7. Program DBSR_MAT

7.1. Outline of the DBSR_MAT calculation

This program sets up the Hamiltonian and overlap matrices in the *B*-spline representation in the inner region. In *B*-spline basis, it is convenient to re-write the expansion (2.8) for the R-matrix basis function in the form

$$\Psi^{J\pi}(\mathbf{X}_{N+1}) = \mathrm{A}\sum_{i=1}^{n_c} \overline{\Phi}_i^{J\pi}(\mathbf{X}_N;\hat{\mathbf{r}}_{N+1}\sigma_{N+1}) r_{N+1}^{-1} u_i(r_{N+1}) + \sum_{j=1}^{n_b} b_j \Theta_j^{J\pi}(\mathbf{X}_{N+1}), \tag{7.1}$$

where the radial channel function

$$u_i(r) = \begin{pmatrix} P_i(r) \\ Q_i(r) \end{pmatrix} \tag{7.2}$$

are directly expressed as combination of *B*-splines

$$P_i(r) = \sum_{m=1}^{n_p} p_m^i B_m^{k_p}(r), \qquad Q_i(r) = \sum_{m=1}^{n_q} q_m^i B_m^{k_q}(r). \tag{7.3}$$

In this notation, each basis function can be represented by vector of expansion coefficients as

$$\mathbf{c} = [\mathbf{a}_1,...,\mathbf{a}_{n_c},\mathbf{b}]^T \tag{7.4}$$

where each $\boldsymbol{a}_i$ is a column vector of *B*-spline coefficients for the given channel functions $\overline{\Phi}_i$

$$\mathbf{a}_i = [p_1^i, p_2^i,..., p_{n_p}^i, q_1^i, q_2^i,..., q_{n_p}^i]^T \tag{7.5}$$

and $\boldsymbol{b}$ is the column vector of correlation functions coefficients

$$\mathbf{b}_j = [b_1, b_2,..., b_{n_p}]^T . \tag{7.6}$$

The inner-region problem is then reduced to the to a generalized eigenvalue matrix problem

$$\mathbf{Hc} = E\,\mathbf{Sc} \tag{7.7}$$

where **H** is the Hamiltonian matrix in the elementary *B*-spline basis (for scattering calculation, this matrix is modified by inclusion of Bloch operator (2.12)). Schematically, the Hamiltonian matrix **H** has the form

$$\begin{pmatrix} H_{11} & H_{12} & \cdots & H_{1n_c} & h_{11} & \cdots & h_{1n_b} \\ H_{21} & H_{21} & \cdots & H_{2n_c} & h_{21} & \cdots & h_{2n_b} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\ H_{n_c1} & H_{n_c1} & \cdots & H_{1n_c} & h_{n_c1} & \cdots & h_{n_cn_b} \\ h_{11}^T & h_{21}^T & \cdots & h_{n_c1}^T & b_{11} & \cdots & b_{1n_b} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{1n_b}^T & h_{2n_b}^T & \cdots & h_{n_cn_b}^T & b_{n_bn_b} & \cdots & b_{n_bn_b} \end{pmatrix} \tag{7.6}$$

where $H_{ii'}$ are $n_s \times n_s$ matrices for channel-channel interaction ($n_s=n_p+n_q$), $\{b_{jj'}\}$ is an $n_b \times n_b$ matrix that comes from the bound-bound interaction, and h_{ij} is a vector (with length n_s) representing the interaction between the i-th channel and the bound states j. The sub-matrixes $H_{ii'}$ and h_{ij} can be further divided on the blocks related to the large and small components, for example,

$$H_{ij} \rightarrow \begin{pmatrix} H_{ij}^{pp} & H_{ij}^{pq} \\ H_{ij}^{qp} & H_{ij}^{qq} \end{pmatrix} \tag{7.7}$$

That lead to four times more consuming calculations in comparison to the non-relativistic case.

