
**Theoretical Material Science: Electronic structure theory at
the computer**

Exercise Sheet 8: Cohesive properties

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.

This exercise is intended in parts as a live exercise, with the remainder to be completed as **homework**. The calculations can be performed at the “PC pool” during its opening hours.

Please hand in your solutions to the exercise individually, as you would normally do. 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.

For required plots, please use the printer in the PC pool and append them to your exercise materials. As you may know, printed pages usually cost 5 cents per page on that printer. *However*, for the purposes of the exercise, a contingent of free pages has been agreed upon. If you are doing printouts for this class, please mention this to the administrators.

Please note that the page contingent is not gigantic – so be careful and do not print excessively many pages. A few pages at most should be sufficient for the solutions, anyway.

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

Contents

1	Important notes	4
2	Background	4
2.1	Equations of state	4
3	Reminder: FHI-aims at the “PC pool”	5
4	Exercise 19a: $E(V)$ of the diamond phase of Si	5
4.1	Total energies	6
4.2	Cohesive energies	7
4.3	Parameterizing Murnaghan’s equation of state	8
4.4	Automating the process: A shell script	8
5	Exercise 19b: Other phases of Si	11
5.1	(i) Nearest-neighbour distance	11
5.2	(ii) Equilibrium cohesive properties of fcc and bcc Si	11
5.3	(iii) k -grid convergence fcc and bcc Si	12
5.4	(iv) Now which is the stable phase, and at what pressure?	12
6	Exercise 20: $E(V)$ of GaAs and ZnSe	13
6.1	Free atoms	14
6.2	$E(V)$	14
6.3	A final remark	14

1 Important notes

- This exercise should be performed using exclusively the local-density approximation (LDA) to density-functional theory. Do not choose Hartree-Fock theory 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.

2 Background

In past exercises, we have used density-functional theory in the local-density approximation (LDA) to compute first converged total energies of free atoms, and then for the Brillouin zone integration (k -point grid convergence) of a real solid.

The true success of the LDA in solid-state physics is that this level of theory was, for the first time, able to make concrete, reliable predictions for a wide range of properties, starting with *cohesive properties*: The structure, energetics, elastic properties etc. of real materials. For any prediction of further properties especially of experimentally not yet known phases, the correct prediction of “structure” must be the first step. This is why cohesive properties are so important.

As a side note, we remark that there are plenty of areas today where predicting the relative stability of phases at given conditions (pressure, temperature) is an essential step. For instance, we can not access the materials at the core of our own planet, yet every textbook notes which materials are allegedly encountered there: Iron, Nickel, and some impurities. But how can we be sure that this assessment is correct? At the relevant pressures and temperatures, nobody has seen iron in experiment, let alone its phases and state (liquid? solid?).

First-principles calculations of materials at extreme pressure and temperature are in fact a rather current topic of interest today, especially in planetary science. In many aspects, density-functional theory is our only way to obtain results: for instance, the conditions, that hydrogen or helium encounter in Saturn, or (as said above) the compounds that might make up the core of the earth.

We will not address the aspect of temperature today, but we will deal with pressure.

We also recall that the LDA is not the most advanced method today. There is a plethora of other viable developments (generalized gradient approximations, hybrid functionals etc.). Yet, we will restrict ourselves today to illustrate the properties and predictions of LDA: It is *also* important to remember that the electron gas (LDA) is our most important first step: This is the only extended system for which we have viable, generic *exact* results from high-level theories. The development of new methods (especially high temperature) thus continues to be guided what we have learned from LDA, even in density functional theory development today.

For those interested in significantly more background, and very current developments in cohesive properties for materials, there is an interesting set of presentations to be found at the following workshop:

<https://www.ipam.ucla.edu/schedule.aspx?pc=plws4>

If you always wanted to know what goes on in Saturn, how to treat the electron gas at high extremely temperatures, or what is the status of the most accurate Quantum Monte Carlo simulations for Hydrogen, some of these slides contain the best answers to date.

2.1 Equations of state

In thermodynamic equilibrium (at given temperature, pressure or volume, etc.), matter generally follows some kind of *equation of state*.

