Description of utility-programs for BSR complex (folder UTILS/BSR_utils)

In addition to the main BSR programs, a number of short utility-programs are available that assist in the processing and managing the data. This folder contains the utilities which directly working with B-splines or with files produced with the BSR programs.

Quick-references to the utilities

bound_tab	produces the total list of resulting bound states after BSR calculations
bound_bsw	converts the close-coupling B-spline expansions to the c- and bsw-files for individual state
w_bsw	converts the MCHF w-file to the BSR bsw-file
bsw_w	converts the BSR bsw-file to the MCHF w-file
bsw_tab	prepares the text files suitable for plotting the radial functions from bsw-file
bsw123	merging the set of bsw-files with user's choice of orbitals and with optional changing the spectroscopic notation
bs_hwf	provides hydrogen-like bound and continuum pseudo-state orbitals'on the given B-spline basis
print_bs	prints B-splines (in separate files) for given B-spline basis
sum_hh	merges the set of **h.nnn** files to the final H.DAT
h_targb	creates "**target**" file based on the "**H.DAT**" file
h_targn	creates "**target**" file based on the set of **nnn/h.nnn** files
target_exp	change **target** file to fit new experimental state energies, given in file **thresholds**
target_states	provide information about target states including their labels
zf_bb_bsr	calculates f-values between solutions in set of **bound.nnn**
zf_cc_bsr	calculates f-values between set of {**name.c+name.bsw**} files
bsr_dd	calculation of dipole matrix elements between R-matrix states saved in **rsol.nnn** files
photo_tab	produces the tables for channel or total photoionization cross sections and asymmetry parameters
photo_collect	collects data from **bsr_phot.nnnn** files to final **bsr_phot.nnn** file after MPI calculations if they not finished correctly

B O U N D _ T A B

Description:	produces the total list of resulting bound states
Input files:	**bound.nnn**
Output files:	**bound.tab**
Call:	**bound_tab** [**klsp =…**]

This program sorts the energies of states recorded in different **bound.nnn** files and prints them in atomic units, eV or cm^{-1}, relative to the lowest state. The user can define the range of partial waves and the energy range to restrict the output. The end of the file contains the parameters used by program. The range of partial waves is defined by **klsp1**, **klsp2**, and **klsp3**, as the initial, final and step for partial wave. **Emin**, **Emax** restrict the energy region under consideration (zero values mean no restrictions). Nuclear charge **Z** and **AWT** (if they are not zero) are used to define conversation factor from atomic units (Ry, au) to other units (eV, cm-1). **E0** defines the reference points to calculate excitation or bound energies. If reference point is not defined by user, the program chooses the energy of the first target state as the reference points. All above parameters can be redefined by corresponding command-line arguments (as parameter=value), or user can change these parameters directly in the **bound_tab** file and re-run the program.

```
klsp  sol   label     L  S  P     E_Ry        E_eV       E_cm       E_au

   1   1   4s_4s_1S   0  1  1   -0.217888   -5.92898   -47820.4   -676.80847440
   4   1   4s_4p_3P   1  3 -1   -0.147927   -4.02526   -32465.9   -676.73851340
   3   1   4s_4p_1P   1  1 -1   -0.109965   -2.99226   -24134.2   -676.70055090
   8   1   4s_3d_3D   2  3  1   -0.105841   -2.88004   -23229.1   -676.69642660
   7   1   4s_3d_1D   2  1  1   -0.102908   -2.80023   -22585.4   -676.69349370
   2   1   4s_5s_3S   0  3  1   -0.078572   -2.13803   -17244.4   -676.66915820
   1   2   4s_5s_1S   0  1  1   -0.070551   -1.91976   -15483.9   -676.66113670
  12   1   3d_4p_3F   3  3 -1   -0.059778   -1.62664   -13119.7   -676.65036440
   4   2   4s_5p_3P   1  3 -1   -0.056834   -1.54651   -12473.5   -676.64741990
.....................................................................................
   5   9   3d_kd_1P   1  1  1    0.081296    2.21216    17842.3   -676.50928960
   5  10   4p_6p_1P   1  1  1    0.083449    2.27072    18314.6   -676.50713740
*

Z    =  20.000
AWT  =   0.000

E0   =   -676.59058592
Emax =      0.00000000

klsp1 =   1
klsp2 =  14
klsp3 =   1

au_eV=        27.211055
au_cm=    219471.99

nstate =  218
```

Figure 1. Example of the **bound_tab** file.