Matrix $\boldsymbol{S}$ in equation (7.5) is the overlap matrix and in general has the same structure as Hamiltonian matrix. In the simplest case of orthogonal orbitals, it reduces to a matrix consisting of diagonal blocks of the banded overlap matrix between individual B-splines,

$$S_{ij} \rightarrow \begin{pmatrix} \mathbf{B}^{pp} & 0 \\ 0 & \mathbf{B}^{qq} \end{pmatrix}, \qquad B_{ij} = < B_i \mid B_j > \tag{7.8}$$

Each individual matrix element in (7.6) is expressed by the program DBSR_BREIT in the form

$$\begin{aligned} \sum_{ab} t(ab) L(ab) \times D^{ab}(\{n\kappa\};\{n'\kappa'\}) + \sum_{abcd} \sum_{\lambda} r^{\lambda}(abcd) R^{\lambda}(abcd) \times D^{abcd}(\{n\kappa\};\{n'\kappa'\}) \\ + + \sum_{abcd} \sum_{\lambda} s^{\lambda}(abcd) S^{\lambda}(abcd) \times D^{abcd}(\{n\kappa\};\{n'\kappa'\}) \end{aligned} \tag{7.9}$$

where $t(ab)$, $r^{\lambda}(abcd)$, and $s^{\lambda}(abcd)$ are numeric coefficients which depend only on the angular symmetry of the involved CSFs and indices a,b,c,d refer to the corresponding one-electron orbitals. $L(ab)$ stands for one-electron integrals, coming from the Dirac Hamiltonian (2.2):

$$\begin{aligned} L(ab) = < a \parallel h_D \parallel b > = \delta_{k_a k_b} \int_0^{\infty} dr \Bigg[cQ_a \left(\frac{d}{dr} + \frac{k_a}{r} \right) P_b + cP_a \left(-\frac{d}{dr} + \frac{k_a}{r} \right) Q_b \\ - 2c^2 Q_a Q_b + V_{\text{nuc}}(r)(P_a P_b + Q_a Q_b) \Bigg] \end{aligned} . \tag{7.10}$$

$R^{\lambda}(abcd)$ are the Slater integrals resulting from the inter-electron Coulomb interactions

$$R^{\lambda}(abcd) = \int_0^{\infty}\int_0^{\infty}[P_a(r_1)P_c(r_1)+Q_a(r_1)Q_c(r_1)]\frac{r_<^{\lambda}}{r_>^{\lambda+1}}[P_b(r_2)P_d(r_2)+Q_b(r_2)Q_d(r_2)]\mathrm{d}r_1\mathrm{d}r_2 ,$$
$$= R^{\lambda}(P_a,P_b;P_c,P_d)+R^{\lambda}(Q_a,Q_b;Q_c,Q_d)+R^{\lambda}(P_a,Q_b;P_c,Q_d)+R^{\lambda}(Q_a,P_b;Q_c,P_d) \qquad (7.11)$$

and $S^{\lambda}(abcd)$ reflect the Breit interaction (2.37)

$$S^{\lambda}(abcd) = \int_0^{\infty}\int_0^{\infty} P_a(r_1)Q_c(r_1)\frac{r_1^{\lambda}}{r_2^{\lambda+1}}\varepsilon(r_2-r_1)P_b(r_2)Q_d(r_2)\mathrm{d}r_1\mathrm{d}r_2 , \qquad (7.12)$$

where

$$\varepsilon(r_2-r_1) = \begin{cases} 1 & \text{if} \quad r_2 > r_1 \\ 0 & \text{if} \quad r_1 > r_2 \end{cases} . \qquad (7.13)$$

Each term in the (7.9) has corresponding overlap factor $D(\{nl\},\{n'l'\})$, which depends only on the orthogonality of the radial orbitals used in the construction of involved CSFs. In general, the overlap factor is the multiplier of determinants of matrices consisting of one-electron overlaps

$$\begin{vmatrix} <n_1\kappa\,|\,n'_1\,\kappa> & <n_1\kappa\,|\,n'_2\,\kappa> & \dots & <n_1\kappa\,|\,n'_q\,\kappa> \\ <n_2\kappa\,|\,n'_1\,\kappa> & <n_2\kappa\,|\,n'_2\,\kappa> & \dots & <n_2\kappa\,|\,n'_q\,\kappa> \\ \dots & \dots & \dots & \dots \\ <n_q\kappa\,|\,n'_1\,\kappa> & <n_q\kappa\,|\,n'_2\,\kappa> & \dots & <n_q\kappa\,|\,n'_q\,\kappa> \end{vmatrix} \qquad (7.14)$$

The scattering orbitals can appear in the radial integrals L, R^{λ}, or S^{λ}, and also in the determinant factors. To derive the final expression for the matrix elements in the B-spline basis, let us first simplify the overlap factors (7.10) by expanding them over those row (or column) that contain one-electron overlaps with the scattering orbitals. At most, there can be one such row or/and column. The residual overlap determinants depend only on the known bound orbitals and they can be calculated in a standard manner. It is convenient to redefine the angular coefficients c_i. multiplying them by these residual overlap factors. As a result, the initial overlap factor $D(\{n\kappa\},\{n'k'\})$ in expression (7.7) is reduced to one- or two-electron overlaps of the form

$$\langle k\kappa\,|\,n\kappa\rangle, \quad \langle k\kappa\,|\,k'\kappa\rangle \quad \text{or} \quad \langle k\kappa\,|\,n\kappa\rangle\,\langle n'\kappa'\,|\,k'\kappa'\rangle , \qquad (7.15)$$

where $|k\kappa>$ stands for the radial function of the continuum orbital in the inner region.

