CSFs constructed with the ASs included in the databank. In case of new symmetries, we have to perform only the additional calculations and thus the databank can be easily extended.

The next idea concerns the structure of the angular coefficients a_k themselves. Because the calculations of angular integrals follow the method based upon the representation of configuration wavefunctions through Slater determinants, the coefficients a_k can be separated into a factor that depends only on the Slater determinants involved and a factor dependent on the coupling scheme (24). Since many atomic states contain the same Slater determinants, it seems to be much more efficient to first calculate the matrix elements between separate Slater determinants before multiplying them with the coupling-scheme factor depending on the AS under consideration. The Slater determinants involved are defined only by the sets $\{\kappa^q\}$ where q is the shell occupation number. Hereafter, we will refer to the set $\{\kappa^q\}$ as a *configuration symmetry* (CS). Hence, in order to avoid the recalculation of matrix elements between the same Slater determinants, it is most effective to first define the list of all possible Slater determinants and to obtain the angular coefficients for the matrix elements between all involved Slater determinants.

The above ideas have been implemented in the present code. It requires additional sorting of the input configuration list according to their ASs as well as to CSs. The principal features of the present version are given below.

3. Structure and data flow

The block diagram of the program DBSR_BREIT, along with the data flow, is shown in Fig. 1. The only input data file is a *c*-file with a list of the configurations under consideration. The format of *c*-file is the same as in the GRASP complex (see also DBSR_CONF). The default name for this file is **cfg.inp**, and can be changed by the user to any **name.c** file where name is given as argument in the command line. The DBSR_BREIT may work as stand-alone program, however, in the DBSR calculations the main task is to consider different partial waves. In this case, the input *c*-file is supposed to be **cfg.nnn** and output bank has name **int_bnk.nnn** for each partial wave **nnn**. DBSR_BREIT requires only a limited number of other input parameters, which can be defined in the command line in the format '*parameter=value*'. All input parameters are read from the command line by the routine **read_arg**.

Then the program executes fully independent calculations for each partial wave (or for the specific case indicated by name). First, the **read_conf_jj** routine reads the input configurations in spectroscopic notation, decodes them to the internal 'integer' representation and tests the CSFs as to the correctness of angular coupling, the number of electrons, parity and AFTER conditions; **read_conf_jj** also allocates (or reallocates) all the main arrays with dimensions specific for a given case. Then the CSFs are sorted according to their angular and configuration symmetries. At this stage the program also checks if it is a continued calculation, i.e., if there exists a relevant databank from previous calculations. The program analyses information contained in the databank and defines the subset of configurations and operators that need additional consideration, or the information in the databank is sufficient for the given input configuration list.

In the next block, **pre_det_exp**, the determinant expansions for all configurations are generated and recorded in a scratch file. It allows one to avoid a large amount of recalculations for repeating determinant coefficients. The determinant expansion for a given angular symmetry is generated in the routine **det_expn_jj**, on the basis of angular coupling coefficients, calculated in **detc_jq**, and coefficients for determinant expansions of individual shells, stored in the module **mod_det_jq**. The determinant expansion were determined according to the same technique as it was done in program ZAP_NO (Zatsarinny 1996) in the LS-coupling case.

The calculations of a full set of angular coefficients and overlap factors for all input angular symmetries are running by subroutine **conf_loop**. First, routine **det_me** computes the angular coefficients for matrix elements between all Slater determinants involved. This information is accumulated in the module **zoef_list**. The cases of overlaps, one-electron and two-electron operators are treated separately by the routines **zno_0ee**, **zno_1ee**, and **zno_2ee**, respectively. Routine **me_jj** contains expressions for the Coulomb and Breit two-electron matrix elements in the *nljm*-representation, discussed in the sections 2.2 and 2.3. The corresponding angular coefficients for the integrals involved are accumulated in the module **boef_list**, in order to avoid their recalculation for matrix elements with the same configuration symmetries, but different angular symmetries. It is controlled by the routine **check_boef**.

