NTNU
Norwegian University of
Science and Technology

Density Functional Theory Studies of Precipitate Interfaces in Aluminium Alloys, with Focus on Theta'-Al2Cu

Philip Østli

MSc in Physics
Submission date: June 2016
Supervisor: Randi Holmestad, IFY
Co-supervisor: Jesper Friis, SINTEF
Inga Ringdalen, SINTEF
Sigurd Wenner, IFY

Norwegian University of Science and Technology
Department of Physics

NTNU – Trondheim
Norwegian University of
Science and Technology

Density functional theory studies of precipitate interfaces in aluminium alloys, with focus on θ'-Al_2Cu

Philip Østli

June 2016

MASTER THESIS

TEM Gemini Centre, 2015-2016 Department of Physics

Norwegian University of Science and Technology

Main supervisor: Professor Randi Holmestad

Co-supervisors: Dr. Jesper Friis, Dr. Inga Ringdalen, Dr. Sigurd Wenner

Summary

The atomic structure of bulk crystal of aluminium and the precipitate phase θ' connected to age-hardening of Al-Cu alloys have been studied using Density Functional Theory (DFT). The lattice constant of θ' at 0 K having a coherent interface with the aluminium is found, for thicknesses up to six unit cells. Five different interface-configurations have been studied in four alloys: Al-Cu, Al-Cu-Ag, Al-Cu-Li and Al-Cu-Mg-Si. The interface stucture of these is found. The lattice misfit of the precipitate with the aluminium matrix around is found for different thicknesses in each of the four alloys, and compared to the strain length and interface energy. The interfacial energies are computed for each thickness taking into acount the strain energy.

Sammendrag

Krystallstrukturen til bulk aluminium og presipitatet θ' knyttet til utherdbare Al-Cu legeringer har blitt studert med Density Functional Theory (DFT). θ' har ved 0 K en koherent grenseflate med aluminium, og gitterkonstanten til denne er funnet for tykkelser av presipitatet opp til seks enhetsceller. Fire forskjellige grenseflate konfigurasjoner i fire legeringer er blitt studert: Al-Cu, Al-Cu-Ag, Al-Cu-Li og Al-Cu-Mg-Si. Grenseflatestrukturen til disse er bestemt. Gittermistilpasningen til presipitatet i forhold til aluminiumen rundt ble funnet for forskjellige tykkelser og med hver av de ulike grenseflatene i de fire legeringene. Gittermistilpasningen er korrelert til strekklengden i matrisen utenfor grenseflaten og grenseflateneenergien. Energien i grenseflatene ble beregnet for hver tykkelse i hver struktur og inkluderer strekkenergien i gitteret som varierer med gittermistilpasningen.

Preface

This Master's thesis is part of the Master of Science in Physics (MScPHYS) programme at NTNU, and was carried out during the second year of the degree in autumn/spring of 2015/2016. The work is part of the FRINATEK-project ("Fundamental investigations of precipitation in the solid state with focus on Al-based alloys") which is a combination of experimental and computational studies of precipitate phases from a wide range of Al alloy compositions. The thesis is divided into five chapters: The first chapter contains a introduction and a summary of what is done pre date on this topic and what is done in this work. The second part is an introduction to precipitates in alloys and age-hardening in general. In the third part we develop applied density functional theory from the Schrödinger equation and introduce the different approximations and techniques used in modern quantum chemistry computer codes. In chapter four the method used to compute the interfacial energies is explained and all the technical information needed to reproduce the data is given. Chapter five contains the results from each step, and involves some comments along the way. At the end of chapter five there is a general discussion about the findings, and a concluding summary. You can find the list of tables and figures helpful, because there are a lot of them and it gives a good overview. You are not supposed to have long experience in material physics to follow the text, but a common knowledge of quantum physics is advised.

Acknowledgement

I would like to thank the following persons for their great help during the time of writing this thesis. Jesper Friis from SINTEF Materials and Chemistry helped me alot with the calculations and with the usage of the DFT sofware. I would not have gotten far without him. I found great help in Inga Ringdalen from SINTEF when I needed quick response. Sigurd Wenner gave me idéas for the assignment, and provided me with experimental data of the different structures. My main supervisor Randi Holmestad were a great help during the final stages of the writing of the thesis. She pushed me forward during the year and kept herself updated on my progress, and I am very greatful for all the meetings.

Trondheim, June 2016 / Philip Østli

Contents

List of Figures

List of Tables

Chapter 1

Introduction to the work

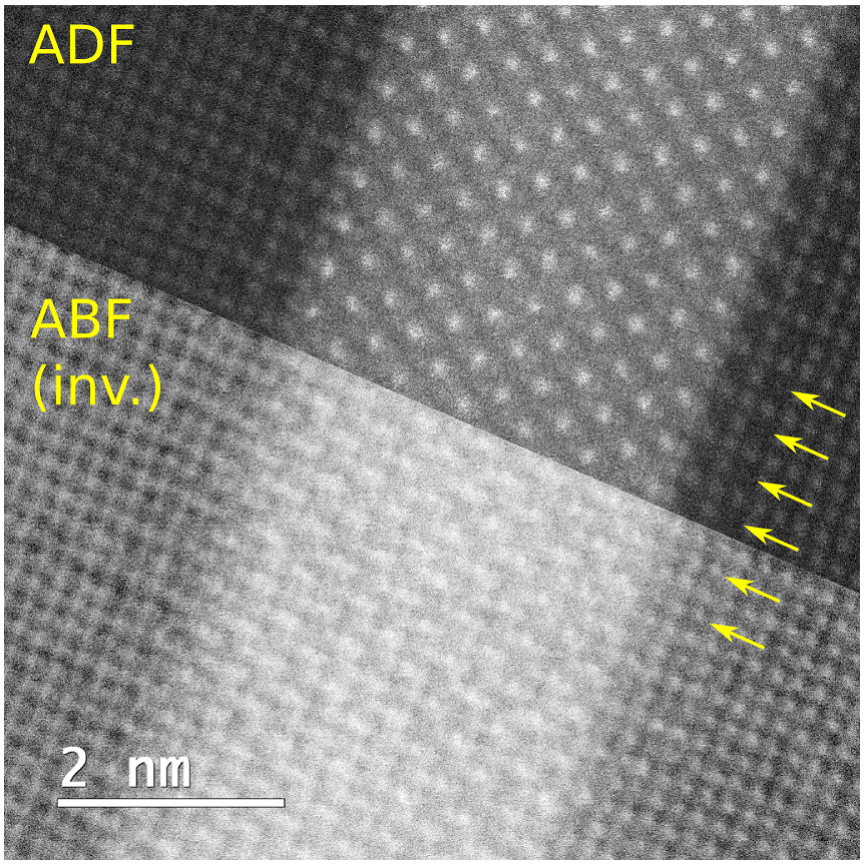


Figure 1.1: STEM images (Annual Dark Field and inverted Annual Bright Field) of a θ' phase with a missing Cu interface layer. Instead, two layers of alternating Al and Li columns are observed. The arrows point at the Li atoms. Imaged in the $[010]_{Al}$ zone axis.

1.1 Motivation

In age-hardenable aluminium alloys we see very thin precipitates forming during aging at medium high temperatures (around 150 °C). These precipitates play an important role when explaining the extreme strength enhancement of the alloy after heat treatments. An important precipitate in copper-alloyed aluminium, is θ'' that has a coherent interface with the aluminium matrix around. Figure 1.2 shows the age hardening curves for Al-Cu alloys with different compositions. These results indicate the same thing; that the main strengthening of the alloy come from the interaction of dislocations with the coherent interface of the θ'' precipitate. What is happening when the θ' is created after θ'', is that the strength suddenly drops and the alloy is said to be overaged. This happens at the same time as the precipitates grow thicker, and the number of precipitates goes down. Dislocations now have more room to propagate in, and the alloy gets softer. What if one could slow down the growth by adding a third element to the alloy, and thus improve the strength even further? This is actually possible by adding small amounts of a third solute element like Ag [1] or Mg [2]. What one can see in the microscope is that this solute tend to migrate to the surface of the precipitates and so enhance their fitting with the aluminium matrix, and also lowering the interface energy between the precipitate and the matrix.

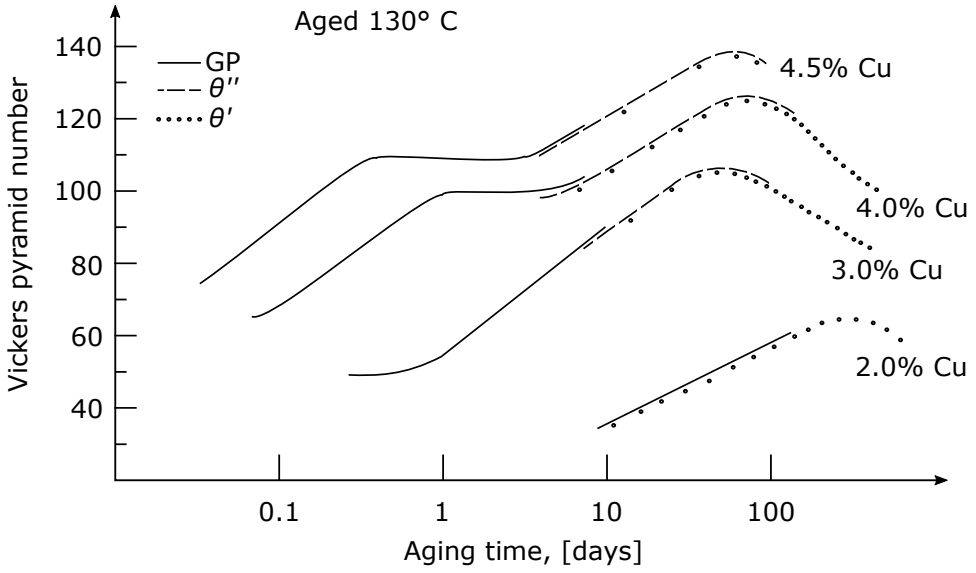


Figure 1.2: Age hardening curves for Al-Cu alloys, showing Vickers pyramid number as the aging in days, held at 130 °C. (adapted from [3])

In this thesis we will look at four (five) different interface-configurations which can

enhance the tensile strength. These are all Al-Cu alloys, some with small additions of extra elements. We will take a look at a normal **Al-Cu** alloy, and comparing two different interfaces, one with a copper layer at the interface, and one without. Then we will look at special interfaces with respectively **1. Silver (Ag), 2. Magnesium and Silisium (Mg, Si), and 3. Lithium (Li)** on them. We will do calculations on equillibrium structure of the different precipitate pluss interfaces using Density Functional Theory (DFT) with the Vienna Ab initio Simulation Package (VASP) computer code. The simulation models for calculations, we create using Atomic Simulation Evironment (ASE). With this software we can do relaxations and calculate total free energy, lattice misfits, precipitate- and interface thicknesses, interface energies and other interesting parameters. We will find that the interface energies are comparable to each other and there is also a correlation between the strain energy and the misfit of the precipitates. We will define these sizes in section 4.2.

The θ' precipitate is firmly studied. Bourgeois et al. describes the mechanism causing the Al-Cu precipitate to grow[4]. By using DFT and TEM analysis tools, they discovered a new interface with interstitial copper atoms at the surface, proving that the no-copper interface is only an intermediate state of precipitate thickening. A partial c-phase at the interface is first observed by Wenner et al. in Al-Cu-Mg-Si alloys[5]. The structure we use in our calculations, are according to the findings of Torsæter et al. [6]. The silver interface is described by Rosalie et al. [1] and we will prove that this is a very beneficial structure in the Al-Cu-Ag alloys. The lithium interface-structure have been observed in our TEM-lab at NTNU by Wenner, but is not yet been published with DFT results and thus we might have some interesting new things to show. In general, studies like these are important for our understanding of the physics of phase stability in Al alloys. Comparing the results from calculations with well established databases available for Al alloys, one gain confidence in the predictive power of first-principles methods for systems where experimental information is missing or incomplete. For instance, recent first-principles predicions in the Al-Cu system have provided results in the areas of ground state stability [7], metastable precipitate equilibri and precipitate morphologies [8] [9]. What can be studied further is how different phases and atom compositions can lower the interface energy even more, so that one can have many thin precipitates that contribute to a stong and light material. Is there a combination of

elements that have not yet been discovered to be the perfect mix for a super strong alloy? And is it possible to predict the strength and other parameters of the alloy? The industry sometimes need materials with a special set of properties that fits their specific problem. The ultimate goal of such studies would be to make something of a cooking-recipe for strong alloys with the desired properties that predicts the constituents needed to make them.