For more detailed description of the Hamiltonian matrix **H**, let us introduce the integrals between individual B-splines. Here we should take into account that we use two different B-spline bases for the large and small components. For one-electron integrals we need matrixes

$$\mathbf{D}_{\kappa}^{\pm} \rightarrow \left\langle B_i \left| \pm\frac{d}{dr}+\frac{\kappa}{r}\right| B_j \right\rangle , \qquad (7.16)$$

$$\mathbf{V}^{nucl} \rightarrow \left\langle B_i \left| V_{nucl}(r) \right| B_j \right\rangle, \tag{7.17}$$

$$L(\bullet\bullet) = \begin{pmatrix} \mathbf{V}_{pp}^{nucl} & c\mathbf{D}_{pq}^{-} \\ c\mathbf{D}_{qp}^{+} & \mathbf{V}_{qq}^{nucl} - 2c^2 B_{qq} \end{pmatrix}. \tag{7.18}$$

To describe the Slater integrals, as seen from (7.11), we need the following four-dimensional arrays

$$R^{\lambda,pppp}(ij;i'j') = \int_0^\infty \int_0^\infty B_i^p(r_1) B_j^p(r_2) \frac{r_<^\lambda}{r_>^{\lambda+1}} B_{i'}^p(r_1) B_{j'}^p(r_2) \mathrm{d}r_1 \mathrm{d}r_2, \tag{7.19a}$$

$$R^{\lambda,qqqq}(ij;i'j') = \int_0^\infty \int_0^\infty B_i^q(r_1) B_j^q(r_2) \frac{r_<^\lambda}{r_>^{\lambda+1}} B_{i'}^q(r_1) B_{j'}^q(r_2) \mathrm{d}r_1 \mathrm{d}r_2, \tag{7.19b}$$

$$R^{\lambda,pqpq}(ij;i'j') = \int_0^\infty \int_0^\infty B_i^p(r_1) B_j^q(r_2) \frac{r_<^\lambda}{r_>^{\lambda+1}} B_{i'}^p(r_1) B_{j'}^q(r_2) \mathrm{d}r_1 \mathrm{d}r_2, \tag{7.19c}$$

$$R^{\lambda,qpqp}(ij;i'j') = \int_0^\infty \int_0^\infty B_i^q(r_1) B_j^p(r_2) \frac{r_<^\lambda}{r_>^{\lambda+1}} B^q(r_1) B_{j'}^p(r_2) \mathrm{d}r_1 \mathrm{d}r_2, \tag{7.19d}$$

The four-dimensional array, $R^\lambda(ij;i'j')$, effectively represents the Slater integral in the *B*-spline basis. The integrals are zero if either $|i-i'| \geq k_s$ or $|j-j'| \geq k_s$. Methods for computing these quantities, which depend only on the basis, are given in Section 12. Each above integral gives rise to a matrix in the H_{ij} blocks in equations (7.6) and (7.7), which describe the channel-channel interaction. To elucidate the structure of these matrices, let us introduce $n_{p,q} \times n_{p,q}$ matrices with elements

$$R^\lambda(\bullet\, a\, \bullet\, b)_{ij} = \sum_{jj'} a_i b_{j'} R^\lambda(ij;i'j') \tag{7.20a}$$

$$R^\lambda(\bullet\, a\, b\, \bullet)_{ij'} = \sum_{i'j} a_{i'} b_{j'} R^\lambda(ij;i'j') \tag{7.20b}$$

where a_i, b_j are the expansion coefficients in the spline basis for the known bound orbitals, similar to the expansions (7.3) for large or small components, depending on the type of integral under consideration. Similar matrixes can be also defined for the $S^\lambda(ij;i'j')$ integrals.