Then, in routine **term_loop**, the angular coefficients between two given angular symmetries are calculated based upon the matrix elements between the involved determinants, stored in module **zcoef_list**, and the determinant expansion coefficients, which are read from scratch file **nud**. These data are accumulated in the module **coef_list**. When all determinants for the given angular symmetries are exhausted, the final angular coefficients are recorded by the routine **add_res** in the scratch file **nui**, and module **coef_list** is initialized for accumulation of data from other matrix elements. The above calculations are repeated in the block **conf_loop** for all combinations of input angular symmetries.

Finally, the new data and data from previous calculations are stored in file **int_res**. This file is then renamed to the initial name **int_bnk**. Such a procedure allows us to prevent losing the data in the initial databank if an error occurs when running DBSR_BREIT. The format of the output databank is given in section 3.5. The output databank has default name **int_bnk** for default input *c*-file **cfg.inp**. For specific input *c*-file **name.c**, the output databank has name **name.bnk**.

All calculations in DBSR_BREIT are carried out in full non-orthogonal mode, without specifications of principal quantum numbers for the orbitals involved. Instead, the positions of the corresponding shells in the configurations are used to specify the one-electron orbitals involved. Consequently, the data obtained can be used further for all atomic states with the given angular symmetries. The angular coefficients and overlap factors for specific orbitals are supposed to be calculated dynamically at the stage when the Hamiltonian matrix is generated (in the DBSR complex, it is realized in the program DBSR_MAT and DBSR_CI).