Chapter 2

Aluminium alloys and precipitation

Aluminium is one of the most used ground metal in the world, after steel (iron). It is highly accessible, easy to recycle and we have a lot of it. In this chapter we will have a brief introduction to alumnium alloys in general, why they are so popular and where they are utillized. Then we will talk about their internal structure and try to understand why they get stronger when subjected to different treatment methods. After that we will look at the aluminium-copper (Al-Cu) system and what phases we can observe in it. We also learn a little about nucleation and precipitation in solid solutions, and different types of interfaces and dislocations.

2.1 Alloy types

Aluminium is a veratile material, and it has a lot of useful properties. It is lightweight and easy to form, a great heat transporter and resistable to corrosion and external magnetic fields. We use it in electrical devices, cars, buildings and as packaging. In its pure form aluminium is soft and bendable and not very useful. But if you add small amounts of a secondary elements such as copper or silver you get a dramatic increase in strength and hardness of the alloy. It is this characteristic that we take advantage of i.e. when we want to build light but strong constructions, or if we want to cut down on energy consumption in production and transportation. The different alloys can be classified according to their secondary added element(s). There is a classification system called "The International Alloy Designation System" of aluminium where the first digit in their group number decides what element is added. In table 2.1 the different classes of alloys

Table 2.1: The International Alloy Designation System is the most widely accepted naming scheme for wrought alloys. Each alloy is given a four-digit number, where the first digit indicates the major alloying elements.

Alloy group	Main alloying elements	Applications
1xxx	-	electrical, chemical
2xxx	Cu, (Mg, Li)	aircraft, (aerospace)
3xxx	Mn	general purpose
4xxx	Si	architectural
5xxx	Mg	architectural, marine constructions
6xxx	Mg, Si	aircraft, boats, automotive parts, general
7xxx	Zn, (Mg, Cu)	aircraft, aerospace, bicycle frames
8xxx	Li, Sn, Zr, B, other	

and their main applications are listed. The Al-Cu alloy is a two-thousand (2xxx) alloy thus the biggest alloying element is Cu, with additional ternary elements added for fineadjustment of different properties or to enhance the treatment prosess. Secondary or ternary element additions give the possibility to tune the properties of the metal approximately without increasing the density of the metal. When adding lithium for instance, which is lighter than aluminium, one can get an increase in stiffness by 5 %, and a reduction in density of 3 % per 1 wt% of lithium added [10]. By adding copper, you can get much more stiffness than with pure aluminium or pure copper. The Al-Cu alloys are one of the earlies age-hardenable alloy types [11] and is one of the most studied.

2.2 Precipitation hardening

Not all properties depend only on the element composition. Most mechanical properties also depend on the microstructure of the alloy. The microstructure can be altered by different methods of treatment. There are mainly five treatments commercial alloys go through; namely casting, annealing (homogenisation), cold-work, solutioning and age hardening. Upon annealing the alloy becomes a solid solution of homogeneous phase, α. What happens when you cool it down is that the solubility of the added alloying metal drops until the host metal becomes supersaturated at temperature, T_m. Then precipitation starts and small nuclei of a new phase, β starts to grow in the solid solution, according to the reaction

$$\alpha\left(x^0\right) = \alpha\left(x^\alpha\right) + \beta\left(x^\beta\right).$$

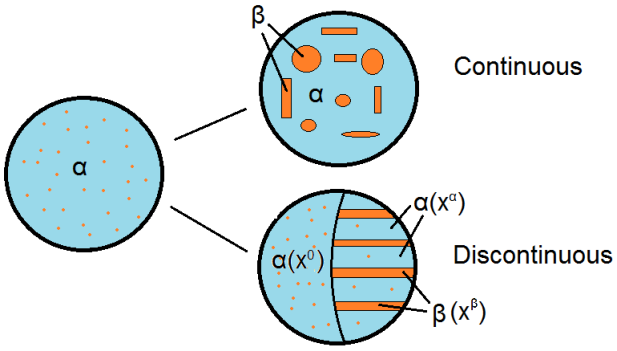


Figure 2.1: The two nucleation-modes in a solid solution. Continous mode precipitates has the same composition. Discontinous mode alters the composition in both precipitate and matrix.

Here x^0 is the initial composition of the α-phase, x^α is the composition of α-phase after precipitation starts, and x^β is the composition of the β-phase.

There are two types of modes by which the precipitation reaction occur. It is *Continous* and *Discontinous*. In the continous mode the β phase grows as discrete particles uniformly spread in the α phase, and the compositions are the same ($x^\alpha = x^0$). In the discontinous mode the β phase grows as plates or needles with alternating α phase surrounding it with a new composition ($x^\alpha \neq x^0$). Both modes are illustrated in figure 2.1. In the case of which the precipitates are thin plates, they usually grow in a specific set of {hkl} planes in the matrix, called habit-planes. Similarly, needles will lie in their habit directions [hkl]. This is the result of energy minimization on the interface between the matrix and the precipitate. The interfaces tend to be the close-packed planes of the crystal since they optimize the atomic matching across the interface (see section 2.4).

To understand why the interfacial energy is important we consider the case of nucleation in a liquid solution (the same arguments can be applied to any surface and any phase state). Consider a temperature ΔT below T_m of which nucleation begins. If the atoms cluster toghether to form a sphere of solid the free energy of the system will be:

$$G = V_S G_v^S + V_L G_v^L + A_{SL}\gamma_{SL}$$

where V_S is the volume of the solid sphere, V_L the volume of the liquid, A_{SL} the solid/liquid interface area, $G_v^{S,L}$ are free energy per unit volume, and γ_{SL} is the solid/liquid

interface energy. The free energy of the system prior to nucleation is

$$G_0 = (V_S + V_L)\, G_v^L.$$

The formation of the solid sphere is then

$$\Delta G = G - G_0 = -\frac{4}{3}\pi r^3 \Delta G_v + 4\pi r^2 \gamma SL \tag{2.1}$$

where $\Delta G = G_v^L - G_v^S$ and γ_{SL} is positive. One can show [12] that

$$\Delta G_v = \frac{L_v \Delta T}{T_m}$$

where L_v is the latent heat of the phase transformation per volume. If substituted into 2.1 and minimized for r we get

$$r^* = \frac{2\gamma_{SL} T_m}{L_v} \frac{1}{\Delta T} \tag{2.2}$$

and

$$\Delta G^* = \frac{16\pi \gamma_{SL}^3 T_m^2}{3L_v^2} \frac{1}{(\Delta T)^2}. \tag{2.3}$$

This is the minimum size and energy needed for the nucleus to continue to grow. It's a barrier they have to overcome. Note how they decrease with increasing undercoling ΔT. To understand why nucleation occure, we can consider the creation of a critical nucleus to be thermally activated, while the activation-energy is ΔG^*. The probability of achieving this energy is proportional to the number of clusters of radius r^* which is proportional to $\exp\{-\Delta G^*/k_B T\}$. Therefore one can show[12] that the nucleation will become possible only if ΔG^* is below $\sim 78 k_B T$. This type of nucleation is called homogeneous because it happens uniformely in the middle of the phase.

For the nucleation to be made easier, the total interface energy need to be reduced. It is possible by nucleating upon a pre-existing interface like grain boundaries, where γ is lower. This type of nucleation is called heterogeneous. Another way of lowering the free energy is to put something on the interface that make nucleation and lattice fitting easier. Here is where the adding elements come in.

2.3 The Al-Cu Alloy System

If you take a solid solution and drop the temperature sufficiently under the solvus temperature one starts to see precipitates forming. But before the equillibium shape is attained, normally one or two metastable phases form before the stable phase appears. In the Al-Cu alloys one see the following sequence of precipitates:

$$\text{GP zones} \rightarrow \theta'' \rightarrow \theta' \rightarrow \theta$$

GP zones are the first constructions to form and have the shape of thin disks with a diameter about 80 Å [13]. The composition of the GP-zones is about 90% copper and 10% aluminium, and they form homogeneously in the Al matrix. They form very rapidly, normally even when quenching (dropping the temperature fast) to avoid precipitation forming.

The $\boldsymbol{\theta''}$**−precipitate** has a tetragonal structure with a = b = 4.0 Å and c = 7.8 Å and is a plate with thickness about 20 Å and length about 300 Å. It has a coherent interface and therefore lots of strain around the long side of the precipitate. It forms homogeneously in the matrix.

The $\boldsymbol{\theta'}$**-precipitate** is also tetragonal with parameters a = 4.04 Å and c = 5.8 Å and takes form as plates. It can be seen in light microscopes because of its length of about 1000 Å = 1 µm. The shortside interface is semicoherent with the matrix. It forms heterogenously at low ageing-temperatures, on helical dislocations [14].

The $\boldsymbol{\theta}$**-precipitate** is the equillibrium precipitate and has a tetragonal structure with parameters a = 6.06 Å and c = 4.87 Å. The interface with the matrix is incoherent and it nucleates heterogeneously at grain boundaries.

The θ'' and the θ'-phases precipitate on the $\{100\}_{Al}$ planes as thin plates. The main strengthening of quenched Al-Cu alloys happens prior to the formation of θ' (as seen in figure 1.2) and indicates that the coherent interfaces interact with the dislocations and prevents them from propagating through the material.

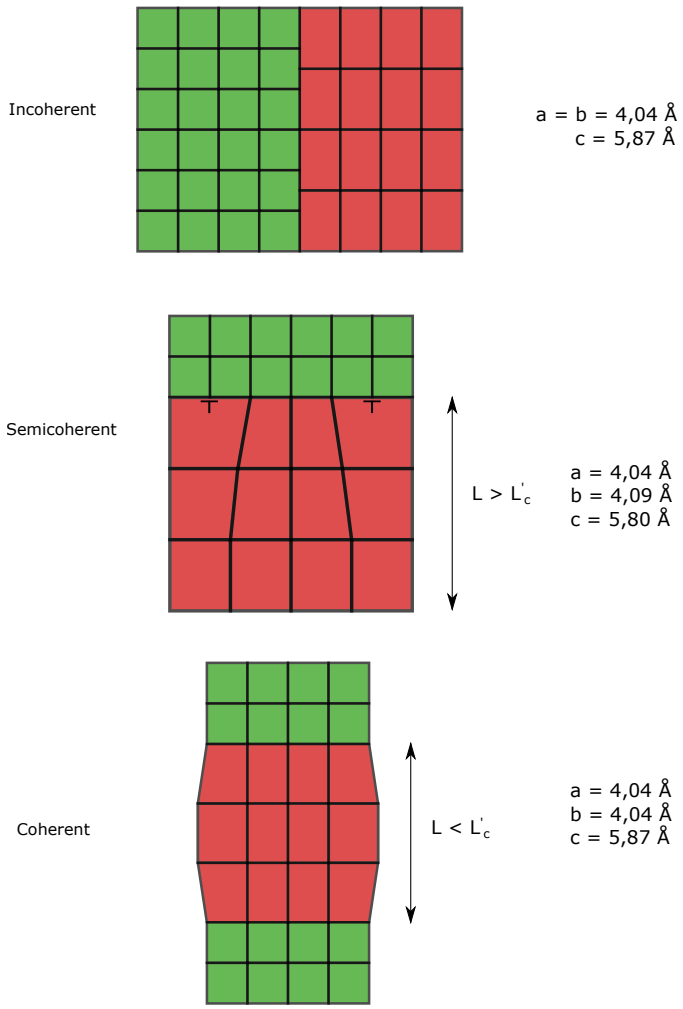


Figure 2.2: The three types of interfaces, incoherent, the semicoherent, and the special case coherent. The transition between the last two happens at the critickal thickness, L_c'

2.4 Interface coherency

When two phases are present they will have an interface that raise the total energy of the crystal. The chemical bondings between two different chemical elements is often weaker than the bonding between same type of elements, or the different structures of the phases can reduce the number of bondings across the interface. The free energy from the mismatch can be divided in two parts. The chemical interface energy (in bondings) and the strain energy (from lattice distortions). When the two phases possess different lattice parameters the atomic coloumns will not have entirely perfect match. It can be solved if the lattice of both phases stretches to match over the interface. When doing this one can obtain semi-coherency (see figure 2.2). When the two lattice parameters are fairly similar in lengths, the interface is coherent. This mean that we have complete matching between atoms across the interface. For a coherent interface the only contibution to the interface energy is the chemical, γ_{ch}, which comes from the difference in binding energy between the two phases.

