Advanced usage of MOLCAS

- Ex 1. Symmetry in Molcas (20 min).
- Ex 2. Transition state optimization (30 min).
- Ex 3. Ways to run RASSCF program (30 min).
- Ex 4. Optional. Using EXPBAS module (20 min)
- Ex 5. Optional. Using RAS-probing (20 min)
- Ex 6. Basic CASPT2/RASPT2 calculation: Butadiene (30 min)
- Ex 7. Basic CASPT2/RASPT2 calculation: Acrolein (20 min)
- Ex 8. Optional. Visualization of excited state calculations (20 min)
- Ex 9. If everything above was too trivial...

Ex. 1. Usage of Symmetry in Molcas

Prepare an input for SCF calculation of water molecule, and run it.

Now repeat the calculation but remove (or comment /* */) a line GROUP=C1. The code will determine the symmetry (as a subgroup of D_{2h}) and use it. Observe the difference between calculation with and without symmetry. Did you get the same energy, charges, dipole moment? Discuss a possibility when these values can differ!

Detect the orbitals which belongs to different irreps. Visualize them!

Optionally repeat the excercise for benzene.

Ex. 2. Transition states and geometry optimization

The search for a transition state is more complex in comparison to minimal energy search. Thus, you have to 'push' your system towards to the solution.

Prepare the following input:

```
 &Gateway
coord
3
modified water coordinates
O              0.39        0.00         0.00
H1             0.71        0.00         0.82
H2             0.71        0.00        -0.82
basis=ANO-S-VDZ
group=C1

>>> Do while
 &SEWARD
 &SCF
   KSDFT=B3LYP
 &SLAPAF
   TS
>>> EndDo
```

Note, that coordinates of water are modified!

Run the calculation, and use *geo.molden* file to observe the convergence to the transition state.

Optional: Another possibility to search TS is to use constrains in the input:

```
 &Gateway
coord=Water.xyz
Basis=ANO-S-VDZ
Group=C1

 Constraints
   a = Angle H2 O1 H3
 Value
   a = 180.0 degree
 End of Constraints

>>> Do while
 &SEWARD
 &SCF
    KSDFT=B3LYP
 &SLAPAF
    FindTS
>>> EndDo
```

Ex. 3. Ways to run RASSCF program

Objective: We will learn how to set up CASSCF/RASSCF calculation based on graphically selected active space

Create an input for SCF calculation of benzene molecule.

```
 &GATEWAY
   coord=benzene.xyz
   basis=ANO-S-VDZ
   group=C1
 &SEWARD
 &SCF
 &GRID_IT
 ASCII
 ALL
```

Run it and open the *dos.grid* file with GV. Detect six π orbitals (What is the difference between σ and π orbitals?), and mark them as active orbitals (use key 'a' for this operation). The header of GV window should say 2:6 (2 is a short name for RAS2, a.k.a. active space). Note that if you did not use keyword ALL in GRID_IT, you might be able to find only 5 π orbitals.

Save the file by pressing F2. Now you should have a file with extension GvOrb: benzene.dos.GvOrb

Modify the input file.

```
 &GATEWAY
```

```
  coord=Benzene.xyz
  basis=ANO-S-VDZ
  group=C1
&SEWARD
&RASSCF
fileorb=$CurrDir/benzene.dos.GvOrb
Spin=1
Charge=0
&GRID_IT
ASCII
name
output
```

Note that $\$CurrDir$ is a case sensitive: this is a location of your input file. Do not forget to add to Extra files: Benzene.xyz and benzene.dos.GvOrb

Run the calculation. Check benzene.output.dos.grid file to verify that orbital space did not change - you should see only 6 π orbitals in the file. If some of the orbitals is a σ character - the calculation converged to a wrong solution.

