
**Hands-On Tutorial on
Ab Initio Molecular Simulations
Berlin, July 12 – 21, 2011**

**Tutorial I: Basics of Electronic-Structure Theory
Manuscript for Exercise Problems**

Prepared by Viktor Atalla, Oliver Hofmann, Sergey Levchenko
July 12, 2011

Contents

1	A guideline through the tutorial	2
2	Basic electronic structure with FHI-aims	6
3	Local structure optimization	9
4	Visualizing electron densities and eigenstates	14
A	Sample bash scripts	16
B	Short introduction into Molden	18
C	Figures	23

1 A guideline through the tutorial

The aim of tutorial 1 is to give a basic introduction to electronic structure calculations by solving representative exercises. As every DFT-code has its own philosophy, this tutorial should also familiarize you with fundamental aspects using FHI-aims. For every exercise, we also provide solutions, including sample input files. However, we strongly recommend to use the provided `control.in` and `geometry.in` only in case of time problems and instead to try to generate the input files on your own. For the tutorials an executable of FHI-aims will be provided on your workstation.

The very basics of FHI-aims

Each calculation should be done in a separate directory containing the two mandatory input files `control.in` and `geometry.in`. FHI-aims is then simply called in this directory. In short:

- **Each calculation one directory**
- **2 input files**
`control.in`
`geometry.in`
- **Start calculation**
`mpirun -np 4 aims.hands-on-2011.scalapack.mpi.x | tee calculation.out`

The above starts an parallel calculation on four processors. The `tee` command allows to show the output on the screen while also logging it to the file `calculation.out`. Please note that for longer runs, this might slow the calculation down somewhat. However, this is not problematic for this tutorial.

The output file contains the basic information and results of the calculation such as the total energy, atomic forces, etc.. Additional output files might be generated according to the specified settings.

geometry.in

The `geometry.in` file contains all information concerning the atomic structure of the system. This includes the nuclear coordinates, which are specified by the keyword `atom`, followed by cartesian coordinates (in units of Å) and the descriptor of the species. For example, the input

```
atom 0.0 0.0 0.0 H
```

specifies a single hydrogen atom at the origin.

control.in

This file contains all other settings for the calculation. In particular, it specifies the physical and technical settings for the equations to be solved. A minimal example, which can be used as a template during the tutorial, would be

```
xc          hf
charge      0
spin        collinear
default_initial_moment hund

sc_accuracy_eev  1E-2
sc_accuracy_rho  1E-4
sc_accuracy_etot 1E-5
sc_iter_limit    300
```

In this example, the following options are set:

- `xc`
This keyword sets the method to be used. Here, it is set to `hf`, which requests a Hartree-Fock calculation.
- `charge`
Set the total charge of the system in units of $|e|$. For a neutral system, this is zero.
- `spin`
This keyword governs the spin treatment. It can be set to `none`, which requests a spin-restricted (unpolarized) calculation, or to `collinear`, which requests a spin-unrestricted (polarized) calculation.
- `default_initial_moment hund`
Sets the initial spin of the atoms. The value "hund" requests an initial moment taken from Hund's rule. Only necessary for `spin collinear` calculations.

The other lines set the convergence criteria of the self-consistent cycle regarding the change of the density (`sc_accuracy_rho`), the sum of eigenvalues (`sc_accuracy_eev`), and the total energy (`sc_accuracy_etot`) between two consecutive cycles.

The remaining block of `control.in` defines all computational parameters associated with the elements (species) - most important the basis set. Each element listed in the `geometry.in` must also be listed in `control.in`. The order of the listing is irrelevant.

FHI-aims is shipped with pre-defined settings for all species. Defaults are provided for three different levels of accuracy, light, tight, and really tight. They can be found in the directory `/usr/local/fhi-aims/aimsfiles.hands-on-2011/species_defaults/` and should be copied and pasted into `control.in`, e.g. via the command

```
> cat /usr/local/fhi-aims/aimsfiles.hands-on-2011/species_defaults/really_tight/01_H_defaults  
>> control.in
```

which pastes the `really_tight` settings of the H-atom into the `control.in` file. The basis functions associated with a given species are tabulated at the end of these default settings:

```
# "First tier" - improvements: -1014.90 meV to -62.69 meV
  hydro 2 s 2.1
  hydro 2 p 3.5
# "Second tier" - improvements: -12.89 meV to -1.83 meV
#   hydro 1 s 0.85
#   hydro 2 p 3.7
#   hydro 2 s 1.2
#   hydro 3 d 7
# "Third tier" - improvements: -0.25 meV to -0.12 meV
#   hydro 4 f 11.2
#   hydro 3 p 4.8
#   hydro 4 d 9
#   hydro 3 s 3.2
```

Each line denotes a specific basis function. Additional basis functions can be included in the calculation by uncommenting the corresponding lines. The basis functions are classified in "tiers". Systematically improved calculations can be performed by enabling additional tiers one after another.

Additional tools and programs

Scripts: For some exercises, scripts are required for dedicated tasks, such as the manipulation of cube files. All scripts that are needed for this tutorial can be found in

`/pub/tutorial1/utilities`

Visualization tools: To visualize structures, vibrational modes, charge density plots, etc., several programs including `molden`, `vmd`, `jmol`, `gdis`, and `XCrysden` are installed on your work stations. In this tutorial we will use `molden` as visualization tool. A brief introduction on how to use `molden` can be found in the appendix B of this handout.

Plotting and Editing: To plot data, `gnuplot`, `xmgrace` and `qtplot` are provided. Also several editors are available, including `gedit`, `emacs`, `vi`, and `kwrite`. The simplest one to handle is probably `gedit`.

A quick summary of the exercises

The exercises will be divided in three parts.

The goal of the first section is to get you familiar with the basic inputs of FHI-aims and to demonstrate that DFT calculations have predictive power for many observable quantities. The second part will guide through the geometry optimization of a molecule and point out how to assure that a reliable result is obtained. The third part is dedicated to visualizing DFT results. Some exercises are marked with a red exclamation mark (!). These exercises demonstrate pitfalls or limitations of the shown approach.