In order to describe the channel-bound interaction, let us introduce the following vectors with elements

$$B(\bullet\, a)_i = \sum_{j=1}^{n_s} a_j B_{ij} \tag{7.21}$$

$$L(\bullet\, a)_i = \sum_{j=1}^{n_s} a_j L_{ij} \tag{7.22}$$

$$R^{\lambda}(\,.\,a\,b\,c)_i = \sum_{ji'j'=1}^{n_s} a_j b_{i'} c_{j'} R^{\lambda}(ij;i'\,j') \tag{7.23}$$

With this notation, table 7.1 represents the contribution of the different terms from the energy expression (7.7), which is usually produced by the angular integration codes, into the interaction matrix (7.5), along with an indication of its structure. For brevity, Table 7.1 only present the structures coming from R^{λ} and L integrals, the contributions from the S^{λ} are similar. Table 7.1 also indicates the contribution to the total overlap matrix arising from the matrix element on the right hand side of equation (7.1). Note that the target wave functions are assumed to form an orthogonal set and diagonalize the target Hamiltonian. Hence, of all the matrix elements containing $\langle k\kappa|k'\kappa\rangle$, only the diagonal ones, with $k=k'$, contribute to the interaction matrix. The same concerns the matrix elements with $L(k\kappa,k'\kappa)$.

As clear from the above discussion, we are directly working not with total channel blocks, H_{ij} but with submatrices (7.7) connecting with the large or small components. That allows us to use the same subroutine as was developed for the non-relativistic case in the BSR complex (Zatsarinny 2006).

In comparison to the non-relativistic case, the Hamiltonian matric (7.6) is four time bigger and require much more time to be set up and much more space to be recorded in the file on the hard drive. The spectrum of this matrix contain both positive and negative ($< mc^2$) spectrum. For further scattering or bound-state calculations, we don't need the negative spectrum. To reduce the memory requirements and obtain the effective no-pair Hamiltonian, we use the following procedure based on the preliminary diagonalization of the diagonal blocks H_{ii}. Calculations are divided on to steps. First, we obtain the *B*-spline representation for each block H_{ii} and diagonalize them as

$$\mathbf{H}^{ii}\mathbf{a}^{i} = E^{i}\mathbf{S}^{ii}\mathbf{a}^{i} \tag{7.24}$$

In comparison to the non-relativistic case, the Hamiltonian matric (7.6) is four time bigger and require much more time to be set up and much more space to be recorded in the file on the hard drive. The spectrum of this matrix contain both positive and negative ($< mc^2$) spectrum. For further scattering or bound-state calculations, we don't need the negative spectrum. To reduce the memory requirements and obtain the effective no-pair Hamiltonian, we use the following procedure based on the preliminary diagonalization of the diagonal blocks H_{ii}. Calculations are divided on to steps. First, we obtain the *B*-spline representation for each block H_{ii} and diagonalize them as

$$\mathbf{H}^{ii}\mathbf{a}^{i} = E^{i}\mathbf{S}^{ii}\mathbf{a}^{i} \tag{7.25}$$

The set of eigenvalue solution $\{\mathbf{a}^{i}\}$ can be considered as new basis which is equivalent to the primary *B*-spline basis. However, now we can chose any subset of these solutions for further consideration. We remove all negative spectrum, thereby transforming the problem to the *no-pair approximation*. We also may remove the channel solutions with extremely large energies that will not influence the completeness. Using such basis we

transform the matrix **H** (7.6) to the new representations and recording it in the file for further diagonalization by program DBSR_HD. The same concerns the total overlap matrix.

Table 7.1. Contribution of different integrals (without indication of angular coefficients) in the interaction matrix (7.6) and (7.7). The symbols a,b,c,d stand for the bound orbitals, while k_i indicates the continuum orbital in channel i.