The quantity that we can calculate most readily is the total energy of a given material at *fixed* structure – unit cell volume, internal atomic coordinates, etc. Happily, there exists an equation

of state (first derived by Murnaghan [1]) that links the total energy to the volume by only a few basic materials parameters:

$$E(V) = E_0 + \frac{B_0 V}{B'_0} \left[\frac{(V_0/V)^{B'_0}}{B'_0 - 1} + 1 \right] - \frac{B_0 V_0}{B'_0 - 1} \quad (1)$$

Indeed, this equation links the energy of a material to its volume. In fact, it is an expansion around the zero-pressure equilibrium volume V_0 , which is characterized by the zero-pressure total energy E_0 . The basic materials parameters involved are:

- The bulk modulus at zero pressure, B_0 ($B = -V \left(\frac{\partial P}{\partial V} \right) = -V \left(\frac{\partial^2 E}{\partial V^2} \right)$)
- The derivative of the bulk modulus at zero pressure, B'_0 ($B' = \left(\frac{\partial B}{\partial p} \right)_T$)

While expression (1) looks complicated at first glance, the beauty of it is that it is actually a rather accurate representation of the equation of state of most materials up to very high compression. The occurrence of B_0 (which is measurable) and its pressure derivative suggests that the elastic properties have been taken account in a linear approximation as a function of pressure, and indeed, this is the case.

However, Eq. (1) is even more useful in practice. Consider the following textbook definition of pressure:

$$p = - \left(\frac{\partial E}{\partial V} \right)_S \quad (2)$$

At least at zero temperature (and more generally, if we do not allow any heat exchange) this means that once we know $E(V)$, we can directly state the pressure that is required to compress a solid to a given volume – or, vice versa, the volume assumed by a given solid at a given pressure.

Finally, since Eq. (1) is a really good approximation, we can compute $E(V)$ in practice. All we need are some specifically calculated total energies $\{E(V_1), \dots, E(V_n)\}$ for a few different volumes $\{V_1, \dots, V_n\}$. We can then fit $E(V)$ as a continuous curve, and have (in principle) access to the high-pressure conditions found even at the center of the Earth.

3 Reminder: FHI-aims at the “PC pool”

As a quick reminder, here is (again) an overview of the most important pieces needed for FHI-aims at the “PC pool”.

- To run the code, create the necessary input files (`control.in` and `geometry.in`) in a working directory of your choice.
- For the present exercise, be sure to use more than one processor to run FHI-aims:

```
mpirun -np 2 aims.scalapack.mpi.x | tee calculation.out
```

As before, although you should not need it, there is a complete manual (pdf) for the FHI-aims code located in

`/media/public/TFKP_2012/FHI-aims.pdf` .

4 Exercise 19a: $E(V)$ of the diamond phase of Si

In this exercise, we will compute the $E(V)$ curve of diamond Si. To do so, we pick five different volumes of Si around the experimental lattice parameter $a_{\text{exp}}=5.43 \text{ \AA}$. We recommend lattice parameter changes of $\pm 1 \%$ and $\pm 2 \%$ for the actual data points chosen. We have already ascertained that a k -space integration grid of density $12 \times 12 \times 12$ is more than sufficient for accurate total energies of diamond Si.

4.1 Total energies

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

```
mkdir Si_diamond
```

and enter the directory with:

```
cd Si_diamond
```

For the example of the experimental lattice parameter value, we recall the geometry input file `geometry.in`:

```
lattice_vector  0.0 2.715 2.715
lattice_vector  2.715 0.0 2.715
lattice_vector  2.715 2.715 0.0

atom   0.0 0.0 0.0 Si
atom   1.3575 1.3575 1.3575 Si
```

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

```
jmol geometry.in &
```

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

Symmetry

and then

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

As a simplification it is possible to use *fractional* atomic coordinates in FHI-aims (atom position in units of the lattice vectors). For example:

```
lattice_vector  0.0 2.715 2.715
lattice_vector  2.715 0.0 2.715
lattice_vector  2.715 2.715 0.0

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

will also produce diamond Si, but in a more error-proof way.

We next recall our computational choices in `control.in`. We used:

```
# Physical settings
xc          pw-lda
spin        none

# SCF settings
sc_accuracy_eev  1E-2
sc_accuracy_rho  1E-4
```

```
sc_accuracy_etot 1E-5
sc_iter_limit    40
# k-grid settings
k_grid 12 12 12
```

Here, we have chosen the LDA, as well as a $12 \times 12 \times 12$ k -space integration grid as established before.

Finally, we add “light” settings to compute the total energy of Si:

```
> cat /media/public/TFKP_2012/species_defaults/light/14.Si_defaults >> control.in
```

We can now run the calculation, for example:

```
> mpirun -np 2 aims.scalapack.mpi.x | tee Si_diamond_a.5.43.out
```

Then, repeat the calculations for different lattice parameters, for example:

- $0.98 \cdot a_{\text{exp}}$
- $0.99 \cdot a_{\text{exp}}$
- $1.00 \cdot a_{\text{exp}}$
- $1.01 \cdot a_{\text{exp}}$
- $1.02 \cdot a_{\text{exp}}$

At the command line, the “batch calculator” utility ‘bc -l’ will act as a kind of “pocket calculator” to get the required values fast.

4.2 Cohesive energies

For a physically more meaningful equation of state (limit of zero energy for infinite lattice parameter), we are interested not so much in total energies but actually in cohesive energies (here per unit cell, i.e., per 2 atoms):

$$E_{\text{coh}} = E_{\text{tot}} - N_{\text{Si}} \cdot E_{\text{free atom}} \quad . \quad (3)$$

E_{coh} and E_{tot} are taken *per unit cell*, N_{Si} is the number of Si atoms per unit cell, and $E_{\text{free atom}}$ is the energy of a free (isolated) Si atom.

What we need is the free atom reference energy. To maintain consistency throughout this exercise, what we recommend is this:

- Create a separate directory for the free Si atom
- Enter that directory and copy over the `control.in` file from the bulk calculation.
- In the `control.in` file, modify:
 - The `spin` keyword to `collinear`
 - Remove the `k_grid` line
 - Add all basis functions up to and including the “second tier” to the basis set (by removing the comment marks)
- Use the “total energy” value printed in the last s.c.f. iteration as the free atom energy

... and this total energy puts you in the position to compute (e.g., using ‘bc -l’) the cohesive energy per unit cell for each bulk calculation.

Note: The resulting values for the cohesive energy per unit cell should be in the range of -10.6 eV.

4.3 Parameterizing Murnaghan's equation of state

We can finally set out to parameterize Murnaghan's equation of state.

We will not do this by hand, but rather use a program that obtains a fit of Murnaghan's equation of state to a few $E(V)$ data points.

The program we will use is a (compiled) Fortran program available as

```
/media/public/TFKP_2012/bin/murn.v4.x
```

Writing such a program is not an incredibly hard task, but the above will simplify our life.

First, create an input file for 'murn.v4.x'. This file must be called `murn_fit.in` and should look like this:

```
input_volumes_only
print_input
fit_range 30 50 100
data_point <volume_1> <energy_1>
data_point <volume_2> <energy_2>
data_point <volume_3> <energy_3>
data_point <volume_4> <energy_4>
data_point <volume_5> <energy_5>
```

Short explanation, line by line:

- The first line specifies that we include the actual unit cell volume (here, two atoms, i.e., $a_{\text{fcc}}^3/4$).
- The input will be repeated in the output file
- We will write out the fitted values of Murnaghan's equation of state for 100 points between 30 \AA^3 and 50 \AA^3
- The remaining lines are your computed cohesive energies at five different volumes. Replace the `<energy_i>` placeholders by the volume in \AA^3 , and the `<volume_i>` placeholders by the cohesive energy in eV per *two* Si atoms.

If you run: `/media/public/TFKP_2012/bin/murn.v4.x | tee murn.out`

you will receive an output file `murn.out` that (i) includes the final fitted values of the cohesive parameters in Murnaghan's equation of state, and (ii) can be visualized directly using `xmgrace`. Just open `murn.out` using `xmgrace` and you should see the $E(V)$ curve for diamond Si.

Do not forget to record the results:

- By fitting to Eq. (1), determine E_0 , V_0 , and B_0 .
- Compare your results to the following experimentally known parameters ($T=0$): Lattice parameter $a_0=5.43 \text{ \AA}$, $B_0=98.9 \text{ GPa}$, cohesive energy $E_{\text{coh}}=4.63 \text{ eV/atom}$. What are the trends? How do they match the expected properties of LDA?

4.4 Automating the process: A shell script

There are, in fact, *two* valuable lessons from the last steps. First, we can indeed determine the cohesive properties in this way. Second, the process was rather tedious. When looking at other phases, we do not wish to repeat the process over and over again. Much rather, we could have used the following *shell script*, which you may copy-paste and adapt for the remainder of this exercise.

Call the script (for example) `run.murn.sh`, and make it executable using the command `chmod u+x run.murn.sh`.

We repeat the script here for completeness (on the next page) but it is important to realize that we are *not* trying to give a full Unix introduction here. This tool is one that works, and can be adjusted even with very limited understanding of the bash shell that surrounds it. It is not a bad idea in principle to understand its workings, but again: Consider this an example of “learning by doing”. Such examples can be incredible valuable in practice, even before reading the Unix textbook in its entirety.

Use copy-paste (e.g., from a pdf), do not attempt to type this script by hand in its entirety!