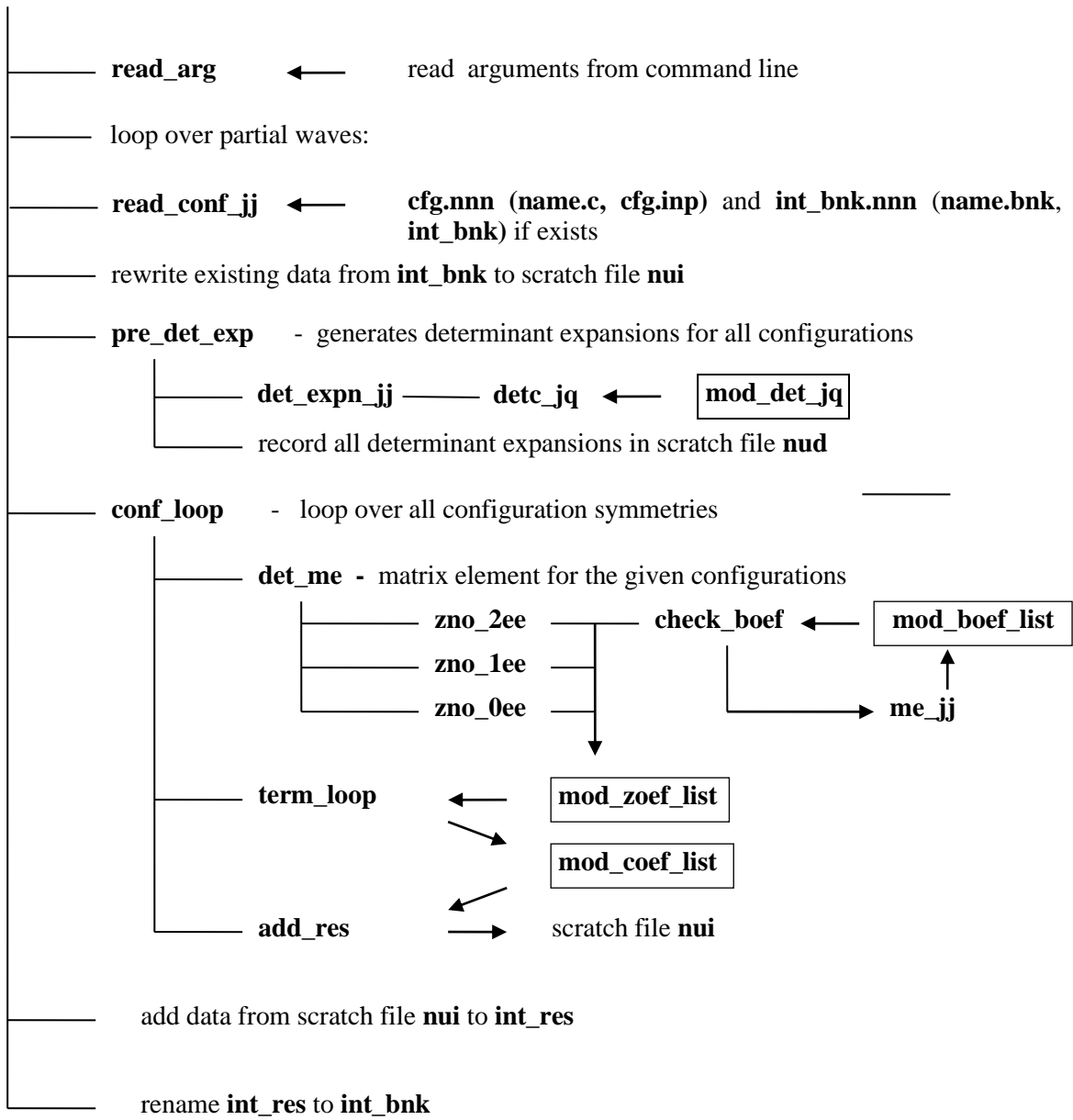
DBSR_BREIT

Fig. 1. Block diagram for the program DBSR_BREIT and data flow (see text).

3.1 Data files

cfg.nnn or name.c	File type: formatted sequential input. Created by program DBSR_CONF or written by user. Read by programs DBSR_BREIT, DSR_MAT, DBSR_HD. Description: contains configuration expansion for partial wave nnn or other configuration expansions.
int_bnk.nnn	File type: unformatted sequential output. Written by program DBSR_BREIT. Read by DBRS_MAT. Description: databank for angular coefficients.
dbsr_breit.log	File type: formatted sequential output. Written by program DBSR_BREIT. Description: running information.

3.2 Input parameters

Input parameters can be provided only in the command line in the format '*parameter=value*' (except name of the case, which if any appear as simple one-word argument) . All data have the default values indicated in the brackets.

name (name.c)	name of input c-file is supposed to be name.c ; resulting angular coefficients will be placed in name.bnk
klsp [0]	Index of partial wave under consideration; if $\neq 0$, input c-file is supposed to be as cfg.nnn and results are recorded in int_bnk.nnn
klsp1, klsp2 [0]	interval of partial waves under consideration (klsp=klsp1,klsp2)
mbreit [0]	if =1, the Breit interaction will be included
mk [9]	maximum multipole index
eps_c [1.d-7]	tolerance for coefficients; the coefficients which are less than eps_c are ignored

3.3. Structure of the angular-coefficients databank

The final expression of matrix elements between two angular symmetries has the form

$$\begin{aligned} \langle \Phi(\gamma J) | H | \Phi(\gamma' J) \rangle = & \sum_{a,b} t(ab) I(a,b) \times D^{ab}(\{n\kappa\}; \{n'\kappa'\}) \\ & + \sum_k \sum_{abcd} f^k(abcd) X^k(abcd) \times D^{abcd}(\{n\kappa\}; \{n'\kappa'\}) \end{aligned} \quad (25)$$

where we introduced the generalized two-body Slater integral

$$\begin{aligned} X^k(abcd) = & \Pi^e(l_\rho, l_{\rho'}, k) \Pi^e(l_\sigma, l_{\sigma'}, k) R^k(abcd) \\ & + \sum_{\nu=k-1}^{\nu=k+1} \Pi^o(l_\rho, l_{\rho'}, \nu) \Pi^o(l_\sigma, l_{\sigma'}, \nu) \sum_{\mu=1}^8 s_\mu^{\nu k}(abcd) S_\mu^\nu(abcd) \end{aligned} \quad (26)$$