2 Basic electronic structure with FHI-aims

Exercise 1 (Getting started - the hydrogen atom). In this exercise, the basics of FHI-aims will be conveyed. This will be done on the simplest system possible, the hydrogen atom, where the exact solution is known. You will see how various computational methods compare to each other and to the exact solution. From a technical perspective, you should learn how to generate input files, read the standard FHI-aims output, and perform basis set convergence tests.

1. Generate a simple `geometry.in` file by hand, which contains only a single hydrogen atom. The input should look like:

```
atom 0.0 0.0 0.0 H
```

2. Generate a simple `control.in` file by hand. Request a Hartree-Fock calculation of an uncharged, spin-polarized hydrogen atom:

```
xc hf
charge 0.0
spin collinear
default_initial_moment hund
```

Additionally, don't forget to set convergence criteria for the SCF cycle. Throughout the whole tutorial, the following settings can be chosen:

```
sc_accuracy_rho 1E-4
sc_accuracy_eev 1E-2
sc_accuracy_etot 1E-5
sc_iter_limit 300
```

Finally, paste the "really-tight" species data of H into the `control.in` file, e.g. via the command

```
> cat /usr/local/fhi-aims/aimsfiles.hands-on-2011/species_defaults/really_tight/01_H.default
>> control.in
```

3. Now, run FHI-aims:

```
> mpirun -np 4 aims.hands-on-2011.scalapack.mpi.x | tee H.out
```

When done, open the output file. If you find the line "Have a nice day" at the end, then your calculation is converged. We are now interested in the total energy. Search for the block

```
| Total energy uncorrected : -0.136054662652353E+02 eV
| Total energy corrected   : -0.136054662652353E+02 eV
| Electronic free energy   : -0.136054662652353E+02 eV
```

Although in this special case, all energies are equal, this is not always the case! For non-metallic systems, as in this tutorial, always use the **Total energy uncorrected** value. Compare it with the exact result for the hydrogen atom (0.5 Hartree \approx 13.6057eV).

TIP: In later exercises, to find this value fast and efficiently, use the command

```
> grep 'Total energy uncorrected' H.out
```

4. Redo the calculation with different basis sets (`minimal`, `tier1`, `tier2`, `tier3`) by (un)commenting the basis functions at the end of the `control.in` file. Plot the total energy as function of the basis set size. Do the results change? From which "tier" on are the numbers converged? **TIP:** To plot the results, simply create a text file (e.g., `file.dat`) with two columns, the number of basis functions and the obtained total energy. This file can be plotted directly using

the command
> `xmgrace file.dat`

5. Finally, compare different methods implemented in FHI-aims by replacing `hf` in `control.in` with

- `pw-lda`
- `pbe`
- `pbe0`

and test the basis set convergence again. Do all methods converge with basis set size? Do all converge to the same result?

Optional part: If you browse through the hydrogen basis set you will note that no hydrogen 1s radial function is included. Change that by adding the line

```
hydro 1 s 1
```

at the end of the `control.in` file. Comment out all other basis functions and run the Hartree-Fock calculation again. How close does it get to the exact result?

Exercise 2 (LiF). One of the first papers which systematically investigated the performance of DFT was Pople *et al*, J. Chem. Phys, 98, 5612, where for a large number of diatomic systems, several different properties were consistently computed and compared to experimental values. In the style of this work, we will calculate the binding curve, atomization energy, and the dipole moment for the simple molecule LiF with various common density functionals. From a technical perspective, this exercise teaches how simple shell scripting can be used to make your (computational) life easier.

1. The first task of this exercise will be to find the equilibrium bond distance of LiF from a series of single point calculations. Start by creating a `geometry.in` file which contains a Li and a F atom, e.g.:

```
atom 0.0 0.0 0.0 Li
atom 0.0 0.0 Dist F
```

In this example, Li is put in the origin and F is located `Dist` Å away from the origin along the z-axis. Hereby, `Dist` is a placeholder which will be replaced by the actual distance later by the script.

2. Create a `control.in` file, and specify a pw-lda calculation for a neutral system. Feel free to copy the file from the previous exercise, but remember to paste the "tight" species defaults for Li and F in. We also want to compute the electronic dipole moment, which is requested using the keyword `output dipole`. Note that although it doesn't matter where you put the keyword, it is recommended to put all manually set keywords *before* the species tags.
3. Next, create a bash-script which, for a series of bond distances (ideally between 1.3Å and 1.9Å with 0.1Å steps, with a denser step-width of 0.02Å between 1.5Å and 1.7Å). For each distance, it should
 - create a unique directory
 - copy the `control.in` and `geometry.in` file
 - replace the bond distance placeholder `Dist` with the bond distance and
 - start FHI-aims.

Appendix A holds a short sample bash script for this task. Alternatively, you may also use the script we provide in

```
/pub/tutorial1/reference/exercise_2_LiF/scripts/run.sh.
```

Run the script and plot the total energies vs. the bond length. This can be scripted as well. An example script is also provided in

```
/pub/tutorial1/reference/exercise_2_LiF/scripts/pp.sh.
```

Which bond length corresponds to the lowest energy?

4. Repeat the bond length determination using pbe, hf, and pbe0. How does the optimal bond length change, how much the total energy? How does the bond length compare to the experimental bond length of 1.56Å?
5. In order to compute the atomization energy, which can then be compared to experimental values, we need the total energy of the free Li and F atom as well. Compute the single atoms with each of the functionals. Check the result carefully - is Hund's rule fulfilled? Does the electronic configuration correspond with what you expect from the aufbau principle? Calculate the atomization energy of LiF by subtracting the free-atom energies from the total energy in the energetic minimum. How does this compare to the experimental value of 137.6kcal mol⁻¹ (5.97 eV)? How much does the atomization energy differ between the methods?
Note: If you experience problems converging the free atom calculations, try adding the keyword
`switch.external_pert 10 safe`
to your control file. It breaks the intrinsic symmetry of an atom by applying an electric field for a given number of scf cycles (here: 10), and then proceeds to converge the system with the field switched off.
6. Finally, let us look at the dipole moment. Search for the corresponding line in the output file. Does the dipole differ much between the methods? How does the dipole at the equilibrium distance compare with the experimental value of 6.33 Debye (1.32 eÅ)? Plot the dipole moment vs. the bond distance. You will find a (mostly) linear correspondence. Do you expect this to go on infinitely? Why?