If the lattice parameters of the unstrained equillibrium precipitate phase and matrix are a'_β and a_α respectively, the *unconstrained misfit d* is

$$d = \frac{a'_\beta - a_\alpha}{a_\alpha} \tag{2.4}$$

where $a'_\beta > a_\alpha$. This is called the disregistry and decides the length between dislocations, D. The dislocation length can easily be shown to be

$$D = \frac{a'_\beta}{\delta} = \frac{a_\alpha a'_\beta}{a_\alpha - a'_\beta}. \tag{2.5}$$

When the dislocation length is longer than 1000 Å one say the interface is coherent (it is approximately the length of a θ'-precipitate). When $d > 0.25$, i.e., one dislocation for every four interplanar spacings, the interface turns to be incoherent.

A semi-coherent interface can in some cases be a coherent interface. If the energy caused by strain from slight misfit is smaller than the formation energy of an edge dislocation the interface becomes coherent. This can happen in a precipitate which is thiner than the critical thickness L'_c after which the interface becomes semi-coherent[15].

Chapter 3

Electronic structure calculations

Density functional theory (DFT) has long been the mainstay of electronic structure calculations in solid-state physics. In the 1990's it became very popular in quantum chemistry. This is because approximate functionals were shown to provide a useful balance between accuracy and computational cost. This allowed much larger systems to be treated than by traditional ab initio methods, while retaining much of their accuracy. DFT is not just another way of solving the Schrödinger equation. Density functional theory is a completely different way of approaching any interacting problem, by mapping it exactly to a much easier-to-solve non-interacting problem.

In this chapter the main principles and equations that led up to the birth of DFT is presented systematically. We start with the several approximations, and simplifications. Then we mention functionals and the Kohn-Sham equation is presented. The approximation of the potentials are explained and a possible extension to finite temperature is discussed.

3.1 Introduction

Given a system of particles consisting of atoms with nucleons and electrons, according to the theory of quantum mechanics, the physical state of this system can be described by the Schrödinger equation:

$$H\Psi\left(\{\mathbf{R}_I;\mathbf{r}_i\}\right) = E\Psi\left(\{\mathbf{R}_I;\mathbf{r}_i\}\right), \tag{3.1}$$

where Ψ is the quantum mechanical wave function, E is the energy and $\mathbf{R}_i$ and $\mathbf{r}_i$ is the positions of the ions and the electrons respectively. H is the Hamiltonian of the system:

$$\begin{aligned} H = &-\sum_i \frac{\hbar^2}{2m_e}\nabla^2_{\mathbf{r}_i} - \sum_I \frac{\hbar^2}{2M_Z}\nabla^2_{\mathbf{R}_i} - \frac{1}{4\pi\epsilon_0}\sum_{I,i}\frac{Z_I e^2}{|\mathbf{R}_I - \mathbf{r}_i|} \\ &+ \frac{1}{4\pi\epsilon_0}\sum_{i,j>i}\frac{e^2}{\left|\mathbf{r}_i - \mathbf{r}_j\right|} + \frac{1}{4\pi\epsilon_0}\sum_{I,J>I}\frac{Z_I Z_J e^2}{\left|\mathbf{R}_I - \mathbf{R}_J\right|} \end{aligned} \tag{3.2}$$

The first to terms represent the kinetic energy of the electrons and nucleons, T_e and T_n. The third term represents the electrostatic attraction between the electrons and nuclei, V_{ne}. The fourth term represents the electrostatic repulsion between the electrons, V_{ee}, and the last term between the nuclei, V_{nn}. m_e is the mass of the electrons, and M_Z the mass of the cores. Z_I is the number of protons in each core, I.

3.1.1 The Born Oppenheimer Approximation

Since the mass of an electron is so small compared to the nuclei, electrons will move much faster than the cores and very rapidly find its ground state configuration. The fact that they relax almost instantly is called “The adiabatic approximation”. In this case the dynamics of the electrons won't be affected by the movement of the cores, and relative to the electrons one can regard the positions of the nuclei as static. We can then neglect the kinetic energy T_n of the cores. Also, since the cores are viewed as fixed, the term V_{nn} will be constant and only shift the eigenvalues by a constant, therefore it is normally left out of the equation. We then get this effective Hamiltonian for the electrons:

$$H_e(\mathbf{r},\mathbf{R}) = -\frac{1}{2}\sum_i \nabla^2_{\mathbf{r}_i} - \sum_{I,i}\frac{Z_I}{|\mathbf{R}_I - \mathbf{r}_i|} + \sum_{i,j>i}\frac{1}{\left|\mathbf{r}_i - \mathbf{r}_j\right|} \tag{3.3}$$

where we have introduced Hartree atomic units: $e = m_e = \hbar = 4\pi\epsilon = 1$. The energy unit is now in Hartrees (1H = 27.2114 eV= 627.5 kcal/mol) and all distances are given in Bohr radii (a_0 = 0.529Å). By using the wave function $\Psi(\mathbf{r},\mathbf{R}) = \phi(\mathbf{r},\mathbf{R})\phi(\mathbf{R})$ and solving the Schrödinger equation

$$H_e(\mathbf{r},\mathbf{R})\phi(\mathbf{r},\mathbf{R}) = E_e(\mathbf{R})\phi(\mathbf{r},\mathbf{R}) \tag{3.4}$$

with static $\mathbf{R}$ one obtain a potential energy surface for the electrons, $E_e(\mathbf{R})$. We then reintroduce the nuclear kinetic energy to solve the Schrödinger equation with the

nuclear motion:

$$[T_n + E_e(\mathbf{R})]\,\phi(\mathbf{R}) = E\phi(\mathbf{R}) \tag{3.5}$$

where $\phi(\mathbf{R})$ is the wave function of the nucleus, and E is the total energy of the system. These two equations are called the Born Oppenheimer approximation [16]. It is still hard to solve the equations because of the coupling between the electrons, V_{ee}, in the last term of equation 3.3. This can be circumfered by assuming the electrons are single particle states subjected to an average force from all the other electrons, a mean-field approximation proposed by Douglas Hartree in 1928[17].

3.1.2 The Hartree Approximation

In the Hartree approximation [17] one assumes that the electronic wavefunction can be expressed as a product of non-interacting single particle wave functions:

$$\Psi(\{\mathbf{r}_i\}) = \prod_i^N \phi_i(\mathbf{r}_i), \tag{3.6}$$

where $\phi_i(\mathbf{r}_i)$ are the single particle wave functions and N is the number of electrons. The total energy of the system will be

$$\begin{aligned} E &= \langle\Psi|H|\Psi\rangle \\ &= -\sum_i \left\langle\phi_i\right|\frac{1}{2}\nabla^2_{\mathbf{r}} + \sum_I \frac{Z_I}{|\mathbf{R}_I - \mathbf{r}_i|}\left|\phi_i\right\rangle + \sum_{i,j>i}\left\langle\phi_i\phi_j\right|\frac{1}{\left|\mathbf{r}_i - \mathbf{r}_j\right|}\left|\phi_i\phi_j\right\rangle \end{aligned} \tag{3.7}$$

where we use Dirac's braket-notation for simplicity. We apply the variational principle [1] to get the single particle eigenvalue equations

$$\left[-\frac{1}{2}\nabla^2_{\mathbf{r}} - \sum_I \frac{Z_I}{|\mathbf{R}_I - \mathbf{r}_i|} + \sum_{i\neq j}\left\langle\phi_j\right|\frac{1}{\left|\mathbf{r}_i - \mathbf{r}_j\right|}\left|\phi_j\right\rangle\right]\phi_i(\mathbf{r}) = \epsilon_i\phi_i(\mathbf{r}), \tag{3.8}$$

where ϵ_i represent the eignevalues. The third term inside the brackets is the Hartree potential, V^H_{ee}. This equation has the form of a nonlinear set of equations for the eigenfunctions $\{\phi_i\}$, and is usually solved by iterative methods.

[1] Refere to a textbook in basic Quantum mechanics ex: Griffiths: "Introduction to Quantum Mechanics" (1995)

3.1.3 The Hartree-Fock Approximation

The next step is to take into account the exchange interacion between the electrons. For fermionic systems, when two like particles exchange states, the total wave function changes sign. In order to incorporate this into the equations, we introduce a "Slater-determinant" [18] for the wave functions:

$$\Psi(\{\mathbf{r}_i\}) = \frac{1}{\sqrt{N!}} \begin{vmatrix} \phi_1(\mathbf{r}_1) & \phi_1(\mathbf{r}_2) & \cdots & \phi_1(\mathbf{r}_N) \\ \phi_2(\mathbf{r}_1) & \phi_2(\mathbf{r}_2) & \cdots & \phi_2(\mathbf{r}_N) \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ \phi_N(\mathbf{r}_1) & \phi_N(\mathbf{r}_2) & \cdots & \phi_N(\mathbf{r}_N) \end{vmatrix} \tag{3.9}$$

where N is the number of electrons. This form has the fermionic property, since interchanging two single particle states is equivalent to interchanging two rows in the determinant, which leads to a change in sign for the total wave function. It also makes it impossible for two particles to be at the same location, since this would make the determinant zero. One can use the same formalism as for the Hartree approximation to get the total energy:

$$\begin{aligned} E &= \langle \Psi | H | \Psi \rangle \\ &= -\sum_i \left\langle \phi_i \right| \frac{1}{2}\nabla_{\mathbf{r}}^2 + \sum_I \frac{Z_I}{|\mathbf{R}_I - \mathbf{r}_i|} \left| \phi_i \right\rangle \\ &\quad + \frac{1}{2} \sum_{ij(i\neq j)} \left\langle \phi_i \phi_j \right| \frac{1}{\left|\mathbf{r}_i - \mathbf{r}_j\right|} \left| \phi_i \phi_j \right\rangle - \frac{1}{2} \sum_{ij(i\neq j)} \left\langle \phi_i \phi_j \right| \frac{1}{\left|\mathbf{r}_i - \mathbf{r}_j\right|} \left| \phi_j \phi_i \right\rangle \end{aligned} \tag{3.10}$$

and the single particle eigenvalue equations

$$\left[-\frac{1}{2}\nabla_{\mathbf{r}}^2 - \sum_I \frac{Z_I}{|\mathbf{R}_I - \mathbf{r}_i|} + \sum_{i\neq j} \left\langle \phi_j \right| \frac{1}{\left|\mathbf{r}_i - \mathbf{r}_j\right|} \left| \phi_j \right\rangle \right] \phi_i(\mathbf{r}) - \sum_{i\neq j} \left\langle \phi_j \right| \frac{1}{\left|\mathbf{r}_i - \mathbf{r}_j\right|} \left| \phi_i \right\rangle \phi_j(\mathbf{r}) = \epsilon_i \phi_i(\mathbf{r}) \tag{3.11}$$

To get the last term on the form $V_i^X(\mathbf{r})\phi_i(\mathbf{r})$, we will need to introduce the single particle exchange density

$$n_i^X(\mathbf{r}', \mathbf{r}) = \sum_{i\neq j} \frac{\phi_i(\mathbf{r}')\phi_i^*(\mathbf{r})\phi_j(\mathbf{r})\phi_j^*(\mathbf{r}')}{\phi_i(\mathbf{r})\phi_i^*(\mathbf{r})} \tag{3.12}$$

from which we can construct the exchange potential

$$V_i^X(\mathbf{r}) = -\int \frac{n_i^X(\mathbf{r}', \mathbf{r})}{|\mathbf{r} - \mathbf{r}'|} \mathrm{d}\mathbf{r}' \tag{3.13}$$

and we get the usual form for a system of eigen equations

$$\left[T_e(\mathbf{r}) + V_e n(\mathbf{r}) + V_{ee}^H(\mathbf{r}) + V_i^X(\mathbf{r})\right] \phi_i(\mathbf{r}) = \epsilon_i \phi_i(\mathbf{r}) \tag{3.14}$$

This is the Hartree-Fock approximation [19]. The effect of the exchange interaction is to ensure that no two electrons occupy the same single particle eigenstate. Similar to the Hartree potential, the exchange potential is repulsive. One of the complications of this exchange potential is that each state, ϕ_i, sees a different effective potential, because they are a function of i.