We use an abbreviation of the quantum numbers $a = (n_a, \kappa_a)$, $b = (n_b, \kappa_b)$, ... in order to describe the different subshells of equivalent electrons (not the one-electron spin-orbitals). The n -values above are the position of the subshell in the given configuration symmetries (not the principal quantum number). The data bank contains the description of the configuration and angular symmetries involved into calculations; description of all possible determinant overlaps and overlap factors; indication of the proceeded matrix elements; finally, list of angular coefficients with integral and overlap pointers. The recorded angular coefficients are $t(ab)$ and $f^k(abcd)$, whereas $s_\mu(abcd)$ coefficients, along the parity restrictions Π^e and Π^o , are supposed to be included at the stage of setting up the Hamiltonian matrix.

Description of records in the data bank:

1. Block describing the configuration symmetries involved:

```
nsymc, lsymc
JT_conf(1:nsymc)
no_conf(1:nsymc)
ip_conf(1:nsymc)
iq_conf(1:lsymc)
kn_conf(1:lsymc)
```

nsymc - number of different configuration symmetries
lsymc - length of the involved arrays
JT_conf(i) - total angular momentum of the given configuration i
no_conf(i) - number of relativistic subshells
ip_conf(i) - pointer to the list of subshells in the iq_conf and kn_conf arrays, ip
iq_conf(ip+j) - occupation number for the subshell j in the configuration i
kn_conf(ip+j) - quantum number κ for the subshell j in the configuration i
(see also routines **read_symc** and **write_symc** in the module **mod_symc.f90**)

2. Block describing the angular symmetries involved:

```
nsymt, lsymt
```

```

it_conf(1:nsymt)
ip_term(1:nsymt)
JS_term(1:lsymt)
VS_term(1:lsymt)
JI_term(1:lsymt)

```

nsymt - number of angular symmetries

lsymt - length of the involved arrays

it_conf(i) - pointer to the configuration for the given angular symmetry i

ip_term(i) - pointer to the list of subshell terms, ip

JS_term(ip+j) - total angular momentum for the subshell j in the angular symmetry i

VS_term(ip+j) - seniority number for the subshell j in the angular symmetry i

JI_term(ip+j) - intermediate term for the given subshell

(see also routines **read_symt** and **write_symt** in the module **mod_symt.f90**)

3. List of overlap determinants involved:

```

ndet,ldet,jdet
for i=1,ndet:
kpd(i),ipd(i),jpd(i),NPD(ipd(i)+1:ipd(i)+kpd(i))

```

ndet - total number of overlap determinants.

ldet - length of the NPD array, containing the description of all overlap determinants.

jdet - average size of one overlap determinant.

kpd(i) - dimension of the i -th overlap determinant, kd.

ipd(i) - pointer to the i -th overlap determinant in array NPD, ip.

jpd(i) - ordering pointer, allowing quick searching the given overlap.

NPD(ip+1:ip+kd) - contains description of the i -th determinant as $NPD(.) = i_1*b_d + i_2$,

where $b_d = 2^{15}$ - packing basis for overlap determinants; i_1, i_2 - pointers to shells in the involved configurations.

(see also routines **read_det** and **write_det** in the module **mod_det.f90**)

4. List of overlap factors involved:

```

ndef,ldef,jdef
for i=1,ndef
kpf(i),ipf(i),jpf(i),NPF(ipf(i)+1:ipf(i)+kpf(i))

```

ndef - total number of different overlap factors.

ldef - length of the NPF array, containing the description of all overlap factors.

jdef - average size of one overlap factor.

kpf(i) - dimension of the i -th overlap factor, kd.

ipf(i) - pointer to the i -th overlap factor in array NPF, ip.

jpf(i) - ordering pointer, allowing quick searching the given overlap factor.