```

#!/bin/bash -l
# Murnaghan fit routine for the fcc (A1) lattice

aims_x=/media/public/TFKP_2012/bin/aims.scalapack.mpi.x
MURN=/media/public/TFKP_2012/bin/murn.v4.x

export OMP_NUM_THREADS=1
export MKL_NUM_THREADS=1
export MKL_DYNAMIC=FALSE
ulimit -s unlimited

atom_type_1="Si"
atom_type_2="Si"
lattice_param=5.43
reference_energy_1=$(echo "-7842.76027177" | bc -l)
reference_energy_2=$(echo "-7842.76027177" | bc -l)

cell_volume=0.25 # the fcc lattice has four primitive unit cells

# Start work - FHI-aims calculations
rm data.dat

for scale in 0.98 0.99 1.00 1.01 1.02
do

    a=$(echo "$scale*$lattice_param" | bc -l)
    echo "Lattice parameter: " $a

    b=$(echo "0.5*$a" | bc -l)
    V=$(echo "$cell_volume*$a*$a*$a" | bc -l)

    cat > geometry.in <<EOF
lattice_vector $b $b 0.0
lattice_vector 0.0 $b $b
lattice_vector $b 0.0 $b

atom_frac 0.0 0.0 0.0 $atom_type_1
atom_frac 0.25 0.25 0.25 $atom_type_2
EOF

    mpirun -np 2 $aims_x < /dev/null > Si_diamond.a_$a 2>&1

    total_energy=$(cat Si_diamond.a_$a | grep 'Total energy corrected' | awk '{printf "%25.8f", $6}')
    energy=$(echo "$total_energy - $reference_energy_1 - $reference_energy_2" | bc -l)
    echo "data_point  "$V"  "$energy >>data.dat

done

# Murnaghan fit from here

echo "unit_cell_volume $cell_volume" > murn_fit.in
echo "input_volumes_only" > murn_fit.in
echo "print_input" >> murn_fit.in
echo "fit_range 30 50 100" >> murn_fit.in
cat data.dat >> murn_fit.in