3.2 Density Functional Theory

Although the Hartree-Fock method is commonly used for atomic and molecular calculations, it is not easily applied to solid state calculations, mainly because of the complicated form of the exchange interaction 3.13. In a series of seminal papers in the 60s, Hohenberg, Kohn and Sham started development of Density Functional Theory (DFT), a method which in an ingenious way resolves some of these problems. Instead of beginning with an explicit expression for the wave function, as in the Hartree-Fock method, DFT is based on a reformulation of the equations depending on the electron density.

3.2.1 Many-body problems

When there is more than one particle in the system, the Hamiltonian and the wave function include one coordinate for each particle. Furthermore, electrons have two possible spin states, up or down, and so the wave function is a function both of spatial coordinates and spin coordinates. A general principle of many-electron quantum mechanics is that the wave function must be antisymmetric under interchange of any two sets of coordinates [20]. In such a quantum system, the repeated interactions between particles create quantum correlations, or entanglement[21]. As a consequence,

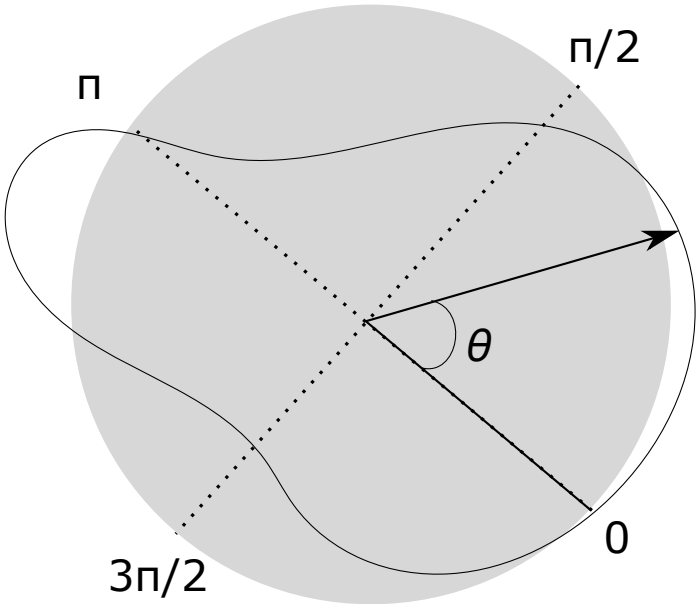


Figure 3.1: A 2D curve which is generated by a function $r = r(\theta)$

the wave function of the system is a complicated object holding a large amount of information, which usually makes exact or analytical calculations impractical or even impossible.

3.2.2 Functionals

A function maps one number to another. A functional assigns a number to a *function*. For example, consider all functions $r(\theta)$, $0 \leq \theta \leq 2\pi$, which are periodic, i.e., $r(\theta + 2\pi) = r(\theta)$. Such functions describe shapes in two-dimensions such as in fig 3.1. For every such curve, we can define A as the area enclosed by it. One write $A[r(\theta)]$ to indicate this functional dependence. Note that, even if we don't know the relation explicitly, we do know it exists. A functional is called local when one needs only to know the function right at a single point to evaluate the contribution to the functional from that point. On the other hand it is a semi-local functional if it depends not only on r, but $\delta r/\delta\theta$. How do you find extremes of a functional? For example, we might want to know what is the minimum area we can enclose inside a loop of string with radius larger than r_0. Thus we need to minimize $A[r]$, subject to the constraint that $r_0 \leq r$. What function $r(\theta)$ solves this?

3.2.3 Energy functional

One of the two theorems of Hohenberg and Kohn [22] is that the ground-state total energy functional of an observable with hamiltonian H, is universal and uniquely defined by V, and takes this form:

$$E[n(\mathbf{r})] = F[n(\mathbf{r})] + \int n(\mathbf{r})V(\mathbf{r})\,\mathrm{d}\mathbf{r} \tag{3.15}$$

V is the external potential, and in most cases it is the same as V_{ne} in equation 3.2. The purpose of DFT is to find the functional $[n(\mathbf{r})]F$. The first attempts of finding it was done by Thomas and Fermi independently in 1927. They used a statistical model to approximate the distribution of electrons. They assumed that electrons are distributed unifomely in space with two electrons in every h^3 of volume. With $n(\mathbf{r}) = \frac{8\pi}{3h^3}p_f^3(\mathbf{r})$ solved for p_f, and combined with the classical expression for the electron density interacion we get the functional:

$$F^{TF}[n(\mathbf{r})] = A_s\int n^{5/3}(\mathbf{r})\,\mathrm{d}\mathbf{r} + \frac{1}{2}\int\int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}\,\mathrm{d}\mathbf{r}\,\mathrm{d}\mathbf{r}' \tag{3.16}$$

with $A_s = \frac{3}{10}\left(\frac{3}{8\pi}\right)^{\frac{2}{3}}?or?\frac{3}{10}\left(3\pi^2\right)^{\frac{2}{3}}$. The Thomas-Fermi model is correct only in the limit of an infinite nuclear charge. Using the approximation for realistic systems yields poor quantitative predictions, even failing to reproduce some general features of the density such as shell structure in atoms. This is because of the inaccurate description of the kinetic energy of the electrons and the lack of a repulsive correlation term and the errors in the exchange term. Using the DFT formulation we can get the Hartree-Fock approximation by using the functional:

$$F^{HF}[n(\mathbf{r})] = \frac{1}{8}\int \frac{|\nabla n(\mathbf{r})|^2}{n(\mathbf{r})}\,\mathrm{d}\mathbf{r} + \frac{1}{4}\int\int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}\,\mathrm{d}\mathbf{r}\,\mathrm{d}\mathbf{r}' \tag{3.17}$$

3.2.4 Kohn-Sham Equations

With the HK-definition 3.15 and using the Born Oppenheimer approximation, one introduce a set of fictive non-interacting fermionic particles with single particle orbitals

called Kohn-Sham orbitals, $\phi_i(\mathbf{r})$ related to the electron density $n(\mathbf{r})$ by

$$n(\mathbf{r}) = \sum_i f_i \left|\phi_i(\mathbf{r})\right|^2 \tag{3.18}$$

where f_i is the occupation number of state i. If we express the many-body wave function, $\Psi(\{\mathbf{r}_i\})$, in the form of a Slater determinant of the Kohn-Sham orbitals, we arrive upon the following expression for $F[n(\mathbf{r})]$ of equation 3.15:

$$F[n(\mathbf{r})] = T[n(\mathbf{r})] + \frac{1}{2}\int\int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}\,\mathrm{d}\mathbf{r}\,\mathrm{d}\mathbf{r}' + E^{XC}[n(\mathbf{r})] \tag{3.19}$$

where

$$T[n(\mathbf{r})] = -\sum_i \left\langle \phi_i \middle| \frac{1}{2}\nabla^2_{\mathbf{r}} \middle| \phi_i \right\rangle \tag{3.20}$$

is the Kohn-Sham kinetic energy. Applying the variational method to minimize the energyfunctional, we obtain the single particle eigenequations:

$$\left[-\frac{1}{2}\nabla^2_{\mathbf{r}} + V^{KS}[\mathbf{r}, n(\mathbf{r})]\right]\phi_i(\mathbf{r}) = \epsilon_i\phi_i(\mathbf{r}) \tag{3.21}$$

with

$$V^{KS}[\mathbf{r}, n(\mathbf{r})] = V(\mathbf{r}) + \int \frac{n(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|}\,\mathrm{d}\mathbf{r}' + \frac{\delta E^{XC}[n(\mathbf{r})]}{\delta n(\mathbf{r})} \tag{3.22}$$

The first term is the external potential, the second term is the single particle exchange potential, from Hartree-Fock theory (3.13). The last term is a potential representing the many-body effects of the system in the form of a functional derivative of the exchange-correlation energy, E^{XC}, with respect to the density, $n(\mathbf{r})$. Equations 3.21 and 3.22 together with the expression for the density (3.18) are collectively known as the Kohn-Sham equations [23]. Since V^{KS} depends on $n(\mathbf{r})$, which depends on the ϕ_i, which in turn depend on V^{KS}, the problem of solving the Kohn–Sham equation has to be done in a self-consistent (i.e., iterative) way. Usually one starts with an initial guess for $n(\mathbf{r})$, then calculates the corresponding V^{KS} and solves the Kohn–Sham equations for the ϕ_i. From these one calculates a new density and starts again. This procedure is then repeated until convergence is reached. A non-iterative approximate formulation called Harris functional DFT [24] is an alternative approach to this.

3.2.5 Exchange-correlation potential

In general the exchange-correlation potential (V^{XC}) can be written as:

$$\frac{\delta E^{XC}[n(\mathbf{r})]}{\delta n(\mathbf{r})} = f\left[n(\mathbf{r}), \nabla n(\mathbf{r}), \nabla(\nabla n(\mathbf{r})), \ldots\right] \tag{3.23}$$

This means that the XC-potential, which is a functional derivative of E^{XC} with respect to the local density, $n(\mathbf{r})$, will not only depend on the value of the density at $\mathbf{r}$, but also on the gradients of the density at $\mathbf{r}$. Here the first approximation in DFT comes in.

3.2.6 LDA and GGA approximations

For a homogeneous electron gas, the gradient terms in 3.23 will vanish. In the Local Density Approximation (LDA), one assumes the following form for the XC-energy:

$$E^{XC}[n(\mathbf{r})] = \int n(\mathbf{r})\epsilon^{xc}[n(\mathbf{r})]\,\mathrm{d}\mathbf{r} \tag{3.24}$$

where ϵ^{xc} is the XC-energy per particle of a homogeneous electron gas with density $n(\mathbf{r})$. This amounts to assuming that for each point, $\mathbf{r}$, with density $n(\mathbf{r})$, the XC-energy is the same as for a homogeneous electron gas with a constant density, n. The XC-energy density is decomposed in exchange and correlation terms linearly. The exchange term is found analytically, but the correlation isn't known analytically. However very accurate quantum Monte Carlo simulations have been done for various densities, and the results are interpolated in different ways, resulting in many LDA methods. A natural extension of the LDA is the inclusion of terms depending on the derivatives of n. When including the gradient of the density one usually get very accurate solutions for the ground state energies and molecular geometries[25]. We get the Generalized Gradient Approximation (GGA):

$$E^{XC}[n(\mathbf{r})] = \int n(\mathbf{r})\epsilon^{xc}[n(\mathbf{r}), \nabla n(\mathbf{r})]\,\mathrm{d}\mathbf{r} \tag{3.25}$$

GGA[26] in general performs better than LDA. One example being that LDA frequently underestimates the lattice parameters and bulk modulus by a few percent with respect to GGA and experimentally derived values in solid state calculations [27]. Potentially more accurate solutions can be obtained by including the second derivative of the density (the Laplacian), and some use hybrid functionals, where the exchange part

of the energy is found by solving Hartree-Fock equation (3.14). Some popular LDA and GGA methods are Perdew-Zunger (PZ81)[28] and Perdew-Wang (PW92)[29] and Perdew-Burke-Ernzerhof (PBE)(1996)[30] to name a few.