NPF(ip+1:ip+kd) - contains description of the i -th overlap factor as $NPF(.) = ipd*b_f + nd$,

where $b_f = 16$ - packing basis for overlap factors; ipd - pointer to the overlap determinant; nd - its power.

(see also routines **read_detf** and **write_detf** in the module **mod_det.f90**)

5. Pointer to the considered matrix element:

```
n
it_done(1:n)
```

n – number of recorded elements (= $nsymt*(nsymt+1)$)

$it_done(ij) > 0$ means that the matrix element between angular symmetries i and j is already in the data bank; $ij = i*(i-1)/2 + j$ with $i > j$; the inclusion or not of the Breit operator is not recorded – it is a responsibility of the user to remember this information.

(see also routines **read_done** and **write_done** in the module **mod_symt.f90**)

6. List of angular coefficients:

C, it, jt, int, idf – repeat up to the end of file

C – angular coefficient for matrix element between angular symmetries it and jt

int – pointer to the relevant radial integral in the packing form

$int = m \times b^8 + k \times b^4 + i_1 \times b^3 + i_2 \times b^2 + i_3 \times b + i_4$, where m – type of integral; k – multipole index; i_1, i_2, i_3, i_4 – pointer on the involved orbitals; $b = 10$ – packing basis.

idf – pointer on the corresponding overlap factor.

In order to clarify the structure of the overlap determinants and overlap factors records above, recall the determinant factors $D_{uv} = \det \{ \langle u_i | v_j \rangle \}$ in Eq.(8) and Eq.(23) are the determinants of matrix formed from all overlap integrals between the one-electron orbitals u_i and v_j , as shown in the example below for $N=6$:

$$\begin{vmatrix} \langle u_1 | v_1 \rangle & \langle u_1 | v_2 \rangle & \langle u_1 | v_3 \rangle & \langle u_1 | v_4 \rangle & \langle u_1 | v_5 \rangle & \langle u_1 | v_6 \rangle \\ \langle u_2 | v_1 \rangle & \langle u_2 | v_2 \rangle & \langle u_2 | v_3 \rangle & \langle u_2 | v_4 \rangle & \langle u_2 | v_5 \rangle & \langle u_2 | v_6 \rangle \\ \langle u_3 | v_1 \rangle & \langle u_3 | v_2 \rangle & \langle u_3 | v_3 \rangle & \langle u_3 | v_4 \rangle & \langle u_3 | v_5 \rangle & \langle u_3 | v_6 \rangle \\ \langle u_4 | v_1 \rangle & \langle u_4 | v_2 \rangle & \langle u_4 | v_3 \rangle & \langle u_4 | v_4 \rangle & \langle u_4 | v_5 \rangle & \langle u_4 | v_6 \rangle \\ \langle u_5 | v_1 \rangle & \langle u_5 | v_2 \rangle & \langle u_5 | v_3 \rangle & \langle u_5 | v_4 \rangle & \langle u_5 | v_5 \rangle & \langle u_5 | v_6 \rangle \\ \langle u_6 | v_1 \rangle & \langle u_6 | v_2 \rangle & \langle u_6 | v_3 \rangle & \langle u_6 | v_4 \rangle & \langle u_6 | v_5 \rangle & \langle u_6 | v_6 \rangle \end{vmatrix}.$$