B O U N D _ B S W

Description:	converts the close-coupling *B*-spline expansions to the *c*- and *bsw*-files
Input files:	**bound.nnn, bound_bsw.inp, target, knot.dat, cfg.nnn, target.bsw**
Output files:	*c*- and *bsw*-files for given states, indicated in the input file
Call as:	**bound_bsw** [**mode=…**] (if given input file **bound_bsw.inp**), or **bound_bsw klsp=... sol=... name=...** , where **klsp** - partial wave index **sol** - state index' **name** - name for given state

The results of the BSR bound-state calculations are recorded in the **bound.nnn** files in the form of the corresponding B-spline close-coupling expansions. The BOUND_BSW utility collects the information for a given bound state indicated in the file **bound_bsw.inp**, and records it as pair (**name.c** , **name.bsw**). These files can be used as input target files for the further BSR calculations or other applications.

Input file is usually created on the base of the **bound_tab** file created by utility BOUND_TAB and contains the list of states to be processed. Each line contains the information for one state as index of the partial wave, index of the state for given partial wave and assigned name. Asterisk denotes the end of the list. The names are usually assigned by user. In case of large-scale calculations with hundreds of states, it is more convenient to use additional optional argument **mode**. In this case, all names are assigned automatically as **mode_nnn_mmm**, where **nnn =<klsp>** – index of partial wave, **mmm=<sol>** index of the state. If needed to output a few individual states, the user can employ the command-argument option indicated above.

B S W _ W

Description:	converts the *bsw*-file to the *w*-file
Input files:	**name.bsw**
Output files:	**name.w**
Call:	**bsw_w name.bsw**

W _ B S W

Description:	converts the *w*-file to the *bsw*-file
Input files:	**name.w, knot.dat**
Output files:	**name.bsw**
Call:	**bsw_w name.w**

B S W _ T A B

Description:	prepares the text files suitable for plotting the radial functions from bsw-file
Input files:	**name.bsw**
Output files:	**name.tab**
Call as:	**bsw_tab name.bsw**

B S W 1 2 3

Description:	merging the set of bsw-files with user's choice of orbitals and with optional changing the spectroscopic notation
Input files:	set of bsw-files under consideration
Output files:	resulting **bsw-file** with chosen orbitals
Call as:	**bsw123** with interactive input (knot.dat is not required)

B S _H W F

Description:	provides hydrogen-like bound and continuum pseudo-state orbitals on the given B-spline basis
Input files:	**knot.dat**
Output files:	**nl_nnn.bsw** and **nl_nnn.c** for each orbital 'nnn'
Call as:	**bs_hwf l=... nsol=... ii=... jj=... emax=… eps_tail=…**
	l - orbital momentum **nsol** - number of solutions [1] **ii** - zero B-splines in the beginning [*l*+1] ' **jj** - zero B-splines in the end [1] ' **emax** - maximum energy [100] ' **eps_tail** - tolerance for the last B-spline [1.d-7] **(Z**-value is coming from **knot.dat)**

P R I N T _ B S

Description:	prints B-splines (in separate files) for given B-spline basis
Input files:	**knot.dat**
Output files:	**bs_nnn** for each spline 'nnn'
Call as:	**print_bs**

S U M _ H H

Description:	merges the set of **h.nnn** files to the final **H.DAT**
Input files:	**h.nnn**
Output files:	**H.DAT**
Call as:	**sum_hh** [**ih1=**… **ih2=**… **folder=**…] ih1 – initial nnn-index [1] ih2 – final nnn-index [999] folder – if > 0, we suppose paths: nnn/h.nnn

