

---

# Theoretical Material Science: Electronic Structure Theory at the Computer

---

Prepared by Christian Carbogno,  
based on an exercise by Lydia Nemeč and Volker Blum  
Berlin, May 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

<b>1</b>	<b>Important Notes</b>	<b>3</b>
<b>2</b>	<b>Exercise 13: Background</b>	<b>3</b>
2.1	Relevance for Density-Functional Theory . . . . .	3
2.2	Brillouin Zone Integrals . . . . .	4
2.3	The Upshot . . . . .	5
<b>3</b>	<b>Reminder: FHI-aims in the Virtual Machine</b>	<b>5</b>
<b>4</b>	<b>Exercise 13a: The Diamond Phase of Si – Total Energy</b>	<b>5</b>
<b>5</b>	<b>Exercise 13b: The Diamond Phase of Si – Electron Density</b>	<b>7</b>
<b>6</b>	<b>Exercise 13c: The Diamond Phase of Si – “Band Gap” in LDA</b>	<b>8</b>
<b>7</b>	<b>Exercise 14: Background</b>	<b>9</b>
7.1	Equations of State . . . . .	10
<b>8</b>	<b>Exercise 14a: <math>E(V)</math> of the diamond phase of Si</b>	<b>11</b>
8.1	Total Energies . . . . .	11
8.2	Cohesive Energies . . . . .	12
8.3	Parameterizing Murnaghan’s Equation of State . . . . .	13
8.4	Automating the Process: A Shell Script . . . . .	14
<b>9</b>	<b>Exercise 14b: Other Phases of Si</b>	<b>16</b>
9.1	(i) Nearest-Neighbour Distance . . . . .	16
9.2	(ii) Equilibrium Cohesive Properties of fcc and bcc Si . . . . .	16
9.3	(iii) $k$ -grid Convergence for fcc and bcc Si . . . . .	17
9.4	(iv) Now which is the stable phase, and at what pressure? . . . . .	17
<b>10</b>	<b>Exercise 15: <math>E(V)</math> of GaAs and ZnSe</b>	<b>18</b>
10.1	Free Atoms . . . . .	19
10.2	$E(V)$ . . . . .	19
10.3	A Final Remark . . . . .	19

# 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 Exercise 13: Background

In a periodic potential, Bloch’s theorem states that the effective single-particle eigenfunctions have the following form:

$$\varphi_{n,\mathbf{k}}(\mathbf{r}) = e^{i\mathbf{k}\mathbf{r}} u_{n,\mathbf{k}}(\mathbf{r}). \quad (1)$$

Here,  $u_{n,\mathbf{k}}(\mathbf{r})$  is a function that has the exact same periodicity as the potential itself. Only the phase of the eigenfunction will be different in different unit cells.  $\mathbf{k}$  denotes the “crystal momentum”, a continuous quantum number whose unique values are restricted to a single unit cell of the reciprocal space. Conventionally, one discusses the first Brillouin zone (the Wigner-Seitz cell of the reciprocal lattice), but in fact any other unit cell of reciprocal space would do as well.

For a *given* periodic potential,  $\mathbf{k}$  is a good quantum number, and the effective single-particle equations can be solved independently for any given  $\mathbf{k}$ , without the solutions from any other  $\mathbf{k} \neq \mathbf{k}'$  interfering. Eigensolutions at different  $\mathbf{k}$ -points are decoupled. We only have to solve one  $\mathbf{k}$ -point at a time, not all of them at once.

For each  $\mathbf{k}$ -point, the single-particle Schrödinger equation still has multiple different solutions, just like in a non-periodic system. Different states  $n$  label these different solutions  $\{\varphi_{n,\mathbf{k}}(\mathbf{r})\}$  with different eigenvalues  $\epsilon_{n,\mathbf{k}}(\mathbf{r})$ .

### 2.1 Relevance for Density-Functional Theory

Happily, our “most productive” workhorse theory, Kohn-Sham density-functional theory, relies on solving effective single-particle equations. We can do this in a periodic potential. The Kohn-Sham eigenfunctions and eigenvalues then take exactly the form above:  $\{\varphi_{n,\mathbf{k}}(\mathbf{r}), \epsilon_{n,\mathbf{k}}(\mathbf{r})\}$ . For a *given* potential  $v_{\text{eff}}(\mathbf{r})$ , these solutions can be obtained independently at each  $\mathbf{k}$ -point. Formally, they are decoupled.