Integration over angular variables simplify the matrix, leaving only radial overlaps for orbitals with same κ -values (f are g notation is used for the corresponding radial functions) :

$$\begin{vmatrix} \langle f_1^{\kappa_1} | g_1^{\kappa_1} \rangle & \langle f_1^{\kappa_1} | g_2^{\kappa_1} \rangle & 0 & 0 & 0 & 0 \\ \langle f_2^{\kappa_1} | g_1^{\kappa_1} \rangle & \langle f_2^{\kappa_1} | g_2^{\kappa_1} \rangle & 0 & 0 & 0 & 0 \\ 0 & 0 & \langle f_3^{\kappa_2} | g_3^{\kappa_2} \rangle & 0 & 0 & 0 \\ 0 & 0 & 0 & \langle f_4^{\kappa_3} | g_4^{\kappa_3} \rangle & \langle f_4^{\kappa_3} | g_5^{\kappa_3} \rangle & \langle f_4^{\kappa_3} | g_6^{\kappa_3} \rangle \\ 0 & 0 & 0 & \langle f_5^{\kappa_3} | g_4^{\kappa_3} \rangle & \langle f_5^{\kappa_3} | g_5^{\kappa_3} \rangle & \langle f_5^{\kappa_3} | g_6^{\kappa_3} \rangle \\ 0 & 0 & 0 & \langle f_6^{\kappa_3} | g_4^{\kappa_3} \rangle & \langle f_6^{\kappa_3} | g_5^{\kappa_3} \rangle & \langle f_6^{\kappa_3} | g_6^{\kappa_3} \rangle \end{vmatrix}$$

The matrix has, in general, the Jordan form. Each submatrix leads to the specific overlap determinants, recorded in block 3 (they can be defined by the radial functions involved), whereas the combination of the overlap determinants is referred as overlap factor, block 4.

4. Related formulas

The DBSR program used the core notation defined as the closed shells which are common for all configurations involved and which are frozen during the calculations. The core orbitals are supposed to be orthogonal to all valence orbitals. The DBSR_BREIT program considers the core as "physical vacuum" and ignores its orbital during generations of the angular coefficients. The contribution of core to the total energy and the core-valence interactions can be incorporated additionally using simple analytic expressions. Below we proved the formulae which can be used in this respect. In order to keep the expressions in form applicable to both Coulomb and Breit interaction, let introduce the generalized two-body Slater integral

4.1 Configuration (term-average) energy.

The term-average energy of the given configuration with orthogonal one-electron orbitals can be expressed through one-electron integrals I and two-electron Slater integrals X^k as (Grant 2008)

$$E_{Av} = \sum_a q_a I(a, b) + \sum_a \frac{1}{2} q_a (q_a - 1) \left[X^0(aa, aa) - \frac{[j_a]}{[j_a] - 1} \sum_{k=1}^{2j_a} \begin{pmatrix} j_a & k & j_a \\ \frac{1}{2} & 0 & -\frac{1}{2} \end{pmatrix}^2 X^k(aa, aa) \right] \\ + \sum_{a, b > a} q_a q_b \left[X^0(ab, ab) - \sum_{k=|j_a - j_b|}^{j_a + j_b} \begin{pmatrix} j_a & k & j_b \\ \frac{1}{2} & 0 & -\frac{1}{2} \end{pmatrix}^2 X^k(ab, ba) \right] \quad (27)$$

where q_a, q_b are the occupations numbers for the subshells a and b . Note that the E_{av} expression given in the description of DBSR_HF (Zatsarinny and Froese Fischer 2016) contains the error, where the 3J-symbol is replaced by $C^{(k)}$ reduced matrix element. The right expression should be

$$E_{Av} = \sum_a q_a I(a, b) + \sum_a \frac{1}{2} q_a (q_a - 1) \left[X^0(aa, aa) - \frac{1}{[j_a]([j_a] - 1)} \sum_{k=1}^{2j_a} \langle j_a \| C^{(k)} \| j_a \rangle^2 X^k(aa, aa) \right] \\ + \sum_{a, b > a} q_a q_b \left[X^0(ab, ab) - \frac{1}{[j_a][j_b]} \sum_{k=|j_a - j_b|}^{j_a + j_b} \langle j_a \| C^{(k)} \| j_b \rangle^2 X^k(ab, ba) \right] \quad (28)$$

The core energy can be easily obtained from the above expressions substituting $q_a = [j_a]$ and $q_b = [j_b]$.

