Theoretical Material Science:
Electronic Structure Theory at the Computer

Prepared by Hagen-Henrik Kowalski,
based on an exercise by Cristian Carbogno
Berlin, June 2017

Some rules on expected documentation from this exercise.

The computational exercises are intended as "hands-on" experience with actual, numerical electronic structure theory. Our main goal is to fill some of the basic concepts with life for real systems.

If you are fast, you may finish the exercise below in the actual exercise class. If not, we ask you to try finish as much as you can, and then come back at another point, follow the script, and finish the exercise.

You can do the actual computations and hand in your solutions to the exercise in pairs of two. What is expected is a record of the basic data that we ask for (e.g., in table form), rough answers to the questions asked (answers can be short, but should be there, and should indicate that you understood the meaning of your data), and plots, where required. You can just qualitatively draw the plots by hand in your homework or print them on paper or screenshot them and then hand-in your homework electronically via email as PDF.

Please hand in all your solutions as usual at the beginning of the exercise in the following week.

Contents

1 Important Notes

- This exercise should be performed using exclusively the generalized gradient approximation (GGA) to density-functional theory. To be precise **choose** the **PBEsol** functional. Do not choose the local-density approximation (LDA) by accident.

- Throughout this exercise, use only "light" settings for the species defaults of FHI-aims. This will suffice to demonstrate the principle. In a complete scientific project, one would want to verify the key results with converged "tight" settings, but this is not required here due to the time involved.

- Please closely inspect the example for the control.in file given in Sec. 5.1. It contains several new keywords. You may use it as a template

2 Background

In this exercise, we will use the structure predictions of exercises 14 and 15 as a basis to investigate the vibrational properties of diamond silicon.

In the present exercise, we will use the forces calculated in Kohn-Sham density-functional theory (DFT-GGA) to approximate the potential-energy surface (PES) in a second order Taylor expansion around the equilibrium volume,

$$V^{\mathrm{BO}}(\{\mathbf{R}\}) \approx V^{\mathrm{BO}}(\mathbf{R}_0) + \frac{1}{2} \sum_{\alpha\beta,IJ} \underbrace{\frac{\partial^2 V^{\mathrm{BO}}(\mathbf{R}_0)}{\partial R_I^\alpha \partial R_J^\beta}}_{\Phi_{IJ}^{\alpha\beta}} s_{\alpha,I} s_{\beta,J}. \tag{1}$$

Here the $\Phi_{IJ}^{\alpha\beta}$ are the force constants (or Hessian), $\mathbf{s}$ the displacement vectors, and V^{BO} the Born-Oppenheimer surface.
We already know from the lecture that, strictly, this "approximation" will be limited in reach as it does not account for anharmonic effects.

More accurate methods exists such as molecular dynamics but these are usually exceedingly expensive. Accordingly the methods used in this exercise are frequently employed in the solid state community to date.

As a reminder below you see the five points that are specifically relevant for the present exercise :

- Γ: Located at (0,0,0) in units of the reciprocal lattice.

- L: Located at (0.5,0.5,0.5) in units of the reciprocal lattice of the primitive fcc cell.

- X: Located at (0.0,0.5,0.5) in units of the reciprocal lattice of the primitive fcc cell.

- W: Located at (0.25,0.5,0.75) in units of the reciprocal lattice of the primitive fcc cell.

- K: Located at (0.375,0.375,0.75) in units of the reciprocal lattice of the primitive fcc cell.

3 IMPORTANT: Phonopy in the Virtual Machine

To calculate phonons in the harmonic approximation we will use the phonopy package but before you can get started some other packages have to be installed. This however is fairly easy, first install pip, h5py, and pyyaml by typing:

```
sudo apt install python-pip
pip install h5py
pip install pyyaml
```

Once these are installed download phonopy from the link
https://thcloud.rz-berlin.mpg.de/s/6FEiHHM3ftK6F8B.
After downloading unpack the file, for convenience reasons you should rename the folder simply phonopy. Before installing phonopy you have to add a line to your python path in your bashrc file (this file can only be accessed from the home folder). This can be opened by typing:

```
vi .bashrc
```

At the end of the bashrc file you have to add the lines:

```
export PATH=/home/theory/phonopy/bin:$PATH
export PYTHONPATH=/home/theory/phonopy/lib/python
```

After saving and exiting the bashrc file do not forget to type:

```
source .bashrc
```

Next change into the folder phonopy and install the package by typing:

```
python setup.py install --home=.
```

Now can run the program simply by typing phonopy-FHI-aims into the terminal.
Later in the exercise you will also need some pre-calculated simulations. They can be downloaded from the link
https://thcloud.rz-berlin.mpg.de/s/3PgxVnmhRMT2LqI.
These folders contain force evaluations with a number of atoms that would not be possible on a local machine let alone on a virtual one.