Exercise 3. (Bonus exercise: O₂) Bonus: [O₂ !] An important part of every calculation is always to critically look at the output and ensure that the result is reasonable. For some systems, defaults may not be adequate, or assumption which commonly work well may prove to be wrong. A prime example is the spin treatment in systems with degenerate orbitals, such as O₂.

1. Set up a calculation similar to the previous exercise for O₂. Leave out the `spin` keyword in `control.in`
2. Using PBE and PBE0, calculate the binding curve in the interval [0.8,1.6] Å with a step-width of 0.1 Å and the atomization energy. Look up the occupation of the Kohn-Sham orbitals. Which spin treatment is chosen by default? Does this make sense here? Is Hund's rule fulfilled?
3. Repeat the calculation with spin-polarized settings - is Hund's rule now fulfilled?

Compare the results: Do both spin settings yield the same equilibrium bond length? Calculate the difference in the total energy at the equilibrium bond length. Which one is lower? How does it compare to the experimental value of 1.0 eV? How does the atomization energy compare to the experimental value of 5.18 eV?

3 Local structure optimization

Exercise 4 (Planar NH₃ !). In this exercise you should learn how to perform geometry optimizations. Specifically we relax the NH₃ molecule starting from planar initial guess for the geometry.

1. Create geometry.in

```
atom  0.0  0.0  0.0  N
atom  0.8 -0.5  0.0  H
atom -0.8 -0.5  0.0  H
atom  0.0  1.0  0.0  H
```

2. Create control.in. Start with a template of the previous exercises (using `spin none`), add the keyword for structure relaxation and use the "light" species defaults for N and H atoms.

```
relax_geometry trm 1E-3
> cat /usr/local/fhi-aims/aimsfiles.hands-on-2011/species_defaults/light/01_H.default
    >> control.in
> cat /usr/local/fhi-aims/aimsfiles.hands-on-2011/species_defaults/light/07_N.default
    >> control.in
```

3. Run FHI-aims.

```
> mpirun -np 4 aims.hands-on-2011.scalapack.mpi.x | tee aims.NH3_planar_relaxation.out
```

4. To visualize the results, copy the tool `create_relax_movie.pl` from `/pub/tutorial1/utilities` into the working directory. Apply it to the output and pipe the result to a new file using the command

```
> ./create_relax_movie.pl NH3_planar_relaxation.out > NH3.molden
Open the file NH3.molden using molden.
> molden NH3.molden
```

How does the fully relaxed structure look like (do you believe this to be the actual energetic minimum)?

Exercise 5 (Vibrational analysis planar NH₃ !). Now we need to check whether our obtained geometry is a saddle point or a real local minimum. This is done by performing a vibrational analysis. This is done by the wrapper `aims.vibrations.hands-on-2011.mpi.pl` in a folder that contains a `control.in` and `geometry.in` file.

1. From exercise 4 use the same `control.in` and the fully optimized planar geometry (from the output of 4 `grep "Final atomic structure"`).

2. Run the vibrational script.

```
> aims.vibrations.hands-on-2011.mpi.pl NH3_planar
(The suffix NH3_planar labels your calculation output)
```

3. Inspect the `NH3_planar.vib.out` file. The eigenmodes of this molecule can be visualized via the `troublemaker.pl` tool, which can be found in the folder `/pub/tutorial1/reference/utilities`.

In order to generate the molden-readable file `NH3_planar.molden`, apply it to the `*xyz` file of the output using the syntax

```
> ./troublemaker.pl -xyz2molden NH3_planar.xyz > NH3_planar.molden
```

Visualize your result using molden

```
> molden NH3_planar.molden
```

The script `aims.vibrations.hands-on-2011.mpi.pl` produces several output files - most importantly the `NH3_planar.vib.out` and the `NH3_planar.xyz` file. The `NH3_planar.vib.out` file contains basic information regarding vibrational frequencies, zero point energies and infrared (IR) intensities. The `NH3_planar.xyz` file contains in addition geometrical information about the vibrational modes and can be read by visualization programs.

If the Hessian is positive definite (i.e. has only positive eigenvalues), the structure is in a local minimum. If not we found a saddle point. Formally, this corresponds to imaginary vibrational frequencies. However it is also common to output them as negative frequencies. There are $3N-6$ non-zero modes ($3N-5$ for linear for molecules), where N is the number of atoms. Due to numerical errors you will find the 'zero' frequencies to be in the range of $\sim \pm 15 \text{ cm}^{-1}$.

How does the mode with imaginary frequency look like?

Exercise 6 (Getting NH₃ right). An imaginary frequency indicates that the curvature of the PES along that mode is negative. Therefore a reasonable guess would be to distort the geometry along that direction and optimize the structure again. This is what we are going to do now!

1. Taking the `NH3_planar.xyz` file from exercise 5 and distort the planar geometry along the imaginary mode using the script
> `./troublemaker.pl NH3_planar.xyz`
2. For the distorted geometry optimize the structure again (as described in exercise 4).
> `mpirun -np 4 aims.hands-on-2011.scalapack.mpi.x | tee aims.NH3.distorted_relaxation.out`

How does the optimized structure look like?

How does the energy compare with the planar optimized structure of exercise 4?

Exercise 7 (3D NH₃ is local minimum). Check whether the 3D geometry of NH₃ is stable, by performing a vibrational analysis as described in exercise 5.

Is it stable?

How does the lowest mode in energy (which is not a translation or rotation) look like?

Exercise 8 (IR spectrum). Besides information about the structural stability a vibrational calculation also gives the IR spectrum. This can be extracted from the `*vib.out` file with the script `extract_harmvib.sh` which is contained in the `utilities` folder of this tutorial. It gives a two column file (where the first 6 frequencies are commented out)

```
> ./extract_harmvib.sh NH3_planar.vib.out > ir_NH3_planar.dat
```

This can be read by plotting tools (e.g. `xmgrace`):

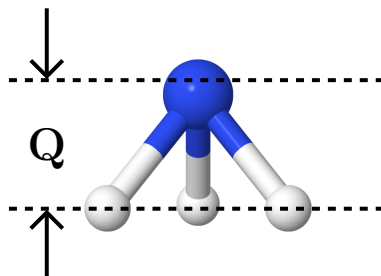
```
> xmgrace ir_NH3_planar.dat
```

In this exercise we will compare the IR spectrum for different numerical settings, different XC functionals and different geometries.