4.2 Core valence interaction

At generation of the Hamiltonian matrix, the core valence interaction can be included through the re-definition of the one-body $I(a, b)$ integrals. Note that these integrals are defined for $\kappa_a = \kappa_b$, what also means $j_a = j_b$ $l_a = l_b$. Based on the above expressions for the average energy, we get

$$I(a, b) \rightarrow I(a, b) + \sum_c \left[[j_c] X(ac, bc) - \frac{1}{[j_a]} \sum_{k=|j_a - j_c|}^{j_a + j_c} \langle j_a \| C^{(k)} \| j_c \rangle^2 X^k(ac, cb) \right] \quad (29)$$

where sum over c includes the summation over all core orbitals.

5. Additional options

5.1 Simplified treatment of matrix elements with orthogonal orbitals

The DBSR_BREIT has no option for the special case of orthogonal orbitals, which widely used in different applications. Instead, we provide is two additional subroutines, **coef_1conf_jj.f90** and **coef_2conf.f90**, which can be used for the calculations with orthogonal orbitals. These subroutines are the simplified versions of the DBSR_BREIT code and placed in the ZCONFJJ library. They can be used in different programs, in particular, they are used in the DBSR_HF program

The **coef_1conf_jj** subroutine treats the matrix elements of two-body interelectron interaction for one atomic configuration state, which can be represented from the direct and exchange integrals as

$$\left\langle \Phi(\gamma J) \left| \sum_{i < j} \frac{1}{r_{ij}} + b_{ij}^0 \right| \Phi(\gamma J) \right\rangle = \sum_{a \leq b} \sum_k f^k(ab) F^k(a, b) + \sum_{a < b} \sum_k g^k(ab) G^k(a, b) \quad (30)$$

where $F^k(a, b) = X^k(ab; ab)$ and $G^k(a, b) = X^k(ab; ba)$. Below is reproduced the beginning part of the subroutine with short descriptions of input/output parameters:

```
!=====
!      Subroutine coef_1conf(no,ln,jn,iq,Jshell,Vshell,Jintra,kmax,coefs)
!=====
!      computes the angular coefficients for 1 atomic state
!
!      Input:
!
!      no,ln,jn,iq,Jshell,Vshell,Jintra - description of the input state
!
!      no - number of subshells
!      ln(1:no) - l-values for subshell orbitals
!      jn(1:no) - 2j-values for subshell orbitals (integers)
!      iq(1:no) - occupation numbers
!      JSHELL(1:no) - 2J-values for subshell terms
!      VSHELL(1:no) - seniority numbers
!      Jintra(1:no) - 2J-values for intermediate terms
!                      Jintra(1) = JSHELL(1)
!                      Jintra(no) - total 2J-value of the state
!      kmax - max. multipole index (= max(jn))
!
!      Output:
!
!      coefs(1:no,1:no,0:kmax) - angular coefficients for
!
!      direct integrals:      coef(i,j,k) with i <= j
!      exchange integrals:    coef(i,j,k) with i > j
!-----
```

The **coef_2conf_jj** subroutine treats the matrix elements of two-body interelectron interaction between two atomic configuration states, which can be represented as sum of the Slater integrals as

$$\left\langle \Phi(\gamma J) \left| \sum_{i < j} \frac{1}{r_{ij}} + b_{ij}^0 \right| \Phi(\gamma J) \right\rangle = \sum_{a,b,a',b'} \sum_k f^k(ab; a'b') X^k(ab; a'b') \quad (31)$$