Term	Contribution	Remarks
Channel-channel interaction		
$R^\lambda(a,k_i;b,k_j)$	$R^\lambda(\,a\,\bullet\,b\,\bullet)$	banded matrix,
$R^\lambda(k_i,a;k_j,b)$	$R^\lambda(\,\bullet\,a\,\bullet\,b)$	direct interaction
$R^\lambda(k_i,a;b,k_j)$	$R^\lambda(\,\bullet\,a\,b\,\bullet)$	full matrix,
$R^\lambda(a,k_i;;k_j,b)$	$R^\lambda(\,a\,\bullet\,\bullet\,b)$	exchange interaction
$R^\lambda(k_i,a;b,c)\ \langle d \mid k_j\rangle$	$R^\lambda(\,\bullet\,a\,b\,c)\bullet B(\,\bullet\,d)$	full matrix,
$R^\lambda(a,k_i;b,c)\ \langle d \mid k_j\rangle$	$R^\lambda(\,a\,\bullet\,b\,c)\bullet B(\,\bullet\,d)$	contribution due to non-orthogonality
$R^\lambda(a,b;k_i,c)\ \langle d \mid k_j\rangle$	$R^\lambda(\,a\,b\,\bullet\,c)\bullet B(\,\bullet\,d)$	
$R^\lambda(a,b;c,k_i)\ \langle d \mid k_j\rangle$	$R^\lambda(\,a\,b\,c\,\bullet)\bullet B(\,\bullet\,d)$	
$\langle k_i \mid a\rangle\,\langle b \mid k_j\rangle\,R^\lambda$(or L)	$B(\,\bullet\,a)\bullet B(\,\bullet\,b)\ *\ R^\lambda$ (or L)	full matrix, contribution due to non-orthogonality
$\langle k_i \mid k_j\rangle\,R^\lambda$ (or L)	$B(\,\bullet\,\bullet)\ *\ E_i$	banded matrix, only for $i = j$
$L(k_i, k_j)$	$L(\,\bullet\,\bullet)$	banded matrix, only for $i = j$
$L(k_i,a)\,\langle b \mid k_j\rangle$	$L(\,\bullet\,a\,)\bullet B(\,\bullet\,b)$	full matrix, contribution due to non-orthogonality
$\langle k_i \mid\ k_j\rangle$	$B(\,\bullet\,\bullet)$	banded matrix, only for $i = j$, contribution to overlap matrix
$\langle k_i \mid a\rangle\,\langle b \mid k_j\rangle$	$B(\,\bullet\,a)\bullet B(\,\bullet\,b)$	full matrix, contribution to overlap matrix due to non-orthogonality
Channel-bound interaction		
$R^\lambda(k_i,a;b,c)$	$R^\lambda(\,\bullet\,a\,b\,c)$	vector
$R^\lambda(a,k_i;b,c)$	$R^\lambda(\,a\,\bullet\,b\,c)$	
$R^\lambda(a,b;k_i,c)$	$R^\lambda(\,a\,b\,\bullet\,c)$	
$R^\lambda(a,b;c,k_i)$	$R^\lambda(\,a\,b\,c\,\bullet)$	
$L(k_i,a)$	$L(\,\bullet\,a\,)$	vector
$\langle k_i \mid a\rangle\,R^\lambda$(or L)	$B(\,\bullet\,a\,)\ *\ R^\lambda$ (or L)	vector, contribution due to non-orthogonality
$\langle k_i \mid a\rangle$	$B(\,\bullet\,a\,)$	vector, contribution to overlap matrix
Bound-bound interaction		
$R^\lambda(\,a, b;\, c, d)$		scalar
$L(\,a\,, b)$		scalar
c		scalar, contribution to overlap matrix

7.2. Structure and data flow.

The block diagram of the program DBSR_MAT, along with the data flow, is shown in Fig.7.1. In the beginning, the program reads the data common to all partial waves. It includes the parameters for the calculation (routine **read_arg**), the *B*-spline parameters (**read_knot_dat**), and the description of the target states **(read_target_jj**). Then the program executes fully independent calculations for each partial wave, subroutine **SUB1**. First, the program reads the configuration list from the relevant *c*-file **cfg.nnn** and allocates all main arrays with dimensions specific for a given partial wave. The most critical dimension here is the size of the interaction matrix, which usually occupies most of the memory. The size of the interaction matrix is mainly defined by the product of the number of scattering channels and the number of *B*-splines, whereas the (*N*+1)-electron bound channels in the close-coupling expansion are usually either absent or their number is relatively small in the present approach. In order to speed up the calculation, we place the entire interaction matrix in the RAM memory. On the other side, this limits the number of scattering channels which we can include in the physical model by the size of the computer memory. In the parallelized version, the interaction matrix distributed over different cores.

The calculations of the matrix elements are performed for each type of integrals sequentially. This part is controlled by the routine **state_res**, whose flow diagram is placed at the bottom of Fig.7.1. Each time this routine reads the relevant **int_bnk** file from the beginning, chooses the integrals under consideration and provides their contribution to the total interaction matrix. Such a procedure reduces the number of integrals which should be considered at one time, and this increases the effectiveness of their processing. The standard procedure of preliminary calculations and storing of all two-electron integrals requires four-dimensional arrays. This cannot be realized in the case of a large number of different one-electron orbitals, which often occur in the present approach. Hence, the two-electron integrals are calculated dynamically when they are required. Note that in contrast to the standard *R*-matrix approach, the present implementation uses a larger number of non-orthogonal orbitals for a more accurate representation of the target states. Typical calculations may involve up to several hundred of different one-electron radial functions. The present code has no practical limit on the maximum number of different one-electron orbitals that may be used to describe the target and the scattering states.