3.3 Solving the Kohn-Sham Equations

There are tons of algorithms for solving the various steps of the equations and also several choise of basis set for the single particle wave functions, $\{\phi_i(\mathbf{r})\}$. The main issues is that with an infinite system, the number of wave functions becomes infinite. And since the electronic wave functions have to extend over the entire space, with the system being infinite, an infinite basis set is required to represent it. To overcome this we apply periodic boundary conditions and use Bloch waves to represent our ionic structure.

3.3.1 Pseudopotentials

The many electron Schrödinger equation can be very much simplified if electrons are divided in two groups: valence electrons and inner core electrons. The electrons in the inner shells are strongly bound and do not play a significant role in the chemical binding of atoms; they also partially screen the nucleus, thus forming with the nucleus an almost inert core. Binding properties are almost completely due to the valence electrons, especially in metals and semiconductors. This separation suggests that inner electrons can be ignored in a large number of cases, thereby reducing the atom to an ionic core that interacts with the valence electrons. The use of an effective interaction, a pseudopotential, that approximates the potential felt by the valence electrons, was first proposed by Hellmann in 1935 [31] and Fermi in 1936 [32].

Norm-conserving and ultrasoft are the two most common forms of pseudopotential used in modern plane-wave electronic structure codes [33]. They allow a basis-set with a significantly lower cut-off (the frequency of the highest Fourier mode) to be used to describe the electron wave functions and so allow proper numerical convergence with reasonable computing resources. An alternative would be to augment the basis set around nuclei with atomic-like functions, as is done in Linear Augmented-Plane-Wave methods (LAPW), first proposed by Slater in 1937[34].

The projector augmented wave method (PAW)[35] is a technique used in ab initio

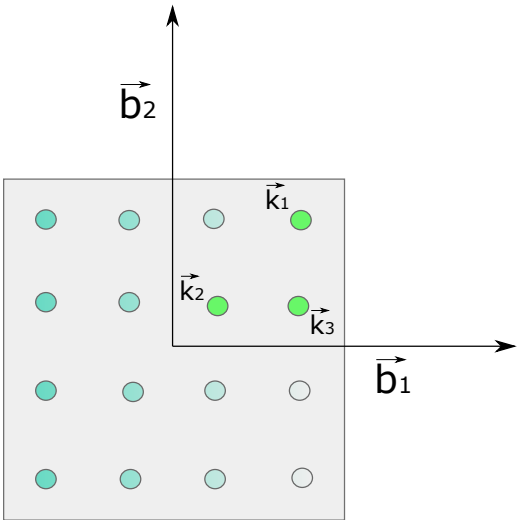


Figure 3.2: The three inequivalent k-points in a quadratic 2-dimensional lattice with 4x4 point density.

electronic structure calculations. It is a generalization of the pseudopotential and LAPW, and allows for density functional theory calculations to be performed with greater computational efficiency. The computer code used in this work implements the PAW method.

3.3.2 k-point sampling

When calculating different properties as the desity of states and electron charge density, one integrate over the Brillouin-zone, and in a computer one has to do this discretely. So one have to choose a density of k-points to sample from. The idéa of Monkhorst and Pack [36] is to use a homogeneous mesh which centers on the atom cores and use symmetry to cut down on number of terms of the integral. The Monkhorst-Pack grid is defined by:

$$\mathbf{k}(n_1, n_2, n_3) = \sum_{i=1,2,3} \frac{2n_i - N_i - 1}{2N_i} \mathbf{b}_i \tag{3.26}$$

where $n_i = 1, 2, \ldots, N_i$ and N_i is the number of k-points in i direction. $\mathbf{b}_i$ is the reciprocal lattice-vectors. As an example a quadratic 2-dimensional lattice with 4 k-points gives 16 points to evaluate. By using the symmetry of the quadratic lattice (3.2), on see that there are only three inequivalent points. Then we can approximate the integral by

$$\frac{1}{\Omega_{BZ}} \int_{BZ} F(\mathbf{k})\, \mathrm{d}\mathbf{k} \approx \frac{1}{4} F(\mathbf{k}_1) + \frac{1}{4} F(\mathbf{k}_2) + \frac{1}{2} F(\mathbf{k}_3) \tag{3.27}$$

where Ω_{BZ} is the volume of the Brillouin-zone (BZ). Thus breaking it down to only three terms as illustrated in figure 3.2.

Chapter 4

Atomistic modelling

To simulate a realistic environment including dislocations and strain from plate edges and so on, we should do the simulations on a whole precipitate with a lot of aluminium around, to adjust for misfit strains. Such a system would contain thousands of atoms. Due to restrictions in computational power, it is not possible for us to do this, because it would take such long time. It might howewer be possible to calculate on say 2000 atoms, if one was to use the whole supercompter for this specific task. But because of the busyness and nature of the supercomputer, waiting lines and limitations, we need to look for a smart solution instead. First we need to narrow our model down to a few hundred atoms. For a typical relaxation of a simple metal slice with 270 atoms, it takes about 17 hours on a 64-core's multithread computer. And the computation-time scales exponentially with number of atoms. One have to choose between good precision versus shorter computation-time. Typicaly one does a set of calculations and look at the results, does some adjustments and start them over again, then come back and look at the results, comparing them and look for convergence. It can take a long time if not experienced and used to it. Therefore we will explain how to do it.

4.1 VASP

The computer code that were used for the simulations was the Vienna Ab initio Simulation Package (VASP) [37]. It utillizes "Desity Functional Theory (DFT), solving the Kohn-Sham equations, to find an approximate solution to the many-body Schrödinger equation. The interaction between the electron charge densities is described using the

PAW method. To determine the electronic groundstate, VASP makes use of efficient iterative matrix diagonalisation techniques, like the residual minimisation method with direct inversion of the iterative subspace (RMM-DIIS)[38] or blocked Davidson algorithms[39]. The code make use of 4 input files: the POSCAR-, INCAR-, KPOINTS-, and the POTCAR-file. The POSCAR file contains the atom positions and the supercell geometry. The INCAR file holds the parameters used for the calculations such as energy cutoff and precision for the orbitals (more details in). The KPOINTS file defines a grid of k-points where the calculated values are assigned to. The POTCAR file is our choice of pseudopotential for the different atoms. The simulations are done by running the INCAR-file IN1 first for relaxation and IN2 afterwards for high precision energy output (see Appendix B for details about the files). The different running-modes are decided with the isif-tag. If set to three VASP calculates both the volume and the internal positions of the atoms. If set to two it calculates only the internal structure and keeps the volume fixed. If set to seven the volume is varying but the crystalstructure is fixed.

4.2 The supercell

The POSCAR files are made with a python package called ASE - Atomic Simulation Environment [40]. This package lets us build a supercell atom by atom, and has a graphical interface for measurement of angels and bond lengths between the atoms. To ensure reproductivity and consistency the supercells were constructed using a script written in python and using the ASE package. The script can be found in Appendix D. The models we use are cross sections of a precipitate plate seen in one of the two equivallent directions [100] or [010]. So the slab is a 2x2 aluminium atom thick rectangular rod several atoms high going from the aluminium on one side through the precipitate to the aluminium on the other side. The slab is periodically repeated in all directions during calculations. Therefore it is important that the aluminium on both sides is thick enough to avoid that the interface interact with itself. The different precipitates and their interfaces are shown in figure 4.1.

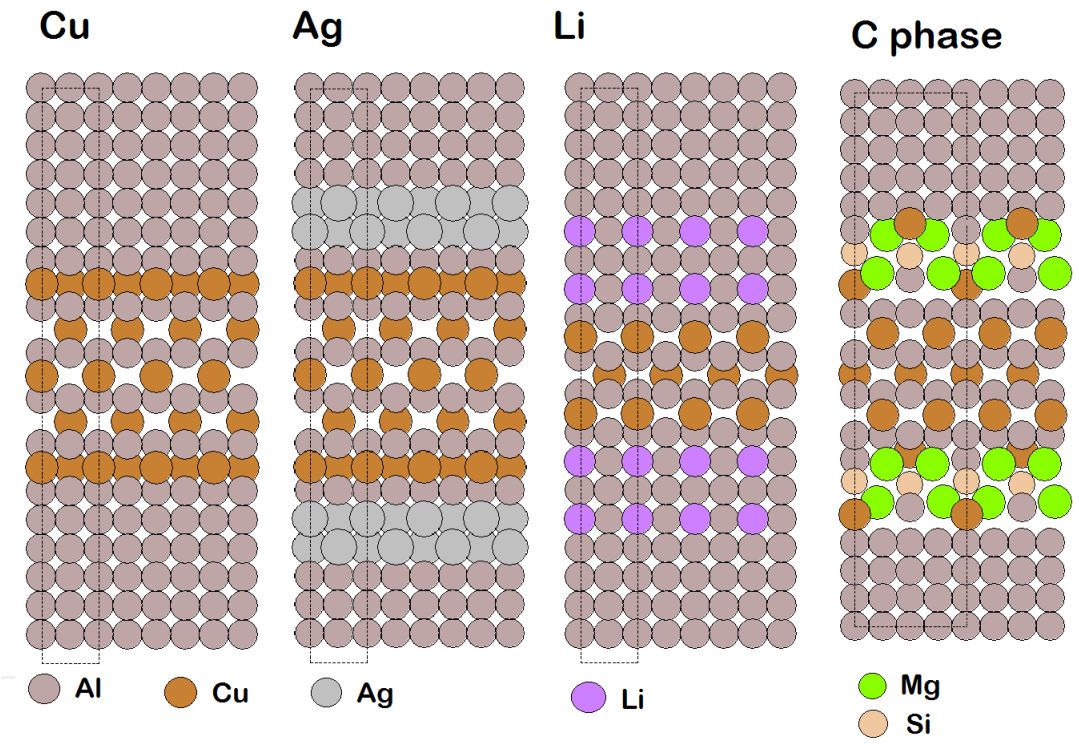


Figure 4.1: The four models used in the simulations. Al-Cu, Al-Cu-Ag, Al-Cu-Li, and Al-Cu-Mg-Si. These are plates laying in the xy-plane (001), and the drawing showing the cross-section of the plates.

4.2.1 Parameters

We need to define the several parameters that we use in the script to generate the calculation supercells. First we have the precipitate thicknes. We define it from the conventional tetragonal unit cell of θ' in figure 4.4. It has the composition Al_4Cu_2 and when we have a precipitate with this composition in the crossection that is two Cu atomlayers thick we call it a 1c thick preipitate. We can have half integer thick θ'-cells from 0.5c up to 6c in this study.

We have a matrix around the precipitate that we need to take into account, and therefore we fix the length of the slab, L to a integer number times a. The total length defines the parameter n that is the precipitate thickness in a's:

$$L = (\#Al \cdot 2 + n) \cdot a \tag{4.1}$$

where $\#Al$ is the number of Al unit cells on one side of the precipitate (see figure 4.2). Which number n to use we discuss in subsection 4.4.1.