4   Reminder: FHI-aims in the Virtual Machine

As a quick reminder, here is (again) an overview of the most important pieces needed for FHI-aims:

- To run the code, create the necessary input files (`control.in` and `geometry.in`) in a working directory of your choice.

- The calling sequence for FHI-aims is:

```
aims | tee calculation.out
```

As before, although you should not need it, there is a complete manual (pdf) for the FHI-aims code located in
/home/theory/fhi-aims.160328_3/FHI-aims.pdf .

In this exercise sheet, we will use a *python* script to visualize and plot band structures and densities of states. This requires additional python libraries that you can install with the command:

```
sudo apt-get install python-matplotlib python-scipy
```

For this purpose, you will be requested to enter the password ("theory") and to acknowledge the installation of additional, required packages (just press "Y" when asked for it). If you encounter any problems, e.g. "E: Could not get lock /var/lib/dpkg/lock", reboot the virtual machine with the command: `sudo shutdown -r now`

After reboot, the commands

```
sudo dpkg --configure -a
sudo apt-get install python-matplotlib python-scipy
```

should do the trick. Alternatively, you can just re-download and re-install the updated virtual machine instance that already contains these packages from https://thcloud.rz-berlin.mpg.de/s/kElVkeik4aSfMEQ . Please note that deleting the old virtual machine will also delete the data that you created with it during the previous exercises.

5 Exercise 21: Phonon band Structure and DOS of the Diamond Phase of Si

In the previous two exercises you have learned about the phonon band structure and the phonon density of states. The objective of this task will be to calculate and compare these quantities for a real material namely silicon in the diamond structure in the primitive, the first, the second, and the fourth cubic supercells.

The fhi-aims calculations for the second and forth cubic supercells are given and can be downloaded from the link

https://thcloud.rz-berlin.mpg.de/s/3PgxVnmhRMT2LqI.

5.1 Evaluating the forces with FHI-aims

To complete this exercise, we require the equilibrium lattice parameter a_0 corresponding to the diamond phase of Si. It is important to note that since we changed the functional we cannot reuse the previously calculated equilibrium lattice constant. Instead use a_0=5.4478 Å for this purpose. The **k**-space integration grid for the first cubic supercell should be chosen as $4\times4\times4$ which is qualitatively sufficient. This means before starting FHI-aims we have to change the geometry.in and control.in files.

As usual, it is a good idea to generate a directory

```
 mkdir Si_diamond_phonon
```

and enter the directory with:

```
cd Si_diamond_phonon
```

We recall the geometry input file `geometry.in`:

```
    lattice_vector      0.0 <0.5*a_0> <0.5*a_0>
    lattice_vector      <0.5*a_0> 0.0 <0.5*a_0>
    lattice_vector      <0.5*a_0> <0.5*a_0> 0.0


    atom_frac    0.0 0.0 0.0 Si
    atom_frac    0.25 0.25 0.25 Si
```

As a check, do not forget to visualize the structure:

```
jmol geometry.in &
```

and (in jmol) use the right mouse buttom and choose from the menu the category

```
Symmetry
```

and then

```
Reload:  {444 666 1} .
```

We next recall our computational choices in `control.in` and add further settings related to the calculation of the phonon band structure and phonon DOS in FHI-aims:

```
# Physical settings
    xc          pbesol

# SCF settings
    sc_accuracy_eev   1E-2
    sc_accuracy_rho   1E-4
    sc_accuracy_etot  1E-4
    sc_accuracy_forces  1E-4
    sc_iter_limit     40
# Mixer settings
    mixer pulay
    n_max_pulay 30
    charge_mix_param 0.05

# Occupation settings
    occupation_type gaussian 0.01

# k-grid settings
    k_grid   4 4 4
# Matrix for the construction of the supercell
    phonon supercell -1 1 1 1 -1 1 1 1 -1

# Phonon density of states
    phonon dos 0 800 800 3 40

# High-symmetry q-points for diamond band structure output
    phonon band 0.0 0.0 0.0 0.0 0.5 0.5 100 Gamma X
    phonon band 0 0.5 0.5 0.25 0.50 0.75 100 X W
    phonon band 0.25 0.50 0.75 0.375 0.375 0.75 100 W K
    phonon band 0.375 0.375 0.75 0 0 0 100 K Gamma
    phonon band 0 0 0 0.5 0.5 0.5 100 Gamma L
```

To form a better understanding of these new keyword we will go through them one by one.