The list of specific integrals needed for the given configurations is generated from the information in the databanks **int_bnk.nnn** for each partial wave. The **state_res** routine includes the checking of the determinant factor $D(\{n\kappa\},\{n'\kappa'\})$ for the specific orbitals (routine **check_det**), and the determination of the structure of integrals according to the Table 7.1, routine **idef_type**. The integrals are then stored in the module **c_data** in different ordered blocks, one block for the given type and multipole index, using routines **add_coef** and **add_cdata**. The number and size of the individual blocks in this module is defined by the input parameters and could be adjusted to the speed and memory of given computer. When all blocks are full, they are merged, if necessary, with the routine **merge_cdata** and then are processed by the routine **gen_matrix** to get the *B*-spline

representation for specific integrals (using routines **O_data**, **L_data**, **R_data** and **S_data,** each for one specific type of integrals). All these routines extensively use subroutines from the DBS library routines. The corresponding contributions are then added to the interaction matrix, located in module **dbsr_matrix**.

The above procedure (driving by routine **SUB1**) is running twice, first for the diagonal channel blocks H_{ii} (7.6) and overlap matrix **S** (7.5), then for all other Hamiltonian blocks. In the first step, the program also diagonalizes each channel block to generate new no-pair basis (routine **diag_channels**). This basis then used in routine **check_mat** to check big overlap matrix elements. If they exceed the given tolerance, additional orthogonal conditions assigned and the procedure is repeating. Then the total overlap matrix is transformed to the new basis (routine **transform_matrix**) and all relevant information is recorded to the file **dbar_mat.nnn** for given partial wave **nnn**. The next step include the calculation of the non-diagonal Hamiltonian blocks H_{ij}, their transformation to the new basis and recording to the disk.

The possible additional step is the correction of the interaction matrix according to the orthogonal conditions imposed on the scattering orbitals, using the routine **dbs_orth**. This is performed according to the projection procedure suggested by Bentley (1994). Suppose a continuum orbital in some *i*-th channel is orthogonal to the bound orbital $u_{njl}(r)$. This can be accounted for by modifying the Hamiltonian according to

$$\boldsymbol{H} \rightarrow (1 - \boldsymbol{B}\boldsymbol{c}\boldsymbol{c}^T)\boldsymbol{H}(1 - \boldsymbol{c}\boldsymbol{c}^T\boldsymbol{B}) \qquad (7.26)$$

where $\boldsymbol{c}$ is the full solution vector (7.2) with all zero elements, except that the channel *i* is replaced by the *B*-spline expansion $\{p_i,q_i\}$ of the orbital $u_{n\kappa}(r)$. This leads to transformation of only those blocks H_{ij} in (7.5) which have one index equal to *i*. Recall that the orthogonal conditions to the common core orbitals are implied automatically in the present implementation. All other conditions should be specified through input data in the **dbsr_par** or **cfg.nnn** files.

The main mode for execution of DBSR_MAT supposes that the target states are orthogonal and diagonalize the target Hamiltonian. As an additional rigorous check of calculations, the program also calculates the target energies and other Hamiltonian and overlap matrix elements between target states. If non-diagonal elements are not zero, or target energies are different from the values indicated in the file **target_jj**, the program provides warning for possible inconsistency. The present version, however, also has the option when target wavefunction are not the eigenvalues of the target Hamiltonian (see input parameter **iitar** below). In this case, the program will additionally diagonalize the target Hamiltonian and will output resulting target energies in file **target_new.nnn**. This option is used mainly for the bound-structure calculations. In this case, the target wavefunctions don't represent real atomic states but some convenient basis.