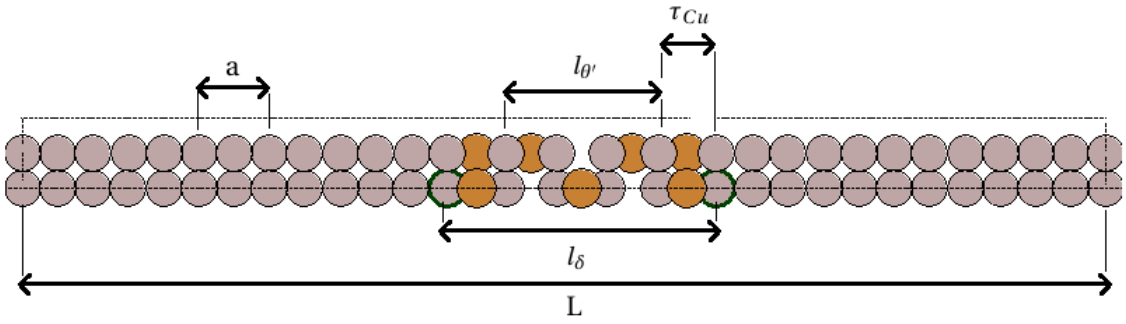


Figure 4.2: The figure marks the lengths defined in this study. In this case it is a 1.5c thick slab with normal interface (Cu) and parameters $\#Cu = 5$ and $\#Al = 6$.

4.3 Calculations

All the results must be justifyed in order to be valid for discussions. People must know where they come from, be able to reproduce them and draw their own conclusions. Therefore comes a section of definitions and equations used to produce the numbers found in the tables and figures.

4.3.1 Precipitate misfit

The tables under section 5.3 have many symbols representing different lengths in the structure. According to figure 4.2, the interface thickness is τ_x where x is the different interfaces, $x \in (Cu, Ag, Li, C)$. The reference is the thickness with no interface: τ, basically half of a, from one atom column to the next. $l_{\theta'}$ is the thickness of the precipitate with no interface. The lattice misfit is defined as:

$$\Delta l_\delta = \frac{l_\delta - n \cdot a}{n \cdot a} \tag{4.2}$$

where l_δ is the thickness of the precipitate *and* the interface after relaxation, $l_\delta = l_{\theta'} + 2 \cdot \tau_x$ (see figure 4.2). This is called the *in situ* misfit and is measured at the incoherent interface opposed to the *disregistry* which is at the coherent interface. Another misfit parameter is the c-distortion $\Delta\bar{c}$. That is the relative shift in lattice parameter c. It tells how much the precipitate has been compressed or extended. For thick precipitates a big c-distortion means a lot of strain energy absorbed because of big misfit.

4.3.2 Interface energy

Interface energy comes from a mismatch in the bondings between phase 1 and phase 2. When the interface is between two different phases there are usally three contributions to the free energy. The chemical- and lattice strain-energy, and the cohesive energy of the separate phases, E_c. To compute the interface energy, γ we must compute the total energy of the slab and subtract the constituents from it until we are left with the excess strain energy from the misfit, and the chemical interface energy. We combine them into one because the strain energy is a big portion of the total γ and pretty much affect the total free energy that controls the structure. The interface energy is defined as

$$\gamma = \frac{E_s}{2a^2}. \tag{4.3}$$

When only the pure θ' phase is present with no interface we get

$$E_s = E_{tot} - 2 \cdot \#Al \cdot E_{Al} - \#c \cdot E_{\theta'} \tag{4.4}$$

where E_{Al} is the energy of one fcc cell aluminium (four atoms), $E_{\theta'}$ is the energy of a unit cell of θ' and $\#c$ is the number of θ'-cells. When we add copper to the interface we need to subtract two extra Cu atoms:

$$E_s^{Cu} = E_{tot} - 2 \cdot \#Al \cdot E_{Al} - (\#c + 1) \cdot E_{\theta'} - 2E_{Cu}. \tag{4.5}$$

We have defined this precipitate so that its thickness is set without the two outermost Cu-layers. We then need to subtract the energy of $(\#c + 1)$ times a θ'-cell. If we add Ag outside of the Cu layer we must subtract extra two Ag unit-cells (one on each side):

$$E_s^{Ag} = E_{tot} - 2 \cdot \#Al \cdot E_{Al} - (\#c + 1) \cdot E_{\theta'} - 2E_{Cu} - 2E_{Ag} \tag{4.6}$$

C-phase at interface:

$$E_s^C = E_{tot} - 2 \cdot \#Al \cdot E_{Al} - \#c \cdot E_{\theta'} - E_{c-phase}^{\tau_C}. \tag{4.7}$$

Lithium is a special case as the script is implemented a bit odd. We subtract $(\#Al + 1)$ as that is the correct number of Al-unit cells (including Li atoms) on one side of the

precipitate. We subtract the energy of substituting two Li atoms in a mesh similar to figure 4.3:

$$E_s^{Li} = E_{tot} - 2 \cdot (\#Al + 1) \cdot E_{Al} - \#c \cdot E_{\theta'} - 2E_{Li} \tag{4.8}$$

Note that either the c-phase case or the lithium case has the copper layer as we see in Ag and pure Al-Cu case.

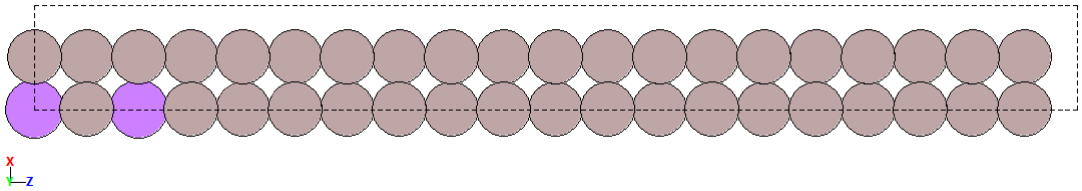


Figure 4.3: Model of $Al_{38}Li_2$ used to fin the Al-Li substitution energy, showing the positions of Lithium atoms.

4.3.3 Matrix lattice parameter

An important size to compute is the aluminium lattice constant, $a_{Al} = a$ at 0 K. This will be the structure of the matrix surrouding the precipitate, and will impose constraines on the θ' phase. It is important that this is correct to avoid track errors. Normally a will deviate from the experimental value, a^{exp} due to thermal expansion at finite temperature. Often the experimental value in reference tables are extrapolated to represent the system at 0 K. The method used to find it is by running a energy sweep over fixed volumes calculating the energy of equivallent conventional unit cells containing four atoms with fixed positions. Since we know the crystal structure is FCC we just vary the volumes ($V = a^3$) around the initial guess V^{exp} and fit the results to an equation of state, $E(V)$. Firstly done with a large span and few points just to get an estimate of were the minima is located. From this we can find an approximate minima to use in a following high-accuracy sweep, closing in at the minima. The equation of state (EOS) mostly used in material science is the Murnaghan's EOS. But we want to use the more accurate Birch-Murnaghan EOS [41]:

$$E(V) = E_0 + \frac{9V_0B_0}{16}\left\{\left[\left(\frac{V_0}{V}\right)^{\frac{2}{3}} - 1\right]^3 B_0' + \left[\left(\frac{V_0}{V}\right)^{\frac{2}{3}} - 1\right]^2 \left[6 - 4\left(\frac{V_0}{V}\right)^{\frac{2}{3}}\right]\right\} \tag{4.9}$$

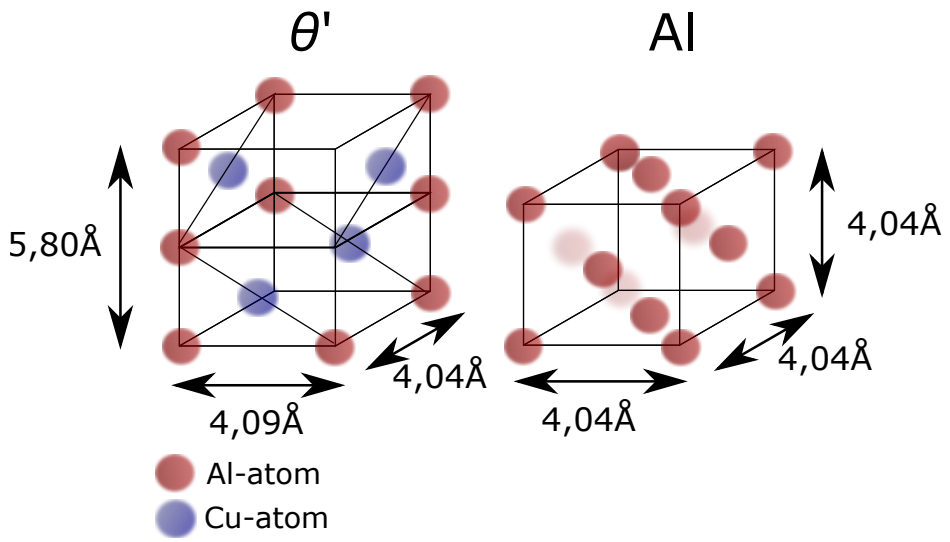


Figure 4.4: The unit cell for the precipitate, θ' and for aluminium, showing the equillibrium lattice constants. A conventional fcc-cell of aluminium contains four Al atoms. A tetragonal unit cell of θ' contains two Cu atoms and four Al atoms.

where E_0 is the equillibrium energy, V_0 is the equillibrium unit cell volume, B_0 is the bulk modulus, and B_0' is the derivative of the bulk modulus with respect to preassure. The parameters from the BM curve fitting of the data are compared to M fit parameters.

4.3.4 Cell size c-phase

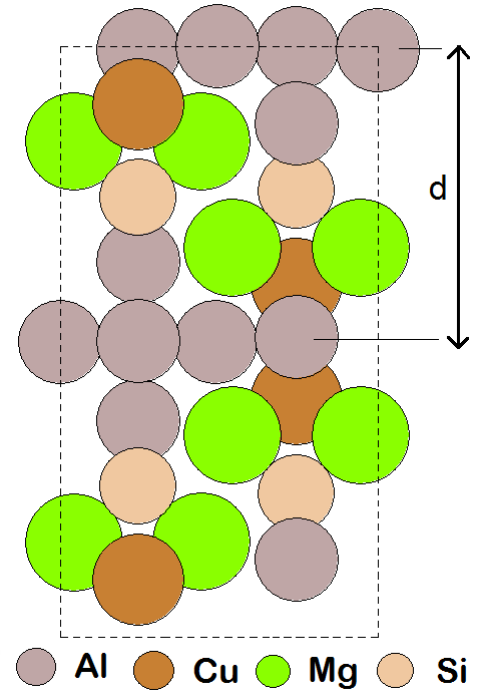


Figure 4.5: Observed structure of the interphase of Al-Cu-Mg-Si alloys.

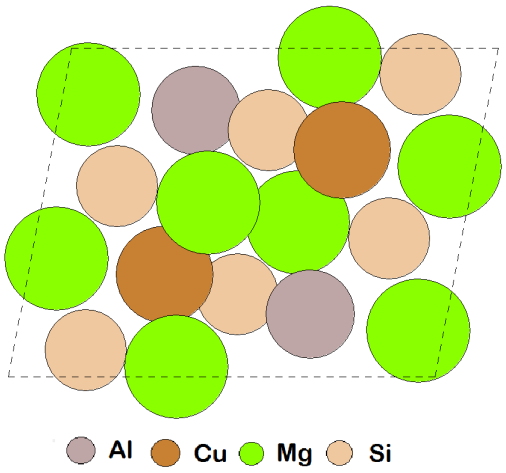


Figure 4.6: Bulk Cphase for reference. This is found by Torsæter et al. [6]

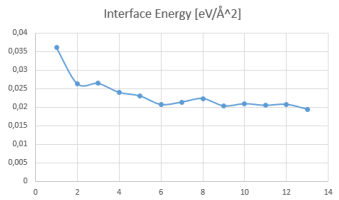
For the cphase case we need to diffrentiate between what is part of the interface-phase (interphase) and what is part of the precipitate and matrix. The model in figure 4.5 shows the structure that is observed in the interphase. It differs from the complete c-phase shown in figure 4.6. A complete c-phase has a chemical composition $Mg_8Al_2Si_6Cu_2$, but at the interface it is observed a $Mg_8Al_{12}Si_4Cu_4$ which is only a part of the bulk c-phase. The atomic positions of the interphase can be found in Appendix E. The elastic response of the partial c-phase were calculated using the different lattice constants, $d = \tau_C$ found after relaxation of the supercell slabs. The atomic positions were held fixed, and the cell doubled and mirrored to become like the model in figure 4.5. And then it was done a static energy calculation (isif=2) on the different structures and the data fit to a curve.