5.1.1 Generating the necessary displacements and evaluating the forces

As you know by now, a bulk system does not only consists of N atoms within the unitcell but of a infinite number of periodic images of said cell. However the Hessian does decay quickly with respect to the distance $\mathbf{R}_{IJ}$ between two atoms. Accordingly a representation using only finite sized supercells is sufficient.
The size of this supercell is determined using the keyword "phonon supercell". It defines a matrix

that transforms the lattice vectors of the unitcell into a different cell.

For example choosing

```
phonon supercell 2 2 2
```

means a using a supercell matrix which creates a supercell in which each lattice vector is doubled. Conversely

```
phonon supercell -1 1 1 1 -1 1 1 1 -1
```

transforms a fcc lattice into a cubic lattice.

Similar to an electronic structure calculation the convergence of the vibrational properties has to be checked carefully with the size of the supercell.

Once you prepared your control.in and geometry.in you can generate the displacements. If you have installed phonopy according to the instructions you just have to type

```
phonopy-FHI-aims
```

into the terminal which generates the folder **phonopy-FHI-aims-displacement-01**.

Change into this folder and run fhi-aims. The output has to be named **phonopy-FHI-aims-displacement-01.out**

5.2 Evaluating the phonon band structure and phonon DOS

Once the force calculation has been finished you can obtain the band structure and the density of states in post processing, no further aims calculations are needed. To obtain the band structure and density of states only minor changes to the control.in file are necessary.

5.2.1 Calculating the band structure

To output the band structure in all of the exercises below, the control.in file now contains a section

```
phonon band 0.0 0.0 0.0 0.0 0.5 0.5 100 Gamma X
phonon band 0 0.5 0.5 0.25 0.50 0.75 100 X W
phonon band 0.25 0.50 0.75 0.375 0.375 0.75 100 W K
phonon band 0.375 0.375 0.75 0 0 0 100 K Gamma
phonon band 0 0 0 0.5 0.5 0.5 100 Gamma L
```

that defines which paths in the Brillouin zone shall be computed and plotted. The naming conventions for reciprocal space that you have encountered in exercise sheet 6 are also used in this case, in spite of the fact that we are now investigating the band structure of phonons (and not of electrons!). Also, the syntax is closely related to the output band tag for the electronic band structure that was introduced in exercise sheet 6. In a nutshell, the line

```
phonon band 0.0 0.0 0.0 0.0 0.5 0.5 100 Gamma X
```

requests the output of $\omega(\mathbf{q})$ on 100 linearly interpolated points along the path leading from $\mathbf{q} = (0, 0, 0)$ to $(0.0, 0.5, 0.5)$.

5.2.2 Calculating the density of states

To evaluate the phonon density of states one simply has to add the line

```
phonon dos 0 800 800 3 40
```

to the control.in file

This requests the calculation of the phonon density of states $g(\omega)$ as defined in exercise 20. For the integration an evenly spaced grid of $40 \times 40 \times 40$ is used (last number in the line), $g(\omega)$ itself will be given on 800 evenly spaced frequency points (third number) from 0 cm^{-1} to 800 cm^{-1} (first and second number).

The third number is due to the approximation made to the delta-function. To evaluate it numerically it is approximated using a gaussian function with a smearing of 3 cm^{-1}.

5.3 Plotting the band structure and the density of states

Phonopy automatically generates plots of the requested properties in form of a pdf file.

However, it also writes the raw data in separate files that end with .dat for example phonopy-FHI-aims-band_structure.dat. To compare the results of different supercells it might be useful to use a program such as xmgrace to plot the raw data instead of using the automatically generated plots. Plotting the raw data of the density of states using xmgrace is straight forward. Simply open the program by typing xmgrace go Data → Import → ASCII and load **phonopy-FHI-aims-dos.dat**. Plotting the band structure using xmgrace is a little bit more tricky. Follow the same path as for the density of states but before importing **phonopy-FHI-aims-band_structure.dat** change "load as" form "Single set" to "NXY".