

Practical Lesson 1: Molecular Builders & Single Point Calculations

Available builders on Wolf cluster

- Avogadro
 - + fast
 - + free
 - + forcefield preoptimization
- Gabedit
 - + aligning structures in coordinate system
 - - relatively slow
- Gaussview
 - + generating specific distances/angles/dihedrals
 - + aligning molecules
 - - commercial
- Nemesis
 - No idea

- The most widely used in QM are XYZ and Z-matrices
- Cube files:
 - Grid of points with specific values of a given quality
 - Electron density, Electrostatic potential, Laplacian of el. density...
 - Generated from Gaussian wavefunction
- Wavefunction files:
 - Gaussian: (Formatted) checkpoint
 - ADF: TAPES
 - Turbomole: mos

Practical task I: Gaussian

- Prepare input file for calculations:
 - Most builders can generate the file for you (usually has to be edited)
 - You can write it yourself from scratch
 - General suffix is either “.com” or “.gjf”
 - Use:
 - 2 cores
 - 3 GB of memory
 - Save the wavefunction
- Gaussian manual:
http://www.gaussian.com/g_tech/g09ur.htm
- Add gaussian module
- *g09* input.com

- Read the logfile, see the structure, extract information
- *qmutil*: nifty module to extract data from gaussian output:
 - *extract-gopt-ene* logfile
 - *extract-gopt-xyz* logfile
 - *extract-gdrv-ene* logfile
 - *extract-gdrv-xyz* logfile
 - *extract-xyz-str* xyzfile framenumbers
 - *extract-xyz-numstr* xyzfile

Gaussian checkpoint

- Stores wavefunction in binary
- Convert into ASCII file:
 - `formchk -3 file.chk`
- Can be read by gaussview ¹
- Analysis of orbitals, electron density
- Export into cubefiles (ASCII grid files)

¹A bug in Gaussview: Change word “independent” to “independant”.

Practical task II: Turbomole

- Turbomole is probably the fastest code available here (for Gaussian-type basis functions)
- Tmolex as GUI (licence not available)
- RI-J approximation of coulombic term - extremely fast (meta)GGA SCF convergence
- Interactive preparation of the input using *define*
- Turbomole manual: `http://www.turbomole-gmbh.com/turbomole-manuals.html`

Preparing the job: Define

- First two items can be skipped
- Molecular geometry:
 - **a coord** Reads in the geometry
 - **ired** Generates internal coordinates
 - * Proceed to next stage
- Basis set:
 - **b all def2-SVP** Assign this basis set to all atoms
 - * Proceed to next stage
- Method
 - **eht** Perform initial guess from Extended Hückel Theory
 - *Accept all defaults*

- Method
 - **dft** Enter the DFT submenu
 - **on** Use DFT
 - **func b-lyp** Select the functional
 - **grid m5** Increase the gridsize to m5
 - * Exit the submenu
 - **ri** Enter the RI submenu
 - **m** Assign memory for RI
 - **2000** As much as possible
 - **on** Use RI
 - * Exit the submenu
 - **dsp** Use dispersion correction
 - **on** Use Grimme D3 correction
 - * Exit the submenu
 - **marij** Multipole-Accelerated RI-J
- * End the define session

Turbomole job

- For running TM in parallel mode use the parallel build
- *module add turbomole:7.00:x86_64:para*
- Infinity selects it by default if `ncpu > 1`

```
mnovak@wolf
```

```
#!/bin/bash
```

```
module add turbomole:7.00
```

```
jobex -ri -c 1024 > dft.out
```

Turbomole output

File	Contents
dft.out	Optimization procedure
energy	Energies of steps
gradient	Gradients of steps
mos	Molecular orbitals
freq.out	Output from aoforce program

Practical task III: ADF

- Only software using Slater-type orbitals
- Up to 2-component relativistic effects (ZORA+SpinOrbit)
- Awful output file structure
- Extremely fast and efficient
- ADF GUI: adfview
- Very bad memory handling

Preparing ADF input

- Using GUI (the easiest way)
- Write from scratch
- Keywords in blocks:

ATOMS*	definition of geometry in xyz
SYMMETRY NOSYMM	Switch off all symmetry
XC*	DFT functional
BASIS*	Basis set
SAVE TAPE21	Save wavefunction
NOPRINT LOGFILE	Do not print input into logfile

* Section terminated by **END** keyword

- Manual pages: <http://www.scm.com/Doc/Doc2014/ADF/ADFUsersGuide/page1.html>

- `adf < input.adf > output.out`