H _ T A R G B

Description:	creates "**target**" file based on the "**H.DAT**" file
Input files:	**H.DAT**
Output files:	**target.h**
Call as:	**h_targb** [**h=…**], where **h** can be used to redefine input name

H _ T A R G N

Description:	creates "**target**" file based on the set of **nnn/h.nnn** files (used in large-scaled calculations where each partial wave is considered in the separate folder)
Input files:	**nnn/h.nnn or nnn/H.DAT**
Output files:	**target.hhh**
Call as:	**h_targn klsp=…**, where **klsp** defines number of **nnn** folders

T A R G E T _ E X P

Description:	change **target** file to fit new experimental state energies, given in file **thresholds** (see also BSR_HD, parameter iexp)
Input files:	**target.theory + thresholds**
Output files:	**target.exp**
Call as:	**target_exp** [**targ1=... targ2=... thresholds=...**] parameter are used to re-define the default names

T A R G E T _ S T A T E S

Description:	provide information about target states including their labels
Input files:	**target + {name.c}**
Output files:	**target_states**
Call as:	**target_states** [**inp=... out=...**] parameter are used to re-define the default names

B S R _ D D

Description:	calculation of dipole matrix elements between R-matrix states saved in **rsol.nnn** files
Input files:	**{rsol.nnn}, cfg.nnn, target.bsw, knot.dat**
Output files:	**dd_nnn_mmm** (nnn,mmm - indexes of partial waves)
Call as:	**bsr_dd klsp=...** klsp – number of partial waves Program makes SYSTEM CALLS to MULT3 and BSR_DMAT3

Z F _ B B _ B S R

Description:	calculates f-values between solutions in set of **bound.nnn**
Input files:	**bound.nnn , cfg.nnn, target.bsw, knot.dat +** [**zf_bb.inp**]
Output files:	**zf_res**
Call as:	**zf_bb_bsr** [**klsp=… msol=… atype=… gf=…**]
	or provide input file **zf_bb_inp** with:
	atype = E1 |E2|M1|… - multipole index
	gf = f |g - output f- or gf-values'
	klsp= ... - number of **bound.nnn** files to be considered
	msol = … - maximum number of solutions in **bound.nnn**
	followed by the list of ilsp(i), mstates(i), i=1,klsp for each partial wave in separate line, where ilsp -index of partial and mstates -number of states
	Program makes SYSTEM CALLS to MULT3 and BSR_DMAT3

Z F _ C C _ B S R

Description:	calculates f-values between set of {**name.c+name.bsw**} files
Input files:	**{name.c+name.bsw}, knot.dat, zf_cc.inp**
Output files:	**zf_res**
Call as:	**zf_cc_bsr** [**inp**=…]
	input file contains:
	atype = E1 |E2|M1|... - multipole index
	gf = f |g - output f- or gf-values
	param =… - additional parameters for bsr_dmat3
	nfiles= ... - number of c-files
	followed by list of c-files in separate lines
	Program makes SYSTEM CALLS to MULT3 and BSR_DMAT3

P H O T O _ T A B

Description:	produces the tables for channel or total photoionization cross sections and asymmetry parameters
Input files:	**bsr_phot.nnn, target**
Output files:	indicated by user
Call as:	**photo_tab** with interactive response from command line

This utility serves for processing the data recorded during photoionization calculations with program BSR_PHOT. The program BSR_PHOT can be run for different energy intervals and different partial waves separately. The results are accumulated in files **bsr_phot.nnn**. Then final tables for channel or total cross sections and asymmetry parameters may be generated with the utility **photo_tab**. More detailed description of this utility and used formulas are given in the BSR_PHOT write-up.