Let's have a quick look at RASSCF output. Identify the number of inactive orbitals and active orbitals. How to compute the total number of electrons?

```
      Basis functions          66

++    Wave function specifications:
      -----------------------------

      Number of closed shell electrons          36
      Number of electrons in active shells       6
      Number of inactive orbitals               18
      Number of active orbitals                  6
      Number of secondary orbitals              42
```

Identify the number of CSF. Check that the starting point was taken from GvOrb file. Identify convergence, weights of the main configurations, and occupation numbers for active orbitals.

```
      printout of CI-coefficients larger than  0.05 for root  1
      energy=    -230.705162
      conf/sym  111111     Coeff  Weight
             1  222000  -0.93637 0.87679
            13  220020   0.15247 0.02325
```

```
          18  2udud0  -0.12299 0.01513
          27  u2d0ud   0.11525 0.01328
          29  202200   0.15247 0.02325
          38  ud2u0d  -0.11525 0.01328
          47  2uudd0  -0.13585 0.01846

 Natural orbitals and occupation numbers for root  1
 sym 1:   1.956411   1.894084   1.894084   0.107100   0.107100   0.041220
```

Check MO coefficients for orbitals, and observe that some obrital energy are ”zero”.

Optional. Choose another active orbitals, and rerun the calculation. Do you get the same active space at the end?

Optional. set up RASSCF calculations. Use 1 and 3 to mark orbitals as belonging to RAS1 and RAS3 space. Add RASSCF keyword into RASSCF input section, followed by two numbers: maximum amount of holes in RAS1, and maximum amount of electrons in RAS3. Note, that using RAS2 simultaneously with RAS1/3 can become very expensive, and use much more memory. If the code stops, complaining about the allocation of memory add into the input:

```
>> export MOLCASMEM=1000
```

Example:

```
&RASSCF
fileorb=$CurrDir/benzene.dos.GvOrb
Spin=1
Charge=0
RASSCF= 1 1
```

The alternative input to CASSCF could be written in the following form:

```
&RASSCF
Inactive=34
Nactel=6 0 0
RAS2=6
Spin=1
```

Inactive - the number of doubly occupied orbitals.
RAS2 - the number of active orbitals
Nactel - number of active electrons, max. number of holes in RAS1, number of electrons in RAS3.
Spin - spin multiplicity (1 for singlet)
Note! since there is no orbital input, the code will select active space around HOMO-LUMO gap.

Ex 4. Optional. Using EXPBAS module

Objective: selection of active space is more simple if a small basis set is used. EXPBAS is a module, which convert orbital file, obtained for a small basis set into an orbital file for another basis set. Note, currently this feature works only with ANO type of basis sets.

The content of benzene.expbas.input:

```
/* first part of the calculation.
   Remove this line for the second run */

&GATEWAY
coord=$MOLCAS/Coord/Benzene.xyz
basis=ANO-S-VDZ
group=C1
&seward
&grid_it
all; ASCII
>>COPY $Project.RunFile $CurrDir/Small.RunFile

>>UNIX echo "Please, select 6 active orbitals"
>>UNIX molcas gv $Project.grid
>>COPY $Project.GvOrb $CurrDir/$Project.GvOrb

>>>>>>>>>>>     exit    <<<<<<<<<<<<<<<<<<<
/*  Remove this line for the second run
 */
```

```
&GATEWAY
coord=$MOLCAS/Coord/Benzene.xyz
basis=ANO-L-VDZP
group=C1

>>COPY $CurrDir/Small.RunFile RUNFIL1
>>COPY $CurrDir/$Project.GvOrb INPORB
>>COPY $Project.RunFile RUNFIL2

&EXPBAS

&SEWARD

&RASSCF
Spin=1
FileOrb=$Project.ExpOrb
```

The calculation split into two parts. The first one - ANO-S-VDZ calculation of benzene molecule.

At the end of this input RunFile is copied into input directory, and GV program is automatically runs. If you run the example on a remote machine, you have to run GV code manually.

Using GV code select 6 active orbitals, and save the file (F2).

To run the second part of the calculation - comment the first block /* */. (Note, that $> EXIT$ command

The second calculation uses larger basis set ANO-L-VDZP.

$>> COPY$ command similar to UNIX cp command. EXPBAS needs as an input: old and new RunFiles, and orbital file. It produces .ExpOrb file which is used by RASSCF.

Check the log file, and identify starting point for CASSCF calculation.

Ex 5. Optional. Using RAS-probing

Objective: In complicated cases, graphical analysis of orbitals is not conclusive. Small RASSCF calculation can be helpful in identifying orbitals which should be included into active space.