Only when we use the solutions  $\varphi_{n,\mathbf{k}}(\mathbf{r})$  from all  $\mathbf{k}$ -points to create a new density  $n(\mathbf{r})$  and, subsequently, a new potential  $v_{\text{eff}}(\mathbf{r})$  in a self-consistency cycle we do couple results from different  $\mathbf{k}$ -points. But this is only an implicit coupling to get to the self-consistent density  $n_0(\mathbf{r})$ . Once we have  $n_0(\mathbf{r})$ , the actual Kohn-Sham equations are still decoupled.

This is an excellent result. A macroscopic solid has  $O(10^{23})$  particles, and if one were to solve the full many-body Schrödinger equation with all electrons explicitly, their motion would *not* be spatially periodic. Sure, the time average of their motion (their density and their correlation functions) would be periodic, but one could not decouple the electrons in different unit cells from one another in a simple way.

Density-functional theory is different: We have hidden the correlated motion of all the “real” electrons inside a simple potential  $v_{\text{xc}}(\mathbf{r})$ . As a result, the physical ground state density is periodic, the effective potential is periodic, and we can use Bloch’s theorem to describe the effective single-particle states of the Kohn-Sham equations. Instead of having to describe all unit cells together and explicitly, we only need to describe the contents of a single unit cell explicitly. For the observable quantities—in Kohn-Sham theory, the ground state total energy  $E_0$  and the ground-state density  $n_0(\mathbf{r})$ —we are still exact, as long as we make no approximations to  $v_{\text{xc}}(\mathbf{r})$ .

Of course, in practice we must approximate  $v_{\text{xc}}(\mathbf{r})$ , but the approximations that we have can be very good indeed. Bloch’s theorem and density-functional theory give us an excellent starting point for a theory of real materials, as long as they are (somewhat) periodic.

## 2.2 Brillouin Zone Integrals

We wish to describe the properties of an infinite, periodic solid by solving the Kohn-Sham equations. If we look at the transition from non-periodic to periodic systems, we realize that some very important quantities now become integrals of the Brillouin zone. For example, the density:

$$n_0(\mathbf{r}) = \begin{cases} \sum_i f_i |\varphi_i(\mathbf{r})|^2 & \text{(non - periodic)} \\ \sum_n \int_{\text{BZ}} d^3k [f_{n,\mathbf{k}} \cdot |\varphi_{n,\mathbf{k}}(\mathbf{r})|^2] & \text{(periodic)} \end{cases} \quad (2)$$

or the sum of occupied eigenvalues:

$$\sum_n \int_{\text{BZ}} d^3k [f_{n,\mathbf{k}} \cdot \epsilon_{n,\mathbf{k}}] \quad . \quad (3)$$

$i$  denotes the states of a non-periodic system, and the  $f_i$  and  $f_{n,\mathbf{k}}$  denote the occupation numbers (for instance, from a Fermi function at finite temperature) in non-periodic and periodic systems, respectively.

We have thus circumvented the “big” problem of the full correlated solid, but a small problem remains. To obtain the density and total energy, we still have to solve continuous integral at (theoretically) infinitely many  $\mathbf{k}$ -points (all located in a single unit cell of reciprocal space). In practice, we thus need another approximation: We need to solve the resulting integrals, for instance by discretizing  $\mathbf{k}$ -space to a few  $\mathbf{k}$ -points only and rewriting the integral approximately as a weighted sum (an integral over a step function, as in elementary mathematics):

$$\sum_n \int_{\text{BZ}} d^3k F(\mathbf{k}) \rightarrow \sum_{\mathbf{k}} [w(\mathbf{k}) \cdot F(\mathbf{k})] \quad . \quad (4)$$

To evaluate this expression, we have to find “a few”  $\mathbf{k}$ -points that are spread across the Brillouin zone (or unit cell of reciprocal space), find their integration weights, and then evaluate the Kohn-Sham equations only at these chosen  $\mathbf{k}$ -points.