- Perform convergence tests with respect to light, tight and really tight settings for the infrared spectrum and for the 3D geometry (LDA functional).
- For light settings, compute the PBE IR spectrum and compare it with the LDA spectrum.
- Compare to the experimental IR spectrum (<http://webbook.nist.gov/cgi/cbook.cgi?ID=C7664417&Units=SI&Type=IR-SPEC&Index=1#IR-SPEC>) and compare the results.
- **Bonus:** for light settings compare the IR spectrum for planar and 3D geometries (LDA functional).

Important: since a vibrational analysis makes only sense for an optimized geometry, make sure that you require a geometry optimization in your `control.in` file, i.e. set `relax_geometry trm 1E-3`. In this case, the vibrational script will optimize the geometry before doing the actual frequency calculation, i.e., you do not need to perform a separate geometry optimization task.

Exercise 9 (Limits of the harmonic approximation !). In order to demonstrate the range of the validity of the harmonic approximation which is the basic assumption for the vibrational analysis, we first restrict the potential energy surface (PES) to one dimension. We consider a cut through the PES where the H atoms remain in one plane and the N moves along the normal of this plane. Denote the distance between the plane of H atoms and the N atom as Q .



- **Calculate the PES by performing constrained structure optimization.**

The PES can be calculated by performing a structure relaxation under the constraint that the H-atoms are fixed within a plane for a given distance Q between N and that plane. The information about the constraints goes into the geometry.in file, where the keyword has to follow immediately after the corresponding atom: `constrain_relaxation .true.` indicates that the whole atom is fixed during the optimization process in all three spatial coordinates, and `constrain_relaxation z` fixes only the z-coordinate of the atom.

```
atom      0.000    0.000    <Q>    N
constrain_relaxation .true.
atom      0.873   -0.504    0.000    H
constrain_relaxation z
atom     -0.873   -0.504    0.000    H
constrain_relaxation z
atom      0.000    1.008    0.000    H
constrain_relaxation z
```

Take the above structure as template for a set of constrained relaxation calculations. Compute the PES by varying Q from 0 to 1.2 Å with a step-width $\Delta Q = 0.04$ Å.

```
> mpirun -np 4 aims.hands-on-2011.scalapack.mpi.x | tee NH3_PES.Q.out
```

Need help scripting ? → Appendix A.

- **Approximate the PES by distorting along the out-of-plane mode relative to the saddle point.**

In particular the structure for $Q = 0$ in the above construction corresponds to the saddle point as we have found it in example 4. Therefore, within the vicinity of $Q = 0$ the PES should be well described by the the out-of-plane normal mode of the planar geometry. The script `troublemaker.pl -mode M -norm N *xyz` distorts the equilibrium geometry along a chosen mode M with a given extension N (which is indicated by the keyword `-norm N`). It has to be applied to a `*xyz` file which we already obtained in the previous exercises. Note that the norm of the mode is not the distance Q between the N atom and the plane of H atoms. For this specific case a distance Q translates into the corresponding norm by $N = Q/0.6966$ (Can you find out this formula by inspecting the corresponding `*xyz` file?).

1. Distort along the out-of-plane mode starting from the planar optimized geometry. In order to make contact with the first part of the exercise, choose the norms in such a way that for the corresponding Q are within a range of 0 and 1.2 Å with a step-width $\Delta Q = 0.04$ Å, (that is N goes from 0 to 1.2/0.6966 Å with a step width $\Delta N = 0.04/0.6966$ Å).

```
> ./troublemaker.pl -mode 1 -norm N NH3_planar.xyz
```

2. Use the output to create a geometry.in file
3. For every geometry obtained in this way calculate the total energy.
Hint: In order to facilitate the comparison of the calculations label your results with respect to the coordinate Q .
`> mpirun -np 4 aims.hands-on-2011.scalapack.mpi.x | tee NH3_planar_Q.out`

Important: Do not perform a structure optimization!

- **Approximate the PES by distorting along the umbrella mode relative to the pyramidal shaped minimum.**

Similar to the saddle point, there exists a Q ($Q = 0.38\text{\AA}$) such that the real minimum geometry is sampled by the above definition of the PES. Therefore, within a certain range of $Q = 0.38\text{\AA}$ the PES should be described by a distortion along the umbrella mode relative to the 3D optimized geometry.

1. Distort along the umbrella mode starting from the 3D optimized geometry. Choose the norms in such a way that Q is between 0 and 0.1\AA with a step-width of 0.04\AA . (In this case a given Q corresponds to a norm N via $N = (-Q + 0.3812)/0.6504$. Why ?)
`> ./troublemaker.pl -mode 7 -norm N NH3_3d.xyz`
2. Use the output to create a geometry.in file
3. For every geometry obtained in this way calculate the total energy.
Hint: In order to facilitate the comparison of the calculations label your results with respect to the coordinate Q .
`> mpirun -np 4 aims.hands-on-2011.scalapack.mpi.x | tee NH3_3d_Q.out`

Important: Do not perform a structure optimization!

Plot the PES as it is obtained from constrained structure relaxations, and its approximations which were obtained by distorting along the out-of-plane/umbrella mode with respect to the planar and non-planar optimized geometry (Figure 1). Take as common reference the total energy of the fully optimized planar structure. Is there a range where the approximation is good?

In the previous exercise we merely *assumed* that in a region near the minimum the reaction along the coordinate Q can be described by a distortion along a vibrational mode in that direction. In the next exercise we want to actually calculate whether this is really true or not. For that we describe a point \mathbf{X} on the PES which we defined in exercise 9 as a zero-point geometry \mathbf{X}_0 (i.e. saddle point or non-planar minimum) plus a linear superposition of vibrational modes \mathbf{M}_i associated to this geometry:

$$\mathbf{X} = \mathbf{X}_0 + \sum_{i=1}^{3N} \lambda_i \mathbf{M}_i, \quad \mathbf{M}_i, \mathbf{X}, \mathbf{X}_0 \in \mathbb{R}^{3N} \quad \lambda_i \in \mathbb{R} \quad (1)$$

with expansion coefficients λ_i and N the number of atoms.