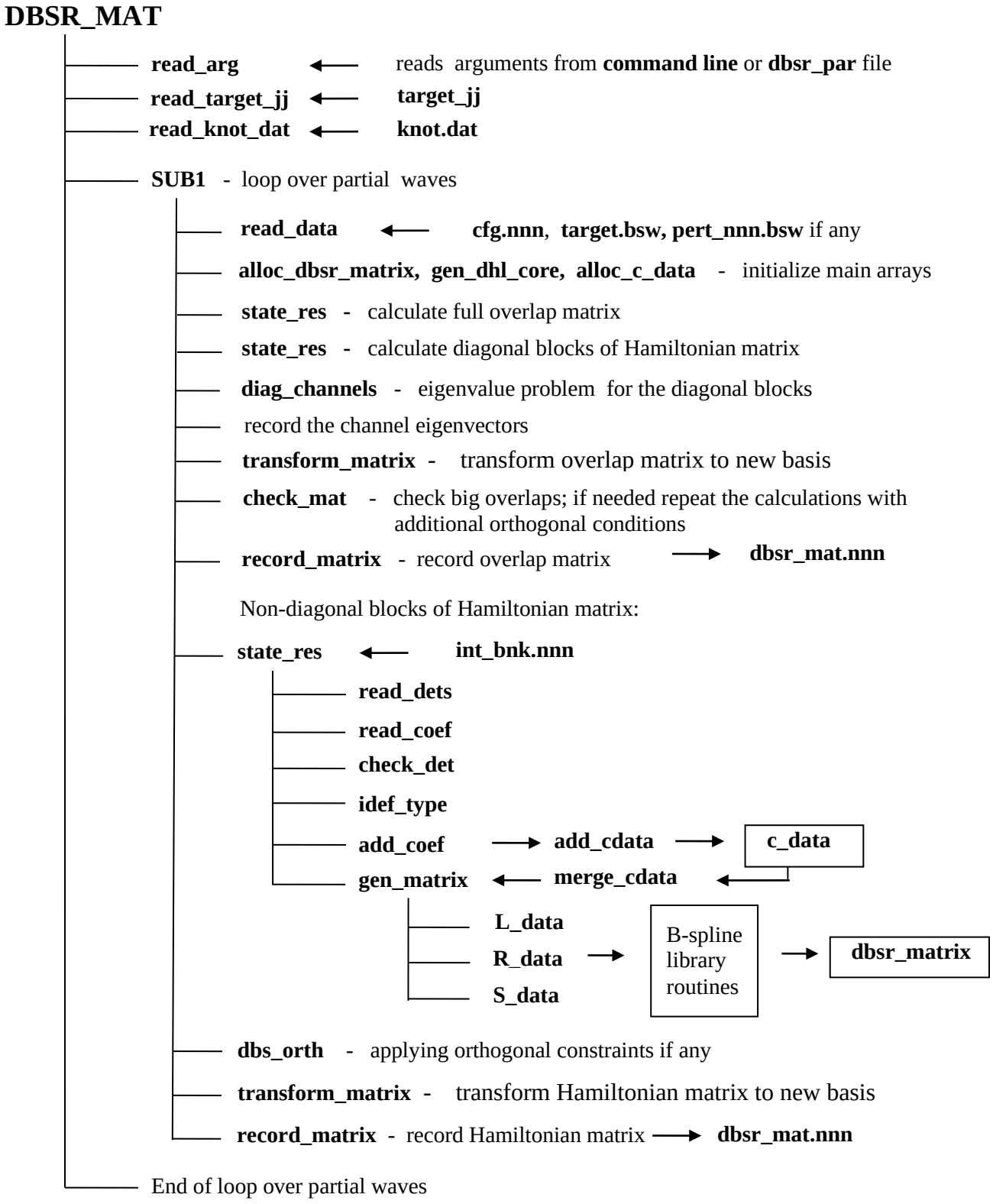

Fig. 7.1. Block diagram for the program DBSR_MAT and data flow (see text).

7.3. Data files

dbsr_par	File type: formatted sequential input. Written by user. Read by routine **read_arg**. Description: input parameters for given run.
target_jj	File type: formatted sequential input. Prepared by user and modified by DBSR_PREP and DBSR_CONF programs. Read by routine **read_target_jj**. Description: contains description of target states and scattering channels.
knot.dat	File type: formatted sequential input. Prepared by user or other programs. Read by routine **read_knot_dat** from DBS library. Description: input parameters that define the *B*-spline grid.
cfg.nnn	File type: formatted sequential input Created by program DBSR_CONF. Read by routine **read_conf_jj**. Description: list of configurations for given partial wave nnn.
int_bnk.nnn	File type: unformatted sequential input. Created by program DBSR_BREIT. Read by routine **state_res**. Description: databank for angular coefficients.
target.bsw	File type: unformatted sequential input. Created by program DBSR_PREP. Read by routine **read_pbsw**. Description: target's one-electron orbitals in *B*-spline basis.
pert_nnn.bsw	File type: unformatted sequential input. Created by program DBSR_PREP. Read by routine **read_pbsw**. Description: perturber's one-electron orbitals in *B*-spline basis, optional.
mat_log.nnn	File type: formatted sequential output. Written by program DBSR_MAT. Description: running information.
dbsr_mat.nnn	File type: unformatted sequential output (main result) Created by program DBSR_MAT. Read by program DBSR_HD. Description: Overlap/Hamiltonian matrixes and asymptotic coefficients.
int_mat.nnn	File type: formatted sequential output. Created by program DBSR_MAT. Description: list of one- and two-electron integrals. This optional output is only for debug purposes and controlled by the parameter **debug**.
target_new.nnn	File type: formatted sequential output. Created by program DBSR_MAT. Description: corrected target energies. This optional output is controlled by the parameter **iitar**.