4.4 Convergence

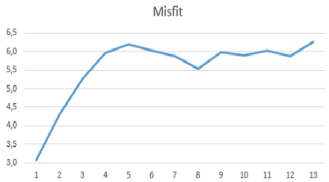
In order to secure reliable results with minimized error, we need to calibrate our tool and figure out the boundaries of our simulation. It is important to use the equillibrim parameters that we obtain with the constraines of our system and not take for granted that the experimental value can be used. Moreover we need to know the approximate error in energy from slightly wrong calculated atomic posistions.

4.4.1 Strain Effects

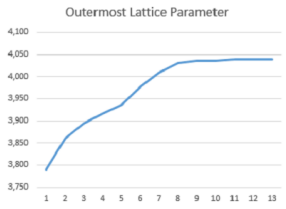
Strain is introduced when there is a force limiting free adjustment in one direction, as with the precipitate inside the aluminium matrix. If the relaxed precipitate does not match the matrix, there will be strain both in the precipitate phase and in the aluminium outside from the interface. The strainlength, l_s depends on the relative hardness between the precipitate and the aluminium. One can think of the atoms as a chain of springs. When you exdend the chain the new positions is determined by the total energy of the springs. And it is not uniform, because you have smaller springs between the atoms near the interface, so the strain is gradually decreasing out from the interface. We need to know how long our slab needs to be to avoid edge-effects that comes from when the strainlength is longer than the length of aluminium in our slab. We did a test to show how this would affect the results of interface energy and precipitate thickness. Figure 4.7 shows a slab with number of aluminium cells, $\#Al$ vaying from 1 to 13.



(a) Interface energy, γ (eVÅ^{-2})



(b) Misfit of precipitate (%)



(c) Outermost Al-cell lattice parameter, a (Å)

Figure 4.7: Strain Effects from to short slabs.

We see the interfacial energy goes up because of the strain energy in the aluminium alone. It affects the misfit reading, and we can potentially end up pointing out the wrong alloy as test winner.

Chapter 5

Results and Discussion

Results are mainly presented in tables and figures. For a overview see the "List of Figures" and "List of Tables" sections in the front. The first results are the equillibrium values for bulk aluminium. Then comes the results for the equillibrium values for θ' both as a separate phase but also as incorporated in the aluminium matrix. The same was calculated for silver. These phases are relaxed and curve-fitted and shown in separate figures in each subsection, with an associated table with parameters for each. Then comes the special interfaces with interstitial copper at the outermost layer of the precipitate, the lithium substitutions in two layers, and the partial c-phase at the interface. These five interfaces have several parameters varying with θ'-thickness. All the structural distances (described in section 4.2) are listed in separate tables under each subsection in section 5.3. The interface energies including the strainenergies are listed in separate tables under each subsection of section 5.4. Then the different values are compared in graphs and discussed if there are correlations between them. Special considerations are discussed in the end of the chapter and some conclusions drawn.

Table 5.1: Bulk properties for aluminium, copper, silver and θ'.

Crystals	Al		Cu		θ'		Ag	
Property	DFT	Exp	DFT	Exp	DFT	Exp	DFT	Exp
Lattice constant (Å)	4.0397	4.04	3.63	3.61	4.085 5.803	4.04[1] 5.80[1]	4.16	4.09
Cohesive energy (eV/at.)	-3.74	-3.39	-3.72	-3.49	-23.47	-	-2.83	-2.95
Bulk modulus (GPa)	76.92	76.93	-	137	-	-	101.6	100.7

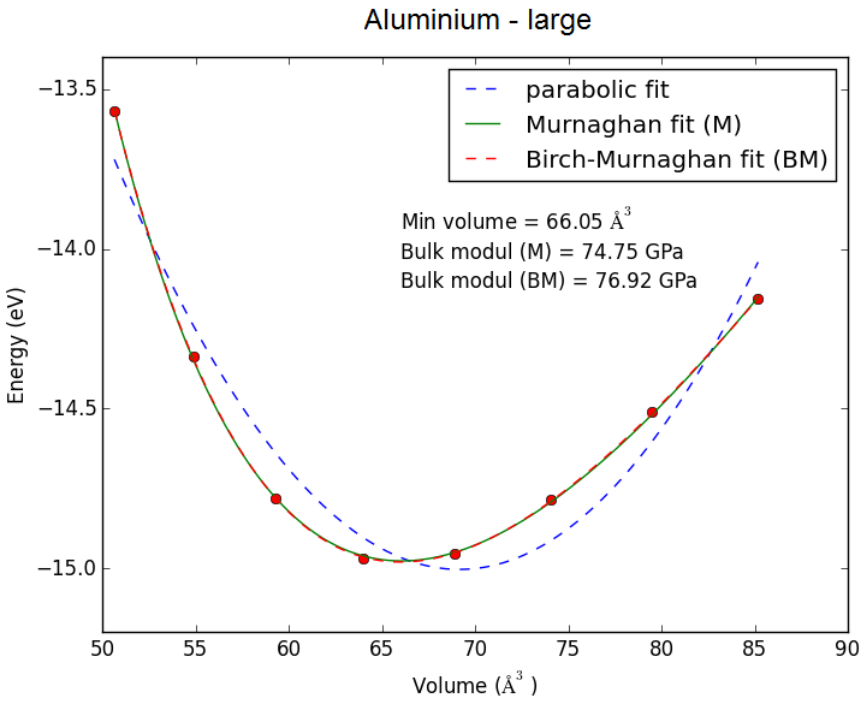


Figure 5.1: Free energy of aluminium unit cell on a large scale. One see that Birch-Murnaghan EOS fits perfectly to the data as opposed to a quadratic curve (the blue dashed line). There is not much differense between M and BM. Minimum is located near 66 Å^3

.

Table 5.2: Murnaghan and Birch-Murnaghan parameters from the big aluminium sweep dataset.

EOS	Energy (eV)	B_0 (GPa)	B_0' (-)	V $(Å^3)$	a (Å)
Murnaghan	−14.9771	74.7484	734.0703	66.0501	4.0423
Birch-Murnaghan	−14.9804	76.9221	750.1290	66.0011	4.0413

5.1 Bulk properties

The bulk properties of aluminium, copper, θ' and silver where computed according to the method described in subsection 4.3.3. The results are shown in table 5.1. They are fully relaxed fcc unit cells. When adding them in the aluminium matrix, some strain is asserted to them and the lattice parameters and total energy changes a little. They will be computed one by one in the following chapters.

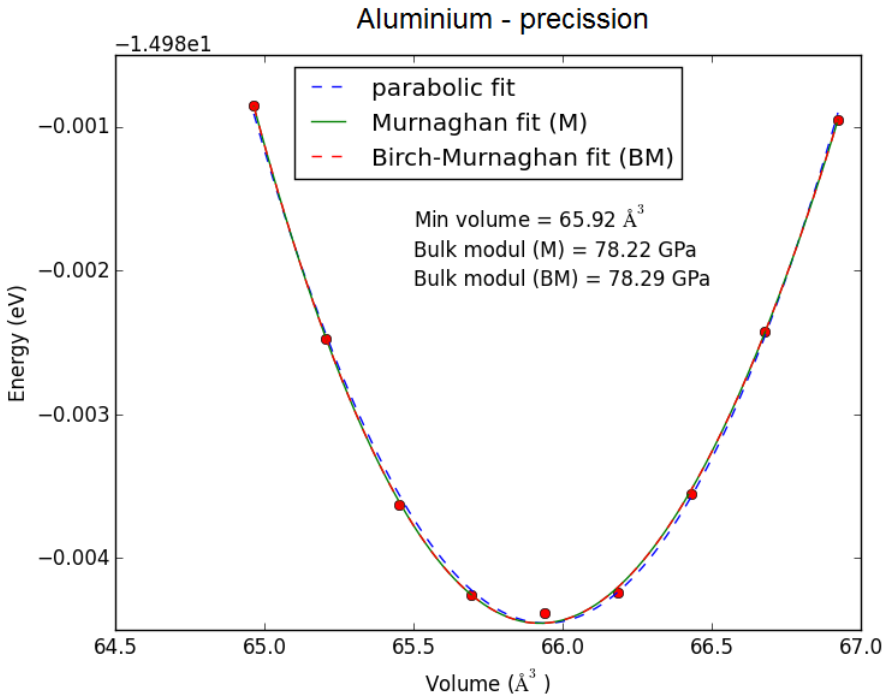


Figure 5.2: Free energy of aluminium unit cell.

5.1.1 Finding aluminium lattice parameter

Bulk aluminium, copper and silver were simulated using a conventional fcc unit cell with four atoms. A plot of the free energy per atom versus cell volume is shown for aluminium in figure 5.1. This series was done with a large intervall to get an estimation of where the minima is located. It also gives the best result for the bulk modulus. The parameters from the curvefitting of the data is shown in table 5.3. The data can be found in in table C.1 in Appendix C. The bulk modulus value of $B_0 = 76.92$ GPa is a very good value, compared to the experimental value of 76.93 GPa. The lattice parameter is near a = 4.04 Å but can be found more accurate. We can enlarge the scale and do a new sweep in a smaller intervall around 4.04 Å. Figure 5.2 shows the plot of the free energy of aluminium near the miniumum value. The data is found in table C.2 in Appendix C. The parameters from the curvefitting is shown in table 5.3.

The value a = 4.0397 Å is very close to the expected observed value of 4.04 Å at 0 K. At room temperature one can expect to see a value of 4.05 Å or more due to thermal expansion. One see that the BM curve fits much better than a quadratic plot for V's far from V_0. Near the miniumum the curve is almost quadratic.

[1] Observed in aluminium.

Table 5.3: Murnaghan and BM parameters from the aluminium small-scale sweep dataset.

EOS	Energy (eV)	B_0 (GPa)	B_0' (-)	V $(Å^3)$	a (Å)
Murnaghan	−14.9771	78.2221	1628.6751	66.0501	4.0397
Birch-Murnaghan	−14.9845	78.2911	1616.7067	65.9237	4.0397

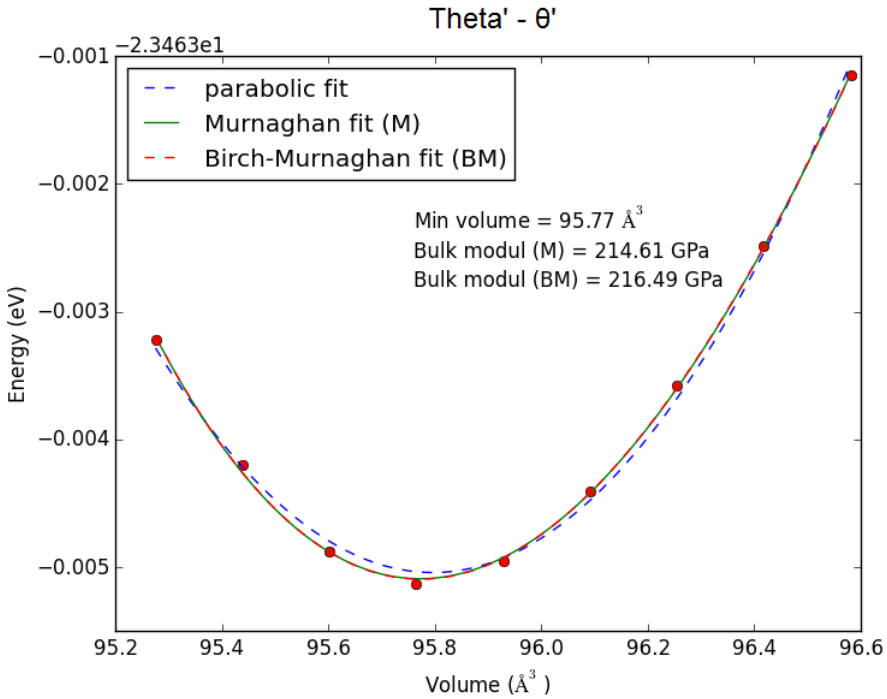


Figure 5.3: Free energy of θ' unit cell.