Exercise 10 (BONUS: Mode mixing). For the PES you obtained in exercise 9 by constrained relaxations, find the expansion coefficients λ_i as defined in equation (1). Take as reference geometry the planar saddle point geometry.

1. Extract the final geometries which you obtained by constrained relaxation from the first part of exercise 9 (for Q from 0 to 1.2\AA with a step-width $\Delta Q = 0.04\text{\AA}$). The script `FinalAtomicStructure.sh` which you find in the `utilities` folder of this tutorial does the job, e.g.
`> ./FinalAtomicStructure.sh aims.NH3_PES_Q.out > relaxed_geo_Q.in`

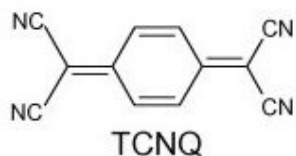
2. For this geometry find the mode expansion coefficients using `troublemaker.pl`

```
> ./troublemaker.pl -project relaxed_geo_Q.in NH3_planar.xyz >  
output_troublemaker_Q.out
```
3. Plot the projection coefficients λ_i as a function of the distance Q . Which modes contribute?
4. Now we can study the energetic contributions of the contributing modes. Similar to exercise 9 perform total energy calculations for the H-stretch and out-of-plane mode. The norms which you need for a given Q are now the expansion coefficients which you just obtained. Where does the H-stretch mode contribute in energy (compare to Figure 2)? Why can we neglect the translational mode for the energetic contribution?

4 Visualizing electron densities and eigenstates

Exercise 11 (Kohn-Sham-Orbitals of TCNQ). Chemical reactions and many important physical effects (such as the formation of interface dipoles) are triggered by electron rearrangements. These are often associated with filling or emptying molecular frontier orbitals. In this exercise, we will demonstrate how charge distributions are visualized and investigate how well the "actual" filling of a molecular orbital corresponds to the associated eigenstate of the neutral molecule.

1. The molecule we will be working with is **7,7,8,8-tetracyano-1,4-quinodimethane** (TCNQ, see Figure), a strong electron acceptor which has recently gained much attention. You can either create the geometry input for this molecule yourself (see the appendix on how this can be done with `molden`) or use the provided input file from the Templates folder.



2. Create a `control.in` input file for a spin-unpolarized, uncharged PBE calculation of the molecule. We want to visualize the highest occupied molecular orbital (HOMO) and the lowest unoccupied molecular orbital (LUMO) in form of a standard cube file. TCNQ contains 104 electrons (6 per carbon, 7 per nitrogen, 1 per hydrogen atom), with two electrons per orbital (with opposite spin). Request the output of the wave functions using the keywords
`output cube eigenstate 52`
`cube filename HOMO.cube`
`output cube eigenstate 53`
`cube filename LUMO.cube`

We also want to look at charge density differences, so we need the total density of the molecule. Add

```
output cube total_density
cube filename total_density_uncharged.cube
```

to the `control.in` file. By default, FHI-aims will figure out the region to plot into the cube files by itself.

However, you are also free to choose the cube region yourself by adding the following lines after the `output cube` keyword:

The line

```
cube origin x y z
```

allows to determine the center of the space to be plotted, with `x y z` being its cartesian coordinates in Å. The edges of the cube file can be configured using the syntax

```
cube edge n dx dy dz,
```

where `n` indicates the number of steps of a particular edge (voxel), and `dx`, `dy`, `dz` indicate the length of each individual step in `x`, `y`, and `z` direction, respectively. Note that each cube file, being a 3D-object, requires 3 edges to be completely defined. E.g., a complete definition of a cube file could look like

```
output cube total_density
cube filename total_density_uncharged.cube
cube origin 0 0 0
cube edge 100 0.1 0.0 0.0
cube edge 150 0.0 0.1 0.0
cube edge 50 0.0 0.0 0.1
```

3. Run the calculation and look at the output. In the final occupation numbers, search for the energies of the HOMO and the LUMO. How do these energy compare to the experimental values of 9.5 eV for the ionization potential and 2.8 eV for the electron affinity? Next, use your favorite tool to plot the cube files. (See appendix on how this is done with molden.) Do HOMO and LUMO both have π character?
4. To see how well the eigenvalues and eigenstates correspond to the physical situation of charging a molecule, prepare and run two more inputs for the molecule with one negative charge. Remember that the total number of electrons is odd, and therefore, `spin collinear` should be used.

Obtain the electron addition and removal energies by subtraction the total energy of the charged system from the total energy of the uncharged system. (This is also known as the ΔSCF approach, which you will encounter again in Tutorial 6). Compare the values to the LUMO eigenvalue of the neutral molecule. How much do they differ?

5. The change in the electron density upon charging can be obtained by subtracting the total density of the charged system from the total density of the uncharged system. To that aim, a python-script called 'subtract_cubes.py' is provided in the utilities folder. Copy this script and the total_density.cube files into a seperate folder and execute

```
> python subtract_cubes.py total_density_uncharged.cube  
    total_density_charged.cube chargediff.cube
```

to subtract the file given as second argument from the file given as first argument. The result will be written to the file provided as final argument. If you changed the geometry of the cube files, please be aware that all files must have the same specification!

Note that the cube files need to be generated with the same cube file settings in `control.in`. Visualize the density differences. Do you see qualitative differences to the eigenstates? Can you explain (or at least speculate) on the the reasons for them?