P H O T O _ C O L E C T

Description:	collects data from **bsr_phot.nnnn** files to final **bsr_phot.nnn** file after MPI calculations if they not finished correctly
Input files:	**bsr_phot.nnnn**
Output files:	**bsr_phot.nnn, photo.nnn**
Call as:	**photo_collect klsp=… np=…**
	klsp – index of partial wave under consideration
	np – number of processors

In MPI versions of BSR_PHOT, each processor considers different set of photon energies and records results into separate **bsr_phot.nnnn** files, where **nnnn** – index of the processor. In the end of calculations, the information collect into the **bsr_phot.nnn** file, where the **nnn** – index of partial wave under consideration. If the calculations finished not correctly (time limits or problem for some energy), the **photo_collect** utility may be used to collect the results for energies which were processed correctly.

f _ v a l u e s

Description:	calculation of f-values for transition between target states based on the asymptotic coefficients in the H.DAT file
Input files:	**H.DAT** or **h.nnn, target**
Output files:	**f_values** or **f_values.nnn**
Call as:	**f_values** [**h**=… **klsp**=…]
	h [H.DAT] – alternative name for input file klsp [1] - partial wave index to choose **h.nnn** file for the process

s _ v a l u e s

Description:	calculation of s-values for transition between target states based on the asymptotic coefficients in the H.DAT file
Input files:	**H.DAT** or **h.nnn, target**
Output files:	**s_values** or **s_values.nnn**
Call as:	**s_values** [**h**=.. **klsp1**=… **klsp2**=… **L1**=… **L2**=...] h [H.DAT] - alternative name for H.DAT file klsp1[0] -

h [H.DAT]	alternative name for H.DAT file
klsp1[0]	minimum index of partial wave to be considered
klsp2[0]	maximum index of partial wave to be considered
L1[-1]	minimum total orbital moment to be considered
L2[-1]	maximum total orbital moment to be considered

Asymptotic coefficients decomposition

LS coupling

The long-range potential coefficients coupling two channels are

$$ACF(i,j,k) = 2a_{ij}^{k} = 2 < \overline{\Phi}_i(x_1 \ldots x_N, \hat{\boldsymbol{r}}_{N+1}\sigma_{N+1}) \,|\, \sum_{n=1}^{N} r_n^k P_k(\cos \hat{\boldsymbol{r}}_N \bullet \hat{\boldsymbol{r}}_{N+1}) \,|\, \overline{\Phi}_j(x_1 \ldots x_N, \hat{\boldsymbol{r}}_{N+1}\sigma_{N+1}) > \quad (1)$$

In tensor notation

$$a_{ij}^{k} = < \alpha_i L_i S_i l_i s; LM_L SM_S \,|\, \boldsymbol{M}^k \bullet \boldsymbol{C}^k \,|\, \alpha_j L_j S_j l_j s; LM_L SM_S > \quad (2)$$

where

$$M_q^k = \left(\frac{4\pi}{2k+1}\right)^{1/2} \sum_{n=1}^{N} r_n^k Y_q^k(\hat{\boldsymbol{r}}_n) \quad (3)$$

and

$$C_q^k = \left(\frac{4\pi}{2k+1}\right)^{1/2} Y_q^k(\hat{\boldsymbol{r}}_{N+1}) \ . \quad (4)$$

To evaluate expression (2), we may use the general expression (see, e.g., Cowan 1981, Eq.11.47) for matrix elements of a scalar product when angular momenta j_1, j_2 correspond to different subsystems

$$< j_1 j_2 jm \mid P^{(k)}(1) \bullet Q^{(k)}(2) \mid j'_1\ j'_2\ j'm' >$$
$$= \delta_{j,j'}\delta_{m,m'}(-1)^{j'_1+j_2+j}\begin{Bmatrix} j_1 & j_2 & j \\ j'_2 & j'_1 & k \end{Bmatrix} < j_1 \| P^{(k)} \| j'_1 >< j_2 \| Q^{(k)} \| j'_2 > \tag{5}$$