7.4. Input data

Input data can be provided in the command line or in the input file **dbsr_par** (the data from the command line overwhelm the data from input file). Below we describe those data from **dbsr_par** which are read by program DBSR_MAT. All data have the default values indicated in the brackets. All default values, along with unit numbers and default file names, are placed in the module **dbsr_mat**.

klsp1 [1]	first partial wave for consideration.
klsp2 [klsp1]	last partial wave for consideration.
klsp [1]	the only partial wave to be considered.
mk [7]	maximum multipole index.
mbreit [0]	if = 1, the Breit interaction is included.
iitar [0]	mode flag for treatment of target states orthogonality: =0 - target state are eigenfunctions of the target Hamiltonian; =1 - target states orthogonal but may not diagonalize the target Hamiltonian; =2 - the most general case of non-orthogonal target states.
s_ovl [0.75]	tolerance for the total overlap matrix elements; if overlaps more than **s_ovl**, additional orthogonal constraints will be assigned.
eps_c [1.d-10]	tolerance for angular coefficients combined with numerical overlap factor; coefficients with absolute value less than **eps_c** are ignored.
eps_det [1.d-10]	tolerance for the value of the numerical overlap factor; overlap factors less than **eps_det** are taken as zero.
pnu [0.5]	parameter η in the Bloch operator (2.12).
RB [0.0]	parameter b in the Bloch operator (2.12).
Edmin [$-2mc^2$]	lower limit of the one-electron spectrum.
Edmax[1.d10]	upper limit of the one-electron spectrum.
ilzero [1]	number of B-splines deleted at r=0 for the large component.
ibzero [1]	number of B-splines deleted at border r=a for the large component.
jlzero [1]	number of B-splines deleted at r=0 for the small component.
jbzero [1]	number of B-splines deleted at border r=a for the small component.
mcbuf [100000]	size of buffer for coefficients read from **int_bnk**. The buffer was introduced in order to reduce the number of read-write operations.
mblock [5000]	the size of one block in module **c_data** used for accumulation of given type of integrals.
nblock [500]	number of blocks in module **c_data.**
kblock [500]	number of blocks for in module **c_data.**
debug [0]	if > 0, additional debug output will be given (channel eigenvalues, integral values and other running information).
mcbuf [100000]	buffer size for coefficients, read from **int_bnk**.

7.5.Output of Hamiltonian matrices and asymptotic coefficients

The main result of the DBSR_MAT program is the generation of the interaction matrix (2.18) in the *B*-spline representation (7.6). The latter is stored in the **dbsr_mat.nnn** files for each partial wave separately. Below is a summary of the output records in these files.

1. **ns,nch,npert,nsp,nsq**

   ns – maximum number of *B*-splines.
   nch – number of channels.
   npert – number of bound (*N*+1)-electron states.
   nsp – number of *B*-splines for the large component.
   nsq – number of *B*-splines for the small component.

2. **nsol, itype** – dimension of the new basis, obtained by diagonalization of channel blocks, and flag for the type of calculations.

3. **ipsol(1:nch)** – number of solution for each channel in new basis.

4. **eval(1:nsol)** – eigenvalues of solutions in new basis.

5. new basis in the B-spline representation:
   (((diag(i,js,ich),i=1,2×ns),js=1,ipsol(ich)),ich=1,nch)

6. Overlap matrix in new basis, recorded in blocks for low trigonal part of matrix (7.6)
   i,j
   H(1:ipsol(i),1:ipsol(j)) – channel-channel blocks, H_{ij}
   nch+i,j
   h(1: ipsol(j)) – channel-bound blocks, h_{ij}
   nch+i,nch+j
   b(i,j) – bound-bound blocks, b_{ij}
   The above records are repeated for all non-zero matrix elements, up to sign of the end:
   0,0

7. Hamiltonian matrix in new basis, recorded in blocks for low trigonal part of matrix.
   The same format as for the overlap matrix above.

8. **mk** - maximum multipole index.

9. **CF(1:nch,1:nch,0:mk)** – matrix of asymptotic coefficients (2.31).