A Sample bash scripts

Sample script for Exercise 2

Below, you find a sample bash script for calculating a molecule with different pre-defined bond distances.

```
#!/bin/bash

# Define the bond distances
BondDistance='1.3 1.4 1.5 1.52 1.54 1.55 1.56 1.57 1.58 1.59 1.60 1.61 1.62
1.64 1.66 1.7 1.8 1.9'

#Cycles thorough all distances
for Dist in $BondDistance; do
    mkdir $Dist #Creates the directory
    cp geometry.in $Dist/geometry.in; #copy geometry.in
    cp control.in $Dist/control.in #copy control.in file
    cd $Dist # Changes directory
    #Search / replace Dist with Bond distance
    sed -i "s/Dist/$Dist/g" geometry.in
    mpirun -np $NP $AIMS_BIN > Ex2_$Dist.out #call aims
    #Collect output in a file
    grep 'Total energy corr' Ex2_$Dist.out | tail -1 >> ../Energies.out
    cd .. #Changes directory back
done # Loop done
```

Sample script for Exercise 9 (first part)

Below, you find a sample bash that creates geometry.in files from a template "geometry_planar_temp", executes a calculation and writes the total energy with respect to the zero-point energy into the file "PES_energy.dat".

```
#!/bin/bash

# Calculator
python_calc () {python -c "print ($@)"}

for Q in {0..30}
do
    # Get Q
    Q=$(python_calc "0.0 + $Q*0.04")

    # Create geometry file by modifying the template geometry_planar_temp
    cat geometry_planar_temp | sed "s/<Q>/$Q/g" > geometry.in

    # Run aims
    mpirun -np 4 aims.hands-on-2011.scalapack.mpi.x | tee aims.NH3_PES_Q_$Q.out

    # reference-point
    if [ $Q == 0.0 ] ; then
        ZP=$(cat aims.NH3_PES_Q_$Q.out | grep 'Total energy uncorrected' | \
            tail -1 | awk '{printf "%12.9f", $6}')
    fi

    # Read and write energy
    cat aims.NH3_PES_Q_$Q.out | grep 'Total energy uncorrected' | tail -1 | \
```



```
awk '{printf "%2.2f\t%12.9f\n", '$Q', ($6 - '$ZP')}' >> PES_energy.dat  
done
```

B Short introduction into Molden

*"Ich seh' es gern, das steht dir frei
nur dass die Kunst gefällig sei!" - Goethe, Faust - der Tragoedie erster Teil*

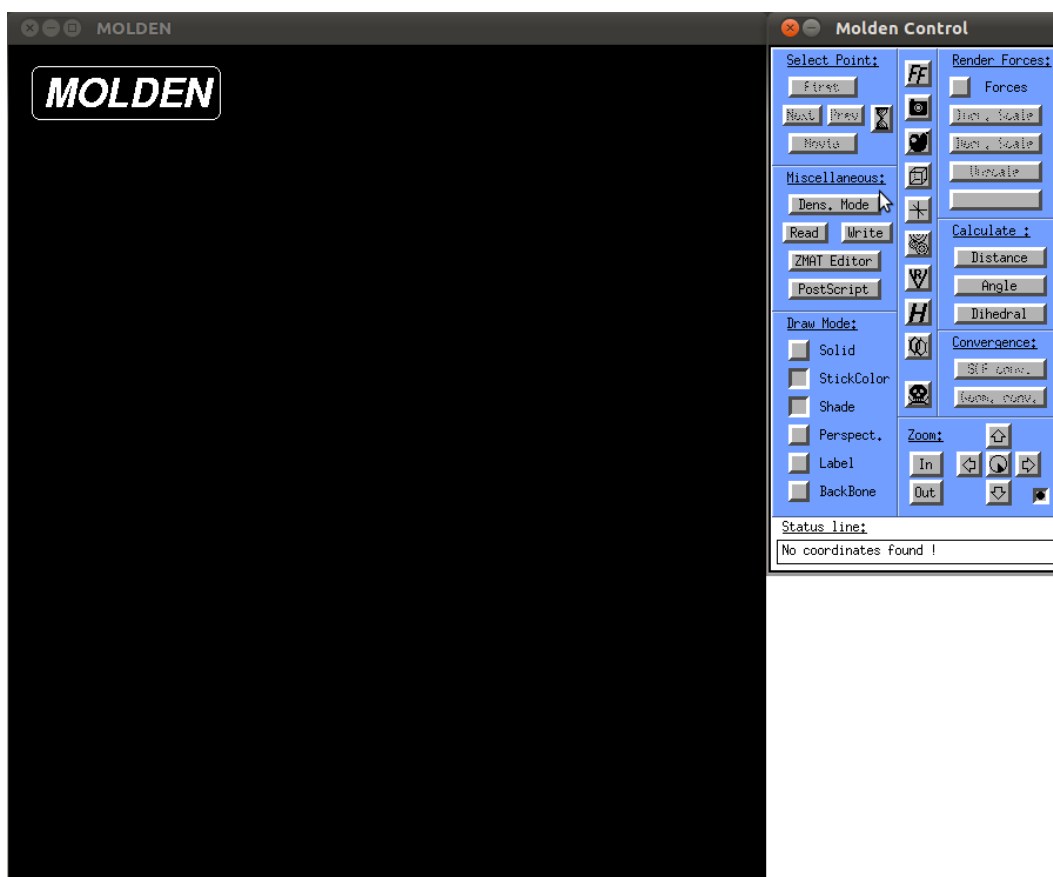
Molden is a small, free tool which allows to visualize non-periodic structures, geometry optimizations, vibrations, and cube files. It can be used for all visualizations during tutorial 1. In the utilities folder, we provide several scripts which convert the output from FHI-AIMS to the molden format.

To open molden, simply type

```
molden or molden filename
```

in the command line.

Two windows will open, one named "MOLDEN", which contains the structure (and is empty if you didn't provide a file), and one named "MOLDEN Control" which contains several buttons. To open a new file, use the "Read" button.



Below, we will describe how to visualize the output from the individual task. Of course, molden can do much more than what is shown here - feel free to play around. If you want to close molden, please use the skull icon in the middle of the control window.

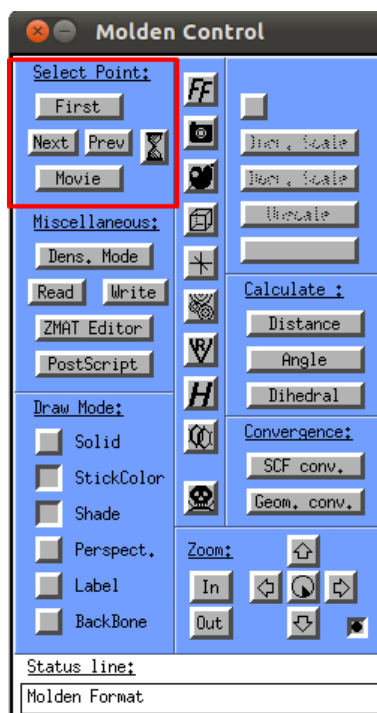
Geometry single points

1. Create a molden file by applying the script `create_relax_movie.pl` to the FHI-AIMS output of a single point calculation
2. Open the file typing `molden filename`. On the top left corner of the Molden Control Window. The geometry of the system should appear in the Molden Window.