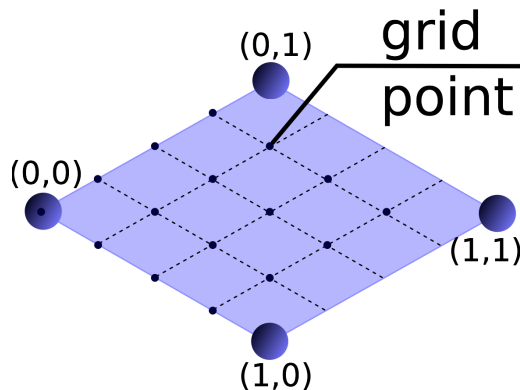


Figure 1: Reciprocal-space unit cell of a two-dimensional hexagonal lattice (easier to illustrate than the three-dimensional case), reciprocal lattice vectors (large blue dots) and even-spaced  $\mathbf{k}$ -point integration grid that covers the unit cell. We could also have drawn the first Brillouin zone (the Wigner-Seitz cell of the reciprocal lattice). The key point is that the  $k$ -space integration must cover a full unit cell area of reciprocal space.

The historically simplest choice to do so is an even-spaced integration grid of  $\mathbf{k}$ -points along the directions of the reciprocal lattice vectors. Figure (1) shows a two-dimensional example for the reciprocal-space unit cell of a hexagonal lattice. Here, we have chosen the following  $\mathbf{k}$ -space integration grid (small black dots):

$$\mathbf{k}(s_1, s_2, s_3) = (s_1/n_1) \cdot G_1 + (s_2/n_2) \cdot G_2 + (s_3/n_3) \cdot G_3. \quad (5)$$

$G_i$  denote the reciprocal lattice vectors (large circles), and  $0 \leq s_i \leq n_i$  are integers. To fully specify a grid of  $\mathbf{k}$ -points in this way, all we need to do is choose integers  $n_1, n_2, n_3$ . The result is indeed

an even-spaced  $\mathbf{k}$ -grid with equal integration weights  $w(\mathbf{k})$  and a grand total of  $n_1 \cdot n_2 \cdot n_3$  points. The number grows fast with the mesh refinement, but we have a definition. We will refer to such a mesh as a “ $n_1 \times n_2 \times n_3$  k-point grid” below.

As a side remark, there are many works that attempt to go beyond this simple choice of  $\mathbf{k}$ -points, to exploit symmetry, or to go beyond this simple integration method. We can not go through all these developments here. By and large, however, the method chosen above works despite all its potential flaws, and is, in a sense, still the *de facto* standard of the field. Since the “ $\Gamma$  point” ( $\mathbf{k}=(0,0,0)$ ) is explicitly included, the type of grid above is also referred to as a  $\Gamma$ -centered grid.

## 2.3 The Upshot

We now have a simple recipe to investigate a bulk solid in density-functional theory. We choose

- the unit cell,
- the positions of the atoms inside the unit cell,
- and mesh factors  $n_1, n_2, n_3$  for the  $\mathbf{k}$ -point grid.

The key point is that, of course, we wish to approximate the results of the continuous Brillouin zone integration, so we need to make sure that our mesh  $n_1, n_2, n_3$  is chosen to be dense enough.

## 3 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`.

## 4 Exercise 13a: The Diamond Phase of Si – Total Energy

Silicon is (arguably) the most popular solid in the world, at least when it comes to solid-state theory. We here focus only on its experimental phase, the diamond phase.

To set up a calculation for Si, we could simply use the conventional (cubic) unit cell of the fcc lattice and place two Si atoms at each of its sites. This choice, however, would give us a total number of eight Si atoms per unit cell, as the conventional fcc cell itself has four sites.

In practice, we can do better and use the *primitive cell* of the fcc lattice instead. It is easy to convince yourself that the following choice of geometry will do:

```

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

```

Looking over the numbers, you should see that these values correspond to a (conventional) lattice parameter of  $a=5.43$  Å. You can visualize (and check) your input by:

```
jmol geometry.in &
```

Then, to repeat unit cell geometries and visualise periodic structures, press the right mouse button and choose from the menu the category

Symmetry

and then

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

Can you see the tetrahedron typical for the diamond structure?