$MURN > murn.out

```

(The actual script ends above this line.) If you look at the script carefully, you should see all the steps reflected that were just done by hand.

5 Exercise 19b: Other phases of Si

Life at high pressure would be no fun if the stable the structure were always the same. Indeed, this is not the case – there are phase changes as a function of pressure. As an example, we will compare the diamond phase to fcc Si and bcc Si.

5.1 (i) Nearest-neighbour distance

We need starting lattice parameters for fcc and bcc Si. Since these are not experimental phases, we need to derive a good starting point some other way. What we do know is the nearest-neighbour interatomic distance in experimental diamond Si. Use this knowledge to derive starting structures for fcc Si and bcc Si with the same nearest-neighbour distance.

5.2 (ii) Equilibrium cohesive properties of fcc and bcc Si

Our starting structures may still turn out to be pretty bad guesses for the actual structure that we encounter. To head off that risk, we proceed as follows for each phase:

1. We begin with a slightly wider scan about the nearest-neighbour based “first estimate”: again, using five points per scan, but in a spacing $\pm 4\%$, $\pm 2\%$, and 0% about the “first-guess” lattice parameter.
2. We then use the new estimate for the equilibrium volume to derive an updated guess for the equilibrium lattice parameter. If not already within our original range, we perform another $E(V)$ scan with five values about the new estimate.

control.in:

Before we begin, there is one more risk we must head off. We have no guarantee that the fcc or bcc phases are semiconductors, like diamond Si. Instead, they could be metals. In that case, there would be a Fermi level with fractional occupation numbers and possibly higher-lying states with a finite occupation. To hedge against that risk, we can make the following choices:

- Add a finite “smearing” (broadening of occupation numbers) around the Fermi level. Add the following line to `control.in`:
`occupation_width gaussian 0.1`
- Include a larger number of Kohn-Sham orbitals than the default to the calculation above the Fermi level. Add the following line to `control.in`:
`empty_states 100`
- Continue to use a $12 \times 12 \times 12$ k -point grid as a reasonably safe choice, but check the convergence for a single geometry (in the next subtask, (iii)).

Changes to the script run.murn.sh:

By all means, use (and modify) the automated script `run.murn.sh` from the previous exercise to run your calculations automatically. In particular, make the following changes:

- Change the reference value `lattice_param` around which you are scanning $E(V)$ to the appropriate new one for fcc or bcc.
- For the bcc phase only, the `cell_volume` parameter needs to be set to 0.5 (the primitive cell has half the volume of the conventional bcc cell)

- The list of “scale” parameters (the list of lattice parameter values around the central value for which we are checking) should be set to 0.96 0.98 1.00 1.02 1.04 .
- Adjust the unit cell geometry (given by the usual `lattice_vector` and `atom_frac` lines) to reflect primitive fcc or bcc. **There should only be one atom per unit cell in these cases!**
- Make sure that the subtraction that gives the value `energy` subtracts the free-atom reference only once, not twice (like in the diamond case).
- Set the `fit_range` to 10 20 100 (the volume of the unit cell has approximately been cut in half, as we now have only one atom, no longer two, per unit cell).

With these modifications made, you should be able to run the “Murnaghan fit” for fcc Si and for bcc Si and find equilibrium volumes, lattice parameters, and cohesive energies as before.

Again, the way to do this is to run the script `run.murn.sh` once for each phase (fcc or bcc):

```
run.murn.sh
```

, then to inspect the output file `murn.out` for the cohesive properties that characterize the equation of state, and to look at the `murn.out` file using `xmgrace`. If the lattice parameter you started with is too far away from the optimum, repeat the process with a set of new lattice parameters (volumes) closer to the optimum.

5.3 (iii) k -grid convergence fcc and bcc Si

Only for the lowest-energy lattice parameter, verify explicitly that the $12 \times 12 \times k$ -space grid calculation was sufficiently converged.

To that end, repeat the calculation for the lowest-energy lattice parameter using a $24 \times 24 \times 24$ k -space grid. By how much does the total energy deviate?

In these cases, be sure to use the total energy value that is extrapolated to $T \rightarrow 0$, not the straight total energy. The background is that the occupation numbers of the orbitals around the Fermi level can be associated with an entropy term, very similar to the analytic electron gas entropy from one of the earlier exercises. This property can be used to derive an extrapolation to zero smearing width even if the actual width used in the calculation is finite.

By inspecting the location and occupation numbers of the “highest occupied” and “lowest unoccupied” levels of both systems (fcc and bcc) in the final s.c.f. iteration, is either material a semiconductor or a metal?

5.4 (iv) Now which is the stable phase, and at what pressure?

This exercise asks the actual physical question: What is the order in which diamond, fcc and bcc Si become stable as a function of decreasing volume (increasing pressure)?

To that end, plot the three final $E(V)$ curves for each material into the same graph, using the cohesive energy and volume *per atom* (i.e., you will have to divide the numbers for the diamond structure by two).

Note: Just because a single structure is lowest in energy for a given volume range does not yet mean that that structure is also the stable one.

If there are two structures 1 and 3 surrounding it, one with lower and one with higher equilibrium volume, then at a given volume V a *mix* of structures 1 and 3 can still be more stable than structure 2 if:

$$\alpha E_1(V) + (1 - \alpha) E_3(V) < E_2(V) \quad . \quad (4)$$

In other words, if we can construct a *tangent* between the $E(V)$ curves for structures 1 and 3 such that the $E(V)$ curve of structure 2 is never below that tangent, structure 2 indeed never becomes stable.

Print out the plot with all three $E(V)$ curves and construct the common tangents between diamond, bcc, and fcc, respectively (just use a ruler). According to Eq. (2), the slope $\partial E/\partial V$ of such a tangent corresponds to a pressure. Indeed, as a function of increasing pressure, such tangents mark the *transition pressure* between two successively stable phases. Can you estimate the transition pressures (again, using a ruler, no sophisticated technology intended)?

This is not an abstract concept. For instance, carbon will only form diamond above such a transition pressure, else it will form the (rather less valuable) graphite. This transition pressure can be calculated *and* measured, and marks a real physical challenge for those trying to manufacture industrial diamonds.

6 Exercise 20: $E(V)$ of GaAs and ZnSe

After the preceding exercise, the present exercise merely marks an “application” of the same concepts to two more materials, in order to gain some broader insight.

For each material, create a separate subdirectory. Then, copy the `run.murn.sh` script from the Si diamond phase to either directory, and modify it to give the geometries of either GaAs or ZnSe for different lattice parameters around the experimental one.

Do not forget that here we are ONLY interested in the zincblende structure, not any others – no fcc equivalent, no bcc equivalent, etc.

The necessary modifications include:

- There are now two different `atom_types` instead of twice the same
- The reference lattice parameter should be changed
- We need the correct free-atom reference energies for both atoms in the structure (see below for pre-computed reference energies to use).
- Change the range of lattice parameters scanned to 2 % steps in the script:

```
for scale in 0.96 0.98 1.00 1.02 1.04
```
- Make sure that the volume range for which the fitted $E(V)$ curve is output for plotting purposes is approximately right:

```
echo "fit_range 30 50 100" >> murn_fit.in
```

In `control.in`, use “light” settings and $12 \times 12 \times 12$ k -space grids.

Note that these are heavy elements. This means that the core electrons feel *relativistic* effects which would influence the cohesive properties. Thus, FHI-aims will force you to choose an appropriate treatment of these relativistic effects (so far, we have been silently using a non-relativistic Schrödinger-like kinetic energy operator only).

In `control.in`, set

```
relativistic atomic_zora scalar
```

which, in brief, invokes a so-called “scalar relativistic” approximation to deal with relativistic effects (ZORA stands for “zero-order regular approximation”). Without going into detail, this particular level of approximation is essentially correct for valence electron properties (without spin-orbit coupling). For core levels, the absolute placement is not yet correct compared to a fully relativistic treatment, but this error is systematic and cancels out in energy differences (like the cohesive energy). This is still better than a non-relativistic treatment, where the valence electron treatment would not be correct as well.

6.1 Free atoms

For each element (Ga, As, Zn, Se), we have computed the reference total energy with light settings and all radial functions up to and including tier 2.

We could compute these ourselves, trying to identify the lowest-energy ground states in LDA (symmetry-broken or not). This search for the proper ground state would require some time and “fiddling” (but it can be done). **To speed up the process, we here simply list the appropriate reference energies (just copy and paste into `run.murn.sh!`):**

Element	Free-atom reference energy [eV]
Ga	-53101.53863693
As	-61806.15461687
Zn	-49042.43684254
Se	-66455.94868451

6.2 $E(V)$

Use the appropriate free-atom reference values in the `run.murn.sh` scripts for either compound. Compute $E(V)$ explicitly for five volumes around the experimental one, and deduce the cohesive properties as before. How do they compare to the experimental values given on the exercise sheet? Tabulate and compare your values. Are they consistent with the known properties of the LDA?

6.3 A final remark

All parts of these exercises were conducted with “light” settings. In fact, in these particular cases we would still have noticed some change when going from “ligh” to “tight” (especially when changing the basis sets). The fully converged basis sets are not much bigger than what was used above, but in all cases, the lattice parameters would have decreased a little bit further. Thus, the use of light settings makes the LDA look even a little better than it should. In a real world application, we would normally attempt to converge the basis set towards at least “tight”.

References

- [1] F. Murnaghan, “The compressibility of media under extreme pressures,” Proceedings of the National Academy of Science **30**, 244 (1944).