We will use a similar to previous example setup: benzene molecule, ANO-S-VDZ basis set. But this time, we will select only RAS1 and RAS3 orbitals (use keys: 1 and 3). We also have to add RASSCF keyword, so we will use only single excitations. And finally, we have to request more memory (MOLCASMEM=1000 means 1Gb of RAM will be used).

Please, select not more than 18 orbitals in total.

```
&GATEWAY
coord=$MOLCAS/Coord/Benzene.xyz
basis=ANO-S-VDZ
group=C1
&seward
&grid_it
all; ASCII

>>UNIX echo "OK. we now select only RAS1 and RAS3 spaces"
>>UNIX molcas gv $Project.grid
>>COPY $Project.GvOrb $CurrDir/$Project.GvOrb

/* note, we need more memory! */
>>export MOLCASMEM=1000

&RASSCF
Spin=1
RASSCF=1 1
FileOrb=$Project.GvOrb
&grid_it
name=RAS-probe
all; ASCII
```

Observe the output and identify the orbitals which can be included into the active space in following calculations. (Check the occupation numbers first!).

Ex 6. Basic CASPT2/RASPT2 calculations: Butadiene

Objective: Selection of active space depends on the problem, and can vary even for the same system.

Note: the input below requires an intermediate run of GV, and so, it can be executed if you have installation of MOLCAS at your computer

Input file butadiene.inp

```
&GATEWAY
  COORD = butadiene.xyz
  BASIS = ANO-S-VDZP
  GROUP =  C1
&SEWARD
&SCF
&GRID_IT
   ALL; ASCII

>>>>>>>> UNIX molcas gv $Project.grid <<<<<<<<

>> COPY $Project.GvOrb $CurrDir/$Project.GvOrb

&RASSCF
  FILEORB=$CurrDir/$Project.GvOrb

&GRID_IT
  FILEORB=$Project.RasOrb
  NAME=RASSCF
  ALL; ASCII

&CASPT2
```

if Molcas farm is used, the input above should be split:

First run:

```
&GATEWAY
  COORD = butadiene.xyz
  BASIS = ANO-S-VDZP
  GROUP =  C1
&SEWARD
&SCF
&GRID_IT
   ALL; ASCII
```

Grid file is modified by GV.

Second run:

```
&RASSCF
  FILEORB=$Project.GvOrb

&GRID_IT
  FILEORB=$Project.RasOrb
  NAME=RASSCF
  ALL; ASCII

&CASPT2
```

Note, that if running on farm:

- GvOrb file should be added as extra file
- CurrDir and WorkDir are the same directories

While selecting the active orbitals we can make different decisions.

We will identify π shaped orbitals. The choice among occupied orbitals is straightforward (14, 15), but for the virtual orbitals we can find a pair 16 and 20, but also 22 and 26.

Output. What should we look at?

- before we go...
  - Always look at the top part, it contains echo of your input and details of calculations
  - Check the header of GATEWAY module: it tells you - how much memory has been used, was it serial or parallel run.

    ```
    )()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()
        MOLCAS executing module GATEWAY with 256 MB of memory
                    at 09:29:35 Thu Aug  1 2013
    ()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()()
    ```

  - Check geometry, and basis set information.

- in RASSCF output
  - Orbital specification
  - Starting orbitals. It is easy to make a mistake here!
  - Charge (especially if you set up input using Nactel keyword)
  - Convergence (problems with convergence might indicate problems with active space)
  - CI-coefficients
  - Natural orbitals and occupation numbers
  - Total energy

    ```
    ::    RASSCF root number  1 Total energy =       -155.00591310
    ```

  - Properties (e.g. do you have symmetry, and reasonable values in Mulliken charges)
- in CASPT2 output
  - Orbital specification
  - details of calculation

    ```
          Type of  Fock operator to use: STANDARD
          Type of HZERO operator to use: STANDARD IPEA
    ```

  - FINAL CASPT2 RESULT:

    ```
          Reference energy:        -155.0059131011
          Total energy:            -155.4741576347
          Reference weight:           0.85661
    ```

    Small reference weight and/or large correction to energy might indicate an intruder state

  - Contributions to the CASPT2 correlation energy

    ```
          Active & Virtual Only:         -0.0108603857
          One Inactive Excited:          -0.1573335768
          Two Inactive Excited:          -0.3000505619
    ```

  - Report on small energy denominators, large coefficients, and large energy contributions
  - Properties

Do we need to include all 6 orbitals, or 4 is enough? Try to do both calculations, and compare the results.