We next need to think about our computational choices in `control.in`. For instance:

```

# 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 (to be adjusted)
k_grid  3 3 3

```

You have seen most of these settings in the earlier exercises.

**Note our choice of the LDA exchange correlation functional for the present calculation. For the present exercise, it is critical that you choose LDA and not, by accident, Hartree-Fock theory. A calculation for bulk Si in Hartree-Fock theory can be done, but at the price of significantly more computer time and memory—as well as the need to deal with the Coulomb operator between electrons in different unit cells, which leads to significantly worse convergence with the number of k-points. In short, do not use Hartree-Fock theory.**

The only interesting *new* choice is the `k_grid` tag. In the present example, it specifies a  $3\times 3\times 3$  grid of k-points, which will be used for all Brillouin zone integrations.

Before we continue, we must still pick settings for the element Si. In the interest of time, we here do NOT use the “tight” settings that we used for the free atom exercises earlier. Instead, we use all default settings—including the basis set—from the “light” settings:

```
> cat /home/theory/fhi-aims.160328.3/species_defaults/light/14_Si_default >> control.in
```

You are absolutely welcome to try out “tight”er settings for the solid, but this is not a requirement for the present exercise.

It is a good idea to generate a directory

```
mkdir Si_diamond
```

and enter the directory with:

```
cd Si_diamond
```

Copy your `geometry.in` file for silicon in diamond structure and your `control.in` file. Now you have all the pieces to run your first periodic calculation.

```
> aims | tee Si_diamond_kgrid_3x3x3.out
```

Then, repeat the calculations for different  $k$ -grids, for example:

- `k_grid 3 3 3`
- `k_grid 4 4 4`
- `k_grid 6 6 6`
- `k_grid 12 12 12`
- `k_grid 24 24 24`

For each calculation, record the final total energy value—the one given in the last self-consistency cycle. How does the total energy converge with the  $k$ -grid? Which level do you consider sufficient for bulk Si?

**In the exercise, we are actually interested in the cohesive energy. This energy is defined as:**

$$E_{\text{coh}} = E_{\text{tot}} - N_{\text{Si}} \cdot E_{\text{free atom}} \quad , \quad (6)$$

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

**For the free atom reference, please use exactly the same total energy that you found in exercise number 10f, for LDA, tight settings, and a tier 2 basis set, which yielded a spherical free atom ( $\approx -7842.789$  eV). Be sure to NOT use the total energy from a Hartree-Fock calculation by accident.**

## 5 Exercise 13b: The Diamond Phase of Si – Electron Density

The *other* physical quantity to be obtained from Kohn-Sham density-functional theory is the density  $n_0(\mathbf{r})$  itself. We will here only visualize the density. In the “real world”, the density would give you immediate access to structure factors in  $X$ -ray diffraction, charge transfer (in less simple solids, that is), and many other quantities.

To visualize the electron density, go back to one of the previous calculations that you consider “converged”, and add the following lines to `control.in`:

```
output cube total.density
cube filename total.density.cube
```

Visualize the result (using jmol). Remember (from exercise 10):

In jmol, you will have to open the “Console” from the “File” menu. Once you have this console, you can use typed commands to specify which isosurface of the density you would like to see.

Once you have opened “jmol” and the “Console”, type:

```
load "total_density.cube"
```

Now we still have to tell jmol which “isosurface” of the electron density we would like to see. For example:

```
isosurface cutoff 0.1 "total_density.cube"
```

Interestingly, you may find that 0.1 gives you a slightly puzzling result. To visualize the density in a better way, increase the cutoff value to perhaps 0.5 ... what do you find? Did you expect this result?

It turns out that visualizing a total density on its own is often not the most informative choice when trying to understand a chemical bond. The density itself is large, and density *changes* are often small compared to the absolute density. You can easily verify this statement by looking at the total density for an underconverged  $\mathbf{k}$ -point grid, which still looks very similar to the converged one—although the total energy is clearly different.