Then coefficients (2) are reduced to

$$a_{ij}^k = (-1)^{L_j+l_i+L} < l_i \| C^{(k)} \| l_j > \begin{Bmatrix} L_i & l_i & L \\ l_j & L_j & k \end{Bmatrix} < \alpha_i L_i \| M^{(k)} \| \alpha_j L_j > \tag{6}$$

This expression can be used for determination radiative matrix elements for transitions between target states from the asymptotic coefficients in *LS* coupling case.

jj coupling

$$a_{12}^k =< \alpha_1 J_1 (l_1 s) j_1; JM_J \mid \boldsymbol{M}^k \bullet \boldsymbol{C}^k \mid \alpha_2 J_2 (l_2 s) j_2; JM_J >$$
$$= (-1)^{J_2+j_1+J} < (l_1 s) j_1 \| C^{(k)} \| (l_2 s) j_2 > \begin{Bmatrix} J_1 & j_1 & J \\ j_2 & J_2 & k \end{Bmatrix} < \alpha_1 J_1 \| M^{(k)} \| \alpha_2 J_2 > \tag{7}$$

Here we can use the uncoupling formula when operator operates only within the first subspace (see Cowam 1981, Eq.11.38):

$$< j_1 j_2 j \| P^{(k)}(1) \| j'_1\ j'_2\ j' > = \delta_{j_2,j'_2}(-1)^{j_1+j_2+j'+k}[j, j']^{1/2}\begin{Bmatrix} j_1 & j_2 & j \\ j' & k & j'_1 \end{Bmatrix} < j_1 \| P^{(k)} \| j'_1 > \tag{8}$$

Then

$$a_{12}^k = (-1)^{J_2+j_1+J+l_1+s+j_2+k}[\mathrm{j}_1, \mathrm{j}_2]^{1/2}\begin{Bmatrix} l_1 & s & j_1 \\ j_2 & k & l_2 \end{Bmatrix} < l_1 \| C^{(k)} \| l_2 > \begin{Bmatrix} J_1 & j_1 & J \\ j_2 & J_2 & k \end{Bmatrix} < \alpha_1 J_1 \| M^{(k)} \| \alpha_2 J_2 > \tag{9}$$

jK coupling

$$a_{12}^k =< \alpha_1 (J_1 l_1) K_1, s; JM_J \mid \boldsymbol{M}^k \bullet \boldsymbol{C}^k \mid \alpha_2 (J_2 l_2) K_2, s; JM_J > \tag{10}$$

First we should uncoupled the J_1 and J_2 by transferring to *jj*-coupling (Cowan 1981, Eq.9.25):

$$< (J_1 l_1) K_1, s; J \mid J_1, (l_1 s) j_1; J >= (-1)^{J_1+l_1+s+J}[K_1, j_1]^{1/2}\begin{Bmatrix} J_1 & l_1 & K_1 \\ s & J & j_1 \end{Bmatrix} \tag{11}$$

Then

$$a_{12}^{k} = \sum_{j_1,j_2} (-1)^{J_1+l_1+s+J}[K_1, j_1]^{1/2}\begin{Bmatrix} J_1 & l_1 & K_1 \\ s & J & j_1 \end{Bmatrix}(-1)^{J_2+l_2+s+J}[K_2, j_2]^{1/2}\begin{Bmatrix} J_2 & l_2 & K_2 \\ s & J & j_2 \end{Bmatrix}$$
$$\times(-1)^{J_2+j_1+J+l_1+s+j_2+k}[j_1, j_2]^{1/2}\begin{Bmatrix} l_1 & s & j_1 \\ j_2 & k & l_2 \end{Bmatrix} < l_1 \| C^{(k)} \| l_2 > \begin{Bmatrix} J_1 & j_1 & J \\ j_2 & J_2 & k \end{Bmatrix} < \alpha_1 J_1 \| M^{(k)} \| \alpha_2 J_2 >$$
$$= \sum_{j_1,j_2} (-1)^{J_1+2J_2+3J+j_1+j_2+l_2+3s+k}[j_1, j_2]\begin{Bmatrix} J_1 & l_1 & K_1 \\ s & J & j_1 \end{Bmatrix}\begin{Bmatrix} l_1 & k & l_2 \\ j_2 & s & j_1 \end{Bmatrix}\begin{Bmatrix} J_1 & k & J_2 \\ j_2 & J & j_1 \end{Bmatrix} \tag{12}$$
$$\times[K_1, K_2]^{1/2}\begin{Bmatrix} J_2 & l_2 & K_2 \\ s & J & j_2 \end{Bmatrix} < l_1 \| C^{(k)} \| l_2 >< \alpha_1 J_1 \| M^{(k)} \| \alpha_2 J_2 >$$