5.1.2 θ' lattice parameters

Table 5.4: Murnaghan and BM parameters from θ' sweep dataset.

EOS	Energy (eV)	B_0 (GPa)	B_0' (-)	V $(Å^3)$	a (Å)
Murnaghan	−23.4681	214.6138	8445.6818	95.7723	5.8704
Birch-Murnaghan	−23.4681	216.4924	8096.4235	95.7728	5.8705

The lattice parameters of the θ' precipitate were found by doing a full degree of freedom relaxation with one unit cell of Al_4Cu_2 with the ISIF = 3 tag. The result is shown in table 5.1. We see that the short-side parameter $b_{\theta'}$ is a tiny bit larger than a_{Al} and gives a disregistry of $\delta = \frac{b_{\theta'} - a_{Al}}{a_a l} = 1.1\%$ and a dislocation length of $D = \frac{a_{\theta'}}{\delta} \approx$ 367 Å, approximately 90 columns of aluminium. That means one might find three times three dislocations in a precipitate with diameter about 1 µm, given that $l_{\theta'} > L_c'$. In this study we look at thicknesses up to 6 θ' cells (6c), about 35 Å which is relatively short and

therefore we might expect the precipitate to have a completely coherent interface.

Now we keep the short parameter of θ', constrained to the matrix ($a = b$) and calculate the new long tetragonal lattice constant when in the aluminium. This will be the reference in the energy calculations. It is not implemented in VASP to fix only two sides and relax in the third, so we need to do a volum sweep, with fixed $b = a$ and varying c from 5.80 to 5.90 and isif = 2. If one was to use isif = 3 the thickness would vary freely to minimize the energy, but we want to keep it fixed to the matrix. The results are given in table C.3 and plotted in figure 5.3. The parameters from the fitting is found in table 5.4. We get a value of $c = 5.87\,\text{Å}$. The energy $E_{\theta'} = -23.4680\,\text{eV}$ is used in the interface calculations.

5.1.3 Silver parameters and relaxation

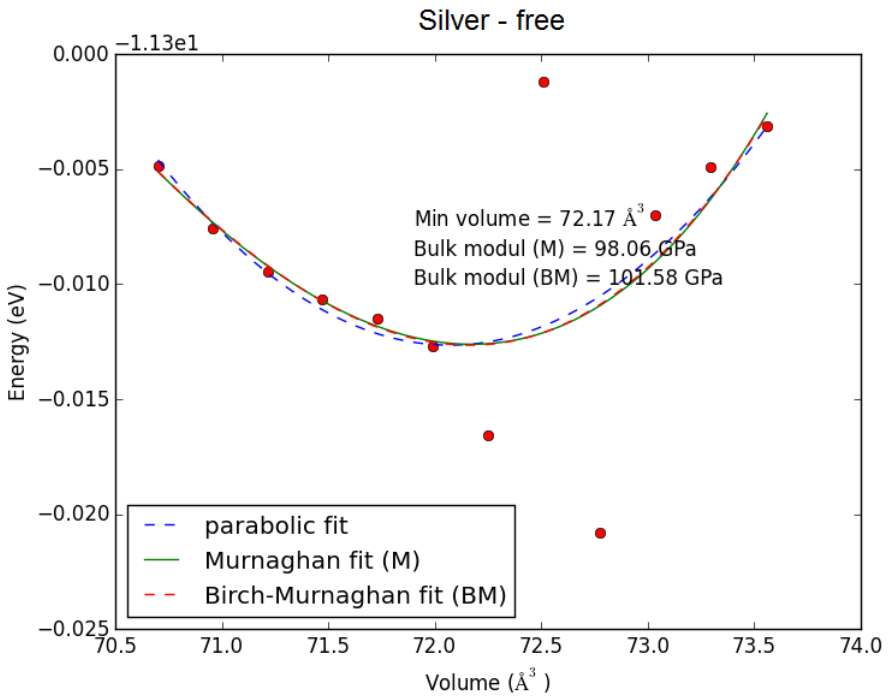


Figure 5.4: Free energy of bulk silver. Three of the points are divergent but still gives usable results.

Table 5.5: Murnaghan and BM parameters from the free silver curvefitting.

EOS	Energy (eV)	B_0 (GPa)	B_0' (-)	V (Å^3)	a (Å)
Murnaghan	−11.3126	98.0591	−5143.2412	72.1730	4.1635
Birch-Murnaghan	−11.3127	101.5809	−4601.5016	72.1647	4.1633

Bulk silver properties were computed with ISIF = 7, the same way as for Cu. The results are shown in table 5.1. It was also attempted to do a volume sweep, howewer some of the points would not converge so it affected the results somewhat. Newertheless we got the value $a_{Ag} = 4.16\,\text{Å}$ for the lattice constant of bulk silver. Figure 5.4 shows the plot of the dataset in table C.4. The bulk parameter is extracted from the curve fit with the parameters shown in table 5.5. The value of $a_{Ag} = 4.16\,\text{Å}$ is a bit larger than the experimental value $a_{Ag}^{exp} = 4.09\,\text{Å}$, in fact 1.7 % larger. But the bulk modulus of $B_0^{Ag} = 101.6\,\text{GPa}$ is almost correct. It could have been even better with a sweep over a larger intervall. The discrepancy is probably due to the choice of pseudopotential.

Now we limit the degree of freedom to one by constraining to sides to the aluminium matrix as we did for θ', and do a energy-volume sweep again. The results are shown in figure 5.5, and the parameters in table 5.6. The data from the fitting can be found in C.5. The equillibrium energy has a value of $E_{Ag} = -11.2791\,\text{eV}$. This value is used in the interfacial energy calculations in section 5.4.

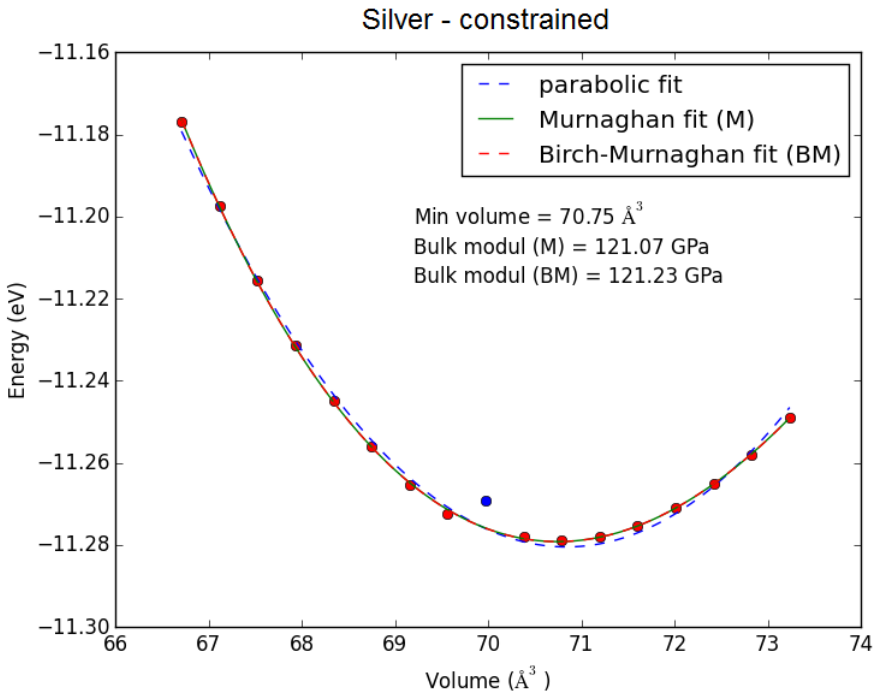


Figure 5.5: Free energy of silver, when constrained to the aluminium lattice. The blue point is a bit off the graph and is not used in the curve fitting.

Table 5.6: Murnaghan and BM parameters from the constrained silver curve fitting.

EOS	Energy (eV)	B_0 (GPa)	B_0' (-)	V (Å^3)	a (Å)
Murnaghan	−11.2792	121.0689	1166.1284	70.7469	4.3365
Birch-Murnaghan	−11.2792	121.2347	1212.1688	70.7438	4.3363

5.2 Special interfaces

The special structures of the copper interface and the lithium interface is not separate phases and must be threated differently. The copper interface is a normal Al-Cu interface with an extra interstitial Cu atom. In the Li-case some Al atoms are substituted with Li. And the distance between them are so short that they interact with each other, therefore we must calculate the difference in energy between a aluminum slab with and without Li substitutions. In the Cu-case we must take the difference between a long θ' slab with and without the extra Cu atom.

5.2.1 Adding copper to the interface

Bulk copper properties were calculated using isif=7 tag, since we already know the spacegroup. The volume relaxation gave the parameters in table 5.1. They are not used to anything but to test the accuracy of the potential. The lattice constant of a_{Cu} = 3.63 Å is fairly close to the experimental value a_{Cu}^{Exp} = 3.61 Å. We did not get the chance to do a energy sweep to find the bulk modulus.

The energy needed to put a copper atom in the interstitial position at the interface in a relaxed thetacell is calculated using a slab of four unit cells with a total of eight Cu atoms and comparing with the same slab with an extra interstitial Cu atom. The length of the slab is to ensure the extra Cu atom does not see itself in the [001] direction and interact with itself because of the periodicity of the cell. The calculation was also done with one,two and three cells as well for comparison. Table 5.7 shows convergence for the longest slab. We also use the same k-point density for the longest slab. The energy is E_{Cu} = −3.6199 eV per atom.

Table 5.7: Energy of θ' slabs with n number of unit cells stacked in [001] direction and *one* extra interstitial Cu atom in each. Also the energy of the same slabs *without* the extra Cu atom is computed.

n	k-points	$E_{n\theta'+Cu}$ (eV)	dE (eV)	$E_{n\theta'}$ (eV)	dE (eV)	E_{Cu} (eV)
1	775	−26.9911	0.2×10^{-7}	−23.4672	-0.8×10^{-10}	−3.5239
2	773	−50.5151	0.4×10^{-8}	−46.9052	0.3×10^{-5}	−3.6098
3	772	−73.9928	-0.1×10^{-5}	−70.3827	0.4×10^{-6}	−3.6101
4	772	−97.4657	-0.3×10^{-6}	−93.8458	-0.6×10^{-6}	−3.6199

Table 5.8: Values from the Li substitution calculation.

Composition	Energy (eV)	dE (eV)
$Al_{38}Li_2$	−146.9012	2.94E-05
Al_{40}	−149.8616	2.99E-06
Difference	2.9603	-

5.2.2 Lithium substitutions

The energy needed to substitute two aluiminium atoms with lithim atoms in the way shown in figure 4.3, is found by doing a static energy calculation on a $Al_{38}Li_2$ slab and a Al_{40}. The difference in energy is the value we subtract in the interface calculations. The result from table 5