3. You can change the display options by playing around with the Draw Options in the bottom left corner of the Molden Control Window. Good results are often achieved by disabling shade and setting "Solid" to "Ball and Stick". Feel free to experiment!

Geometry relaxations

1. Create a molden file by applying the script `create_relax_movie.pl` to the FHI-AIMS output of a geometry relaxation.
2. Open the file typing `molden filename`. On the top left corner of the Molden Control Window, the "Select Point" field should be active now. (See picture)
3. Watch a movie of the relaxation by clicking on the "Movie" button. If the movie is too fast, use the "sand-clock" button to set a delay between the individual frames.
4. You can also walk through each point of the geometry optimization using the "Next" and "Prev" buttons.
5. Clicking on the "Geom. Conv" button in the bottom right corner of the Molden Control window will open a new window with information about the change of energy, maximum force, and maximum step size for each step.

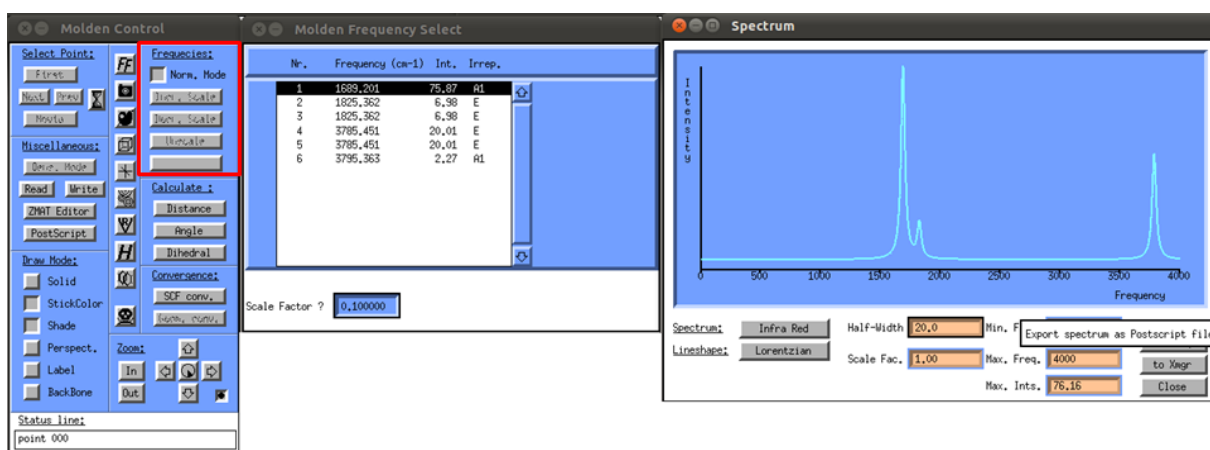


Vibrations

*"Fluch sei dem Balsamsaft der Trauben
 Fluch jener grössten Liebeshuld
 Fluch sei der Hoffnung, Fluch dem Glauben
 und Fluch vor allem
 der Geduld" - Goethe, Faust - der Tragoedie erster Teil*

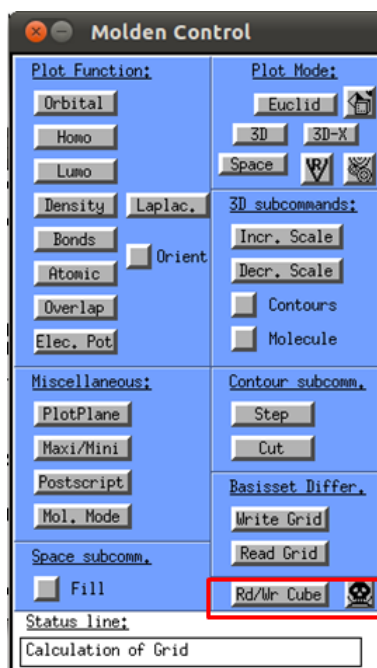
1. Create a molden file by applying the script `> ./troublemaker.pl -xyz2molden NH3.xyz > NH3.molden` to the FHI-AIMS output `NH3.xyz` of a vibration (`troublemaker.pl` can be found in the `/pub/tutorialutilities` folder). calculations.

2. Open the file typing `molden filename`. On the top right corner of the Molden Control Window, the "Norm. Mode" button should be active now, as highlighted in the picture.
3. Click on the button. Two more windows will open. The window called "spectrum" will provide you with an simulated infra-red spectrum based on the eigenvalues and their intensity. The bottom of this window provides several possibilities to alter the appearance of the spectrum or print it. The second window is called "Molden Frequency Select" and contains a list of all eigenmodes along with their frequency and IR intensity. By choosing one of these, the geometry in the molden window becomes animated and displays the corresponding eigenmode. The box scale factor at the bottom of the window can be used to amplify or damp the movement of the atoms.



Cube files

1. Cube files are generated directly by FHI-AIMS using the output `cube` syntax, see manual section 3.19
2. open molden without any filename by typing `molden`
3. Click on the "Dens Mode" button in the left central panel. The Molden control window will change and show different buttons, see picture.
4. Click on the "Rd/Wr Cube" button in the bottom right corner. Choose read, Gaussian and give input the file name of your cube file when asked.
5. When the file opens, use the "Space" button in the top right corner to chose an isovalue for the display of the cube file.



Creating geometries

*"Haett ich nur sieben Stunden ruh,
bräucht den Teufel nicht dazu ... "* Goethe, Faust - der Tragoedie erster Teil

Besides pure visualization, molden also allows to efficiently generate and save geometries for molecular structures. Molden uses a z-matrix editor. A detailed explanation of z-matrices is beyond the scope of this work, but can be found online, e.g. under <http://www.cmbi.ru.nl/molden/zmat/zmat.html>. Even for people not familiar with z-matrices, generating input structures is usually easy and intuitive. This function is briefly explained here for the example TCNQ, the molecule used in the last part of Tutorial 1, and shown in the picture below.