Below is reproduced the beginning part of the subroutine with short descriptions of input/output parameters:

```
!=====
      Subroutine coef_2conf_jj (no1,nn1,ln1,jn1,iq1,Jshell1,Vshell1,Jintra1,&
                             no2,nn2,ln2,jn2,iq2,Jshell2,Vshell2,Jintra2,&
                             mcoef,ncoef,icoefs,coefs)
!=====
! compute the angular coefficients between two atomic states
!-----
! Input:
!
! no1,ln1,jn1,iq1,Jshell1,Vshell1,Jintra1 - description of the first state
! no2,ln2,jn2,iq2,Jshell2,Vshell2,Jintra2 - description of the second state
!
! mcoef - max. number of angula coefficients
!
! Output:
!
! ncoef - number of angular coefficients
! icoefs(1:5,1:ncoef) - description of Slater integrals (k,a,b,a'.b')
! coefs(1:ncoef) - corresponding angular coefficients
!-----
```

5.2 The output of angular coefficients in the table form

In some cases we need the explicit tables for matrix element expansions, and the unformatted **int_bnk** file is not convenient in this respect. On the other hand, the full print-out of interaction matrix may require very large disk space and often contains excessive information. The present code contains the utility **coef_jj**, which allows us to obtain the tables for some *selected* matrix element expansions based on the existing **int_bnk** files. In general, the inclusion of non-orthogonal orbitals leads to a considerable increase in the size of expansions, especially for two-body operators. As a first approximation, it is therefore recommended to use the fully-orthogonal mode (JORT=0) for calculation of spin-other-orbit interaction even with non-orthogonal orbitals.

C O E F _ J J

Description:	output the angular coefficients in the table form
Input files:	c-file (cfg.inp , name.c) and data-bank (int_bnk , name.bnk)
Output files:	coef_jj.tab
Call as:	coef_jj name cfg=.. bnk=.. tab=.. jort=.. ic= .. jc=.. mbreit=..

All input arguments are optional (default values are given in the brackets):

name	name of case; if name is presented, input c-file is supposed to be name.c , data-bank is name.bnk , and output file is name.tab
cfg [cfg.inp]	name of specific input c-file
bnk [int_bnk]	name of specific data-bank
tab [coef_jj.tab]	name for output tables
ic [0]	index of LHS state in c-file [if =0, all LHS states]
ij [0]	index of RHS state in c-file [if =0, all RHS states]
jort [-1]	orbital orthogonality mode: = -1, full orthogonality = +1, partial orthogonality (according orbital set indexes) ' = 0, full non=orthogonality
mbreit [0]	if =1, the Breit interaction will be included

References

Gaigalas G, Fritzsche S and Grant I P 2001 *Comput. Phys. Commun.* **139** 263

Program to calculate pure angular momentum coefficients in jj-coupling (ANCO v.1)

Gaigalas G and Fritzsche S 2002 *Comput. Phys. Commun.* **148** 349

Pure spin-angular momentum coefficients for non-scalar one-particle operators in jj-coupling (ANCO v.2)

Grant I P 2007 *Relativistic Quantum Theory of Atoms and Molecules: Theory and Computation*

(New York: Springer)

Grant I P and Pyper N C 1976 *J. Phys. B* **9** 761

Breit interaction in multi-configuration relativistic atomic calculations

Löwdin P-O 1955 *Phys. Rev.* **97** 1474

Quantum theory of many-particle systems. I. Physical interpretations by means of density matrices, natural spin-orbitals, and convergence problems in the method of configurational interaction

Zatsarinny O 1996 *Comput. Phys. Commun.* **98** 235

A general program for computing matrix elements in atomic structure with nonorthogonal orbitals

BSR: B-spline atomic R-matrix codes

Zatsarinny O and Froese Fischer C 2016 *Comput. Phys. Commun.* **202** 287

DBSR_HF: A B-spline Dirac–Hartree–Fock program

Varshalovich D A, Moskalev A N and Khersonskii V K 1988 *Quantum Theory of Angular Momentum*

(Singapore: World Scientific)