Now let reduce sum over j_1, using the sum rule (Cowan 1981, Eq. 5.33):

$$\sum_x (-1)^{S+x}[x]\begin{Bmatrix} l_1 & j_2 & l_3 \\ l'_3 & l'_2 & x \end{Bmatrix}\begin{Bmatrix} j_2 & j_3 & j_1 \\ l'_1 & l'_3 & x \end{Bmatrix}\begin{Bmatrix} l_1 & j_3 & l_2 \\ l'_1 & l'_2 & x \end{Bmatrix} = \begin{Bmatrix} j_1 & j_2 & j_3 \\ l_1 & l_2 & l_3 \end{Bmatrix}\begin{Bmatrix} l_3 & j_1 & l_2 \\ l'_1 & l'_2 & l'_3 \end{Bmatrix} \tag{13}$$

where S = $j_1 + j_2 + j_3 + l_1 + l_2 + l_3 + l'_1 + l'_2 + l'_3$. Then

$$a_{12}^{k} = \sum_{j_2} (-1)^{J_1+2J_2+3J+j_2+l_2+3s+k-l_2-l_1-k-J_1-J_2-K_1-j_2-J-s}[j_2]\begin{Bmatrix} l_2 & l_1 & k \\ J_1 & J_2 & K_1 \end{Bmatrix}\begin{Bmatrix} K_1 & l_2 & J_2 \\ j_2 & J & s \end{Bmatrix}$$
$$\times[K_1, K_2]^{1/2}\begin{Bmatrix} J_2 & l_2 & K_2 \\ s & J & j_2 \end{Bmatrix} < l_1 \| C^{(k)} \| l_2 >< \alpha_1 J_1 \| M^{(k)} \| \alpha_2 J_2 >$$
$$= \sum_{j_2} (-1)^{J_2+2J+2s-l_1-K_1}[j_2]\begin{Bmatrix} J_2 & l_2 & K_1 \\ s & J & j_2 \end{Bmatrix}\begin{Bmatrix} J_2 & l_2 & K_2 \\ s & J & j_2 \end{Bmatrix} \tag{14}$$
$$\times[K_1, K_2]^{1/2}\begin{Bmatrix} l_2 & l_1 & k \\ J_1 & J_2 & K_1 \end{Bmatrix} < l_1 \| C^{(k)} \| l_2 >< \alpha_1 J_1 \| M^{(k)} \| \alpha_2 J_2 >$$

Using the orthogonal relation (Cowan 1981, Eq.5.31)

$$\sum_x [x]\begin{Bmatrix} j_1 & j_2 & a \\ l_1 & l_2 & x \end{Bmatrix}\begin{Bmatrix} j_1 & j_2 & b \\ l_1 & l_2 & x \end{Bmatrix} = \frac{\delta(a,b)}{2a+1} \tag{15}$$

finally obtain

$$a_{12}^{k} = \delta(K_1, K_2)(-1)^{J_2+2J+1-l_1-K_1}\begin{Bmatrix} l_2 & l_1 & k \\ J_1 & J_2 & K_1 \end{Bmatrix} < l_1 \| C^{(k)} \| l_2 >< \alpha_1 J_1 \| M^{(k)} \| \alpha_2 J_2 > \tag{16}$$