1. Open molden and go to the z-matrix editor. A new window will appear which contains an empty field and several buttons, as shown in the picture below.
2. Begin by clicking on the "Add Line" button and select a carbon atom. A green circle should appear on your screen now.
3. The fastest way to proceed is to mark the carbon atom, choose 'Substitute atom by fragment' from the Zmatrix Editor, and select -Phenyl. This will turn the atom into a completed ring, saturated with hydrogen atoms on all carbon atoms but one. The formally empty box will now contain a list of all atoms. The first row is for bond distance, the second row is bond angles and the fourth row is dihedral angle.
4. To add additional atoms, select the "Add Line" button again. Choose "double" as bond length, and a carbon atom from the periodic table. Molden will now ask you to select 3 Atoms to define the connectivity. These are the atoms to which bond length, angle and dihedral angle are defined. Click on the only carbon atom which has no hydrogen on it, and subsequently on 2 neighboring carbon atoms. A new line should appear.
5. If you made a mistake and want to delete the atom, click on the "delete line" button. If you were successful, connect two more carbon atoms to the carbon atom you just added, and finally, add two nitrogen atoms (triple bond) to the outermost carbon atoms. By now, 2/3 of the TCNQ molecule should be done. To add the "arms" on the other side of the molecule
6. as well, choose the appropriate hydrogen atom and click on the "delete line" button. Add the remaining carbon and nitrogen atoms as in the previous steps. Once you are done, save the geometry by clicking on "Cartesian" in the lower right corner of the Zmatrix Editor, choose "XYZ" and provide a file name. Save by pressing return, and quit molden via the Skull icon.

7. Use an editor or an script to adjust the xyz-format to the FHI-AIMS geometry .in format.



C Figures

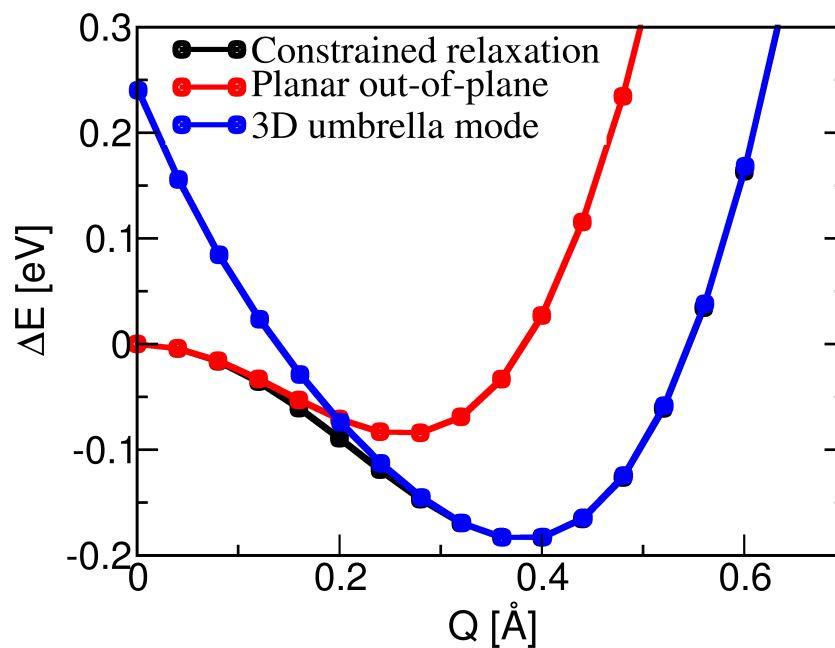


Figure 1: **Ad Exercise 9.** PES as obtained from constrained relaxation calculations (black) and distortions along the out-of-plane (red) / umbrella (blue) modes of the equilibrium geometries. Q denotes the distance between the plane of H atoms and the N atom. In a neighborhood of the minima the PES is sufficiently well described by the modes only. The energy zero was chosen to be the total energy of the saddle point geometry.

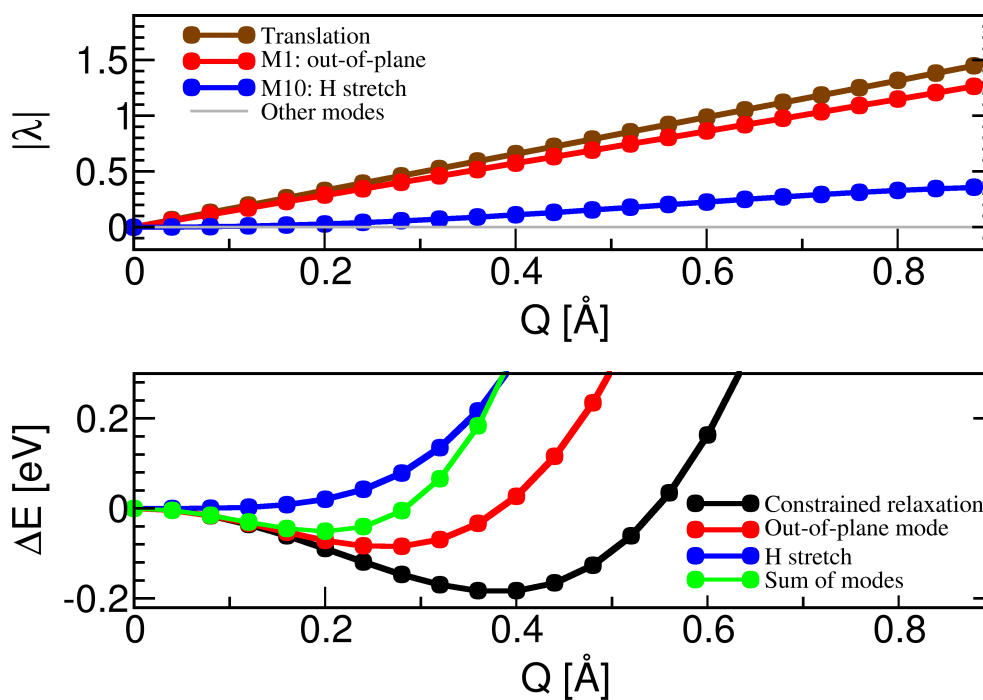


Figure 2: **Ad Exercise 10.** Top: expansion of the geometry of the PES with respect to the modes of the planar NH_3 molecule. Bottom: energetic contributions of these modes.