Ex 7. Basic CASPT2/RASPT2 calculations: Acrolein

Objective: Set up of multistate CASSCF/CASPT2 calculations

In this exercise we will use Acrolein molecule.

Step 1. Preparing the orbitals.

```
&GATEWAY
 coord = $MOLCAS/Coord/Acrolein.xyz
 basis = ANO-S-VDZ; group= c1
&SEWARD
&SCF
&GRID_IT
 All; ASCII
```

run GV, and select the minimal active space (5 orbitals: 13, 14, 15, 16, 18)
Save the orbital file, and rename it to acrolein.GvOrb

Now, we can run CASPT2 calculation:

```
&GATEWAY
 coord = $MOLCAS/Coord/Acrolein.xyz
 basis = ANO-S-VDZ; group= c1
&SEWARD
&RASSCF
 Spin= 1
 fileorb=$CurrDir/acrolein.GvOrb
&GRID_IT
 All; ASCII
&CASPT2
```

Verify that the orbitals after CASSCF calculation are correct. Check the log file. Identify Reference energy, CASPT2 energy, and Reference weight.

We now perform multi-state CASPT2 calculation for all 5 roots:

```
&RASSCF
 Spin= 1
 fileorb=$CurrDir/acrolein.GvOrb
 CiRoot= 5 5 1
&CASPT2
 Multistate= 5 1 2 3 4 5
```

Compare the results of single state CASPT2 and MS-CASPT2.

Ex 8. Optional. Visualization of excited state calculations

create a file acrolein.MS-CAS-grid.input

```
&GATEWAY
 coord = $MOLCAS/Coord/Acrolein.xyz
 basis = ANO-S-VDZ; group= c1
&SEWARD
&RASSCF
 Spin= 1
 fileorb=$CurrDir/acrolein.GvOrb
 CiRoot= 5 5 1
 OutOrbital
 Natural= 5
&GRID_IT
 FILEORB = $Project.RasOrb.1
 NAME = root1; All; ASCII
&GRID_IT
 FILEORB = $Project.RasOrb.2
 NAME = root2; All; ASCII
&GRID_IT
 FILEORB = $Project.RasOrb.3
 NAME = root3; All; ASCII
&GRID_IT
 FILEORB = $Project.RasOrb.4
 NAME = root4; All; ASCII
&GRID_IT
 FILEORB = $Project.RasOrb.5
 NAME = root5; All; ASCII
```

RASSCF code produces individual orbital files RasOrb.1 - RasOrb.5, and for each of these files we run GRID_IT code, and use keyword NAME to give different names for the grid files.

Now, we can visualize the difference between grids, computed for different root.

Use the command:

```
gv.exe acrolein.MS-CAS-grid.root1.grid -a -1.0 acrolein.MS-CAS-grid.root2.grid
```

to get the difference between first grid (the ground state) and the second grid (first excited state).

Red color corresponds to an excess of electrons.

Hint: use +/- buttons to change the shown iso-value.

Ex 9. If everything above was too trivial...

Compute Ionization potential for NO molecule.

For simplicity, use the same interatomic distance for NO and NO^+ : 1.1509 Å, use basis set ANO-S-VDZP.

Questions:

- What is the spin state of NO? What is the spin state of NO^+? What to do if you don't know the answer?
- What would be an appropriate orbital spaces? How can you verify that

statement?

The correct answer 9.26 eV (Hubert&Herzberg)

Dissociation of trans-butadiene molecule

If we would like to study dissociation of butadiene molecule into two vinyl radicals, we also have to break C-C bond. It means, that for our active space we have to add corresponding σ bond. Identify σ and σ^* orbitals, and include them into active space.

Hint. To study a reaction, we need an active space which covers both: reactants and products.

Study the dissociation of trans-butadiene molecule.