We can learn some more by looking at a density difference—in this case the difference between the full crystal and the free atom:

$$\delta n(\mathbf{r}) := n(\mathbf{r}) - \sum_I n_{\text{free atom}}(|\mathbf{r} - \mathbf{R}_I|) \quad , \quad (7)$$

where  $\{\mathbf{R}_I\}$  are the positions of all Si atoms in the entire crystal.<sup>1</sup> The free atoms are here taken to be spherically symmetric, as in exercise 8. Fortunately, FHI-aims supports this output of a difference density by default, by setting:

```
output cube delta_density
cube filename delta_density.cube
```

in `control.in`. Visualize the result in jmol and play with the cutoff parameter for isosurfaces. (Hint: choose a smaller value than what was needed for the full density.) What is the result? Where is the difference density concentrated? Does this match your expectations for a diamond Si crystal?

## 6 Exercise 13c: The Diamond Phase of Si – “Band Gap” in LDA

The “band gap” of an insulating or semiconducting solid is defined as:

$$E_{\text{gap}} = I - A = E(N - 1) + E(N + 1) - 2E(N) . \quad (8)$$

Here,  $I$  is the ionization potential (energy needed to remove an electron from the solid),  $A$  is the electron affinity (energy needed to add an electron to the solid), and  $E(N - 1)$ ,  $E(N)$ , and  $E(N + 1)$  are the energies of the singly positive, neutral, and singly negative *entire solid*, where  $N$  is the number of electrons in the neutral case.

<sup>1</sup>Yes, this sum converges. For instance, the ground state densities of exact free atoms fall off exponentially with the distance from the nucleus. In LDA, the decay behaviour is not exponential—one of the “failures” of LDA for free atoms—but the decay is still fast enough to converge.



The band gap, as written above, is an inconvenient quantity to compute exactly (see lecture script), as (i) the entire solid is involved, and (ii) there is a technical problem with most simple density functionals that would simply “delocalize” that single electron until, by virtue of Janak’s theorem, only the difference between the highest occupied and lowest unoccupied Kohn-Sham eigenvalue of the neutral crystal remains:

$$E_{\text{gap}}^{\text{naive}} = \epsilon_{\text{CBM}} - \epsilon_{\text{VBM}}. \quad (9)$$

We have here used the conventional abbreviations “VBM” for “valence band maximum” and “CBM” for “conduction band minimum” in solids, which in practice mean the highest unoccupied level in the Brillouin zone, and the lowest unoccupied level in the Brillouin zone.

We will come back to the fine details of “band structures” later, but for the specific integration grid used in a particular calculation, FHI-aims gives you its “best estimate” (only an estimate, as we are not sampling the full, continuous Brillouin zone!) of  $E_{\text{gap}}^{\text{naive}}$ ,  $\epsilon_{\text{CBM}}$ , and  $\epsilon_{\text{VBM}}$ . In the output, look for the following section:

```
Highest occupied state (VBM) at      -5.77245422 eV (relative to internal zero)
| Occupation number:      2.00000000
| K-point:      1 at      0.000000      0.000000      0.000000 (in units of recip. lattice)

Lowest unoccupied state (CBM) at      -5.13991586 eV (relative to internal zero)
| Occupation number:      0.00000000
| K-point:      7 at      0.333333      0.000000      0.333333 (in units of recip. lattice)

Overall HOMO-LUMO gap:      0.63253836 eV between HOMO at k-point 1 and LUMO at k-point 7
| This appears to be an indirect band gap.
| Smallest direct gap :      2.52459934 eV for k_point 1 at      0.333333      0.000000      0.333333
```

How do these values evolve with the  $\mathbf{k}$ -grid used? What do you learn about the relative locations of VBM and CBM?

Note that only the *difference* between eigenvalues has a clear physical meaning in solids. The absolute location is shifted by an arbitrary amount: The  $G = 0$  component of Ewald’s potential. Without knowing the boundaries of an experimental crystal (surface orientation etc.), it is not possible to relate this value to an absolute “vacuum level.”

Note also that we have here taken the Kohn-Sham eigenvalues themselves as indicative of a measurable quantity: The band gap. We learned earlier that Kohn-Sham eigenvalues, strictly, have no guaranteed physical meaning, and this is in fact the case. When comparing Kohn-Sham band gaps to experimentally measured values (e.g., from photoemission spectroscopy), one can go horribly wrong.

