Description of utility-programs dealing with the configuration expansions in JJ-coupling (folder UTILS/CONF_JJ)

Quick-references to the utilities

genjconf	prepares the list of configurations (without terms indication) from the list of electron occupations and allowed promotions
genjterm	prepares the list of CSF's from a list of the configurations
cm_j	converts *cm*-file after RCI2 program from GRASP complex or *m*-file after RSCF2 program to *j*-file of DBSR_CI format
jcfile	extracts the state expansion from DBSR_CI j-file
jj_order	ordering the CSFs according their weights
jfile	find the maximum contributions for given configurations from the list of solutions in j-file
jj_list	creates list of states from one or set of j-files

G E N J C O N F

Description:	prepares the list of possible configurations (without terms indication) from the list of electron occupations and allowed promotions.
Input files:	**nlj.inp** (default) or define by parameter **inp**
Output files:	**conf.inp** (default) or define by parameter **out**
Call as:	**genjconf inp=... out=...** or **atom=...**

All arguments are optional. If input file is absent, it will be created for the user to fill it in. Example of input file with comments can be created for the given atom using the parameter **atom**.

```
Core subshells:
  1s   2s   2p-  2p   3s   3p-  3p
Peel subshells:
  3d-  3d   4s
Occupation limits:
    0    2    2
    4    6    2


n_orbitals  =  3        ! number of peel subshels
n_electrons =  8        ! number of peel electrons
parity =  1             ! +1 or -1
k_ref  =  3             ! reference orbitals
k_min  =  0             ! min. promotion from the ref. orbitals
k max  =  2             ! max. promotion from the ref. orbitals
```

Figure 1. Example of the **nlj.inp** file.

```
Core subshells:
  1s   2s   2p-  2p   3s   3p-  3p
Peel subshells:
  3d-  3d   4s
CSF(s):
  3d-( 4)  3d ( 4)
  3d-( 4)  3d ( 2)  4s ( 2)
  3d-( 3)  3d ( 5)  4s ( 1)
  3d-( 3)  3d ( 3)  4s ( 2)
  3d-( 2)  3d ( 6)  4s ( 1)
  3d-( 2)  3d ( 4)  4s ( 2)
  3d-( 1)  3d ( 5)  4s ( 2)
  3d ( 6)  4s ( 2)
***
```

Figure 2. Example of the **conf.inp** file obtained on base of the information in the **nlj.inp, Fig.1.**

G E N J T E R M

Description:	prepares the list of CSF's from a list of the configurations
Input files:	**conf.inp** (default) or define by parameter **inp**
Output files:	**cfg.inp** (default) or define by parameter **out**
Call as:	**genjterm inp=... out=... jmin=… jmax=…** or **genjterm name jmin=… jmax=…**

All arguments except **jmin** are optional (if jmax < jmin, jmax = jmin). If name is given, input file is **name.conf** and output file is **name.c**.

```
Core subshells:
  1s   2s   2p-  2p   3s   3p-  3p
Peel subshells:
  3d-  3d   4s
CSF(s):
  3d-( 4)  3d ( 4)
                  0
                   0+
  3d-( 4)  3d ( 2)  4s ( 2)
                  0
                              0+
*******************************
  3d-( 2)  3d ( 4)  4s ( 2)
        2         2
                   0         0+
  3d ( 6)  4s ( 2)

                   0+
 *
  3d-( 3)  3d ( 3)  4s ( 2)
      3/2       5/2
                   1         1+
*******************************
  3d-( 1)  3d ( 5)  4s ( 2)
      3/2       5/2
                   1         1+
 *
  3d-( 4)  3d ( 4)
                  2
                   2+
*******************************
  3d-( 1)  3d ( 5)  4s ( 2)
      3/2       5/2
                   2         2+
*
```

Figure 3. Example of the **conf.inp** file obtained on base of the information in the **nlj.inp,** Fig.1.

C M _ J

Description:	convert *cm*-file after RCI2 program from GRASP complex or *m*-file after RSCF2 program to *j*-file of DBSR_CI format.
Input files:	**name.cm** (or **name.m**), **name.c**
Output files:	**name.j**
Call as:	**`cm_j name.cm (or name.m)`**

J C F I L E

Description:	extracts the state expansion from DBSR_CI j-file
Input files:	**name.j, name.c**
Output files:	**result.c, result.bsw**
Call as:	**jcfile name.j nn result eps_c**

In the DBSR complex, each target state is described by a pair **name.c** and **name.bsw**. The DBSR_CI program produces the results in the j-files as a list of states with their expansions. The JCFILE utility allows one to extract the expansion coefficients for a given state and create the corresponding *c*- and bsw-files. The user should indicate in the command line the name of the input *j*-file, the index of the needed state, **nn**, the name for the output *c*-file, and the tolerance for the expansion coefficients, **eps_c**. When **eps_c** > 0, configurations in result c-file are ordered according their weights.

O R D E R _ J J

Description:	ordering the configurations according their weights
Input files:	**name.c**
Output files:	**name.cc**
Call as:	**order_jj name.c [eps_c]**
	eps_c [0] - optional cut-off parameter

J F I L E

Description:	finds the maximum contributions for given configurations from the list of solutions in j-file.
Input files:	**name.j, name.c**
Output files:	**name.cc**
Call as:	**jcfile name [eps_c]**
	eps_c [0] - optional cut-off parameter

Example: **jfile 1 sol=1-5 eps_c=0.0001**

In this case, program analyses the first 5 solutions in 1.j and outputs (in 1.cc) all configurations in order of importance. Now we may delete all configurations with small coefficients and used this file for another CI calculations.

J J _ L I S T

Description:	creates list of states in one j-file (given as argument) or in set of j-files
Input files:	**name.j, name.c**
Output files:	**name.list** (can be redefined by parameter **out)**
Call as:	**jj_list name.j** [**unit=… shift=… Z=… awt=… eps_c=…**]
	Optional key-words parameters:
	unit - au, Ry, eV, cm [au]
	shift - overall energy shift [0.d0]
	msol - max.number of solutions [0 -> all]
	Z - nuclear charge
	awt - atomic weight
	eps_c - tolerance for configuration weights [0.2]
	AFi - input file with list of j-files and parameters [**jj_list.inp**]
	AFr - output file **[jj_list]**

List of j-files under consideration are defined as not key-words arguments. This list (F1.j, F2.j, ...), along with other parameters, can also be given in input file defined through **inp**=..., with one filename per line. The order of parameters or merging file does not matter. Parameters **Z, awt** are used for more accurate conversion factor from atomic units to eV. If **shift**=1.d0, the reference point will be the lowest state.

R S A V E

Description: script file to save results of an GRASP calculations under specific name

```
cp rwfn.out   %1.w
cp rcsl.inp   %1.c
mv rmix.out   %1.m
mv rscf.sum   %1.sum
```