On the other hand, Kohn-Sham eigenvalues are the simplest quantity to be compared, and they do arise from a physically motivated potential. Thus, they are still often used for a qualitative look at the “electronic structure” of real solids. Take a look at the “naive band gap” values you extracted for bulk Si. How close are they to experimental values?

[http://en.wikipedia.org/wiki/Band\\_gap](http://en.wikipedia.org/wiki/Band_gap)

The question whether and how much the eigenvalues of a hypothetical “exact” Kohn-Sham theory would or would not reproduce an experimental band gap is still a hotly contested issue today. ( $E(N - 1)$ ,  $E(N)$ , and  $E(N + 1)$  are ground state properties of some system, after all.) We will not touch this issue in detail here.

## 7 Exercise 14: 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.

## 7.1 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:

$$E(V) = E_0 + \frac{B_0 V}{B'_0} \left[ \frac{(V_0/V)^{B'_0}}{B'_0 - 1} + 1 \right] - \frac{B_0 V_0}{B'_0 - 1} \quad (10)$$

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 (10) 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. (10) is even more useful in practice. Consider the following textbook definition of pressure:

$$p = - \left( \frac{\partial E}{\partial V} \right)_S \quad (11)$$

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. (10) 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.

## 8 Exercise 14a: $E(V)$ 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  $a_{\text{exp}}=5.43$  Å. We recommend lattice parameter changes of  $\pm 1$  % and  $\pm 2$  % for the actual data points chosen. For this purpose, a  $k$ -space integration grid with a density of  $12 \times 12 \times 12$  will be used (see previous exercise).

### 8.1 Total Energies

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

```
mkdir Si_diamond_EoS
```

and enter the directory with:

```
cd Si_diamond_EoS
```

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 /home/theory/fhi-aims.160328.3/species_defaults/light/14.Si_defaults >> control.in
```

We can now run the calculation, for example:

```
> aims | 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.

## 8.2 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):

$$E_{\text{coh}} = E_{\text{tot}} - N_{\text{Si}} \cdot E_{\text{free atom}} \quad . \quad (12)$$

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

Again, use the value for the free atom computed in previous exercises (tight, tier 2, LDA, symmetric solution) of approximately  $-7842.789$  eV. This 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.

### 8.3 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 Fortran program. To compile it, just change into the directory that contains the source code (feel free to look at it):

```
cd /home/theory/fhi-aims.160328_3/utilities
```

and type

```
gfortran murn.FH.v4.f -o ../bin/murn.v4.x
```

The binary will then be available as `/home/theory/fhi-aims.160328_3/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  $30 \text{ \AA}^3$  and  $50 \text{ \AA}^3$
- The remaining lines are your computed cohesive energies at five different volumes. Replace the `<energy_i>` placeholders by the volume in  $\text{\AA}^3$ , and the `<volume_i>` placeholders by the cohesive energy in eV per *two* Si atoms.

If you run: `/home/theory/fhi-aims.160328_3/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. (10), 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?

## 8.4 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.

*Do not attempt to type this script by hand in its entirety!*

You can download it either via web browser from

<https://thcloud.rz-berlin.mpg.de/s/H0F8gcN4KUI6vBk>

or by typing

```
curl https://thcloud.rz-berlin.mpg.de/s/H0F8gcN4KUI6vBk/download -o run.murn.sh
```

into the shell.

```

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

aims_x=/home/theory/fhi-aims.160328_3/bin/aims.160328_3.serial.x
MURN=/home/theory/fhi-aims.160328_3/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=$(echo "-7842.78893188" | bc -l)
reference_energy_2=$(echo "-7842.78893188" | 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

    $aims_x < /dev/null > Si_diamond.a_$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.

## 9 Exercise 14b: 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.

### 9.1 (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.

### 9.2 (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_type 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.

### 9.3 (iii) *k*-grid Convergence for 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?

### 9.4 (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:

$$\alpha E_1(V) + (1 - \alpha) E_3(V) < E_2(V) \quad . \quad (13)$$

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). Alternatively, used the line drawing feature of *xmgrace* (*Window*→*Drawing objects*→*Line/Move/Edit*). According to Eq. (11), 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.

## 10 Exercise 15: $E(V)$ 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.

## 10.1 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

## 10.2 $E(V)$

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?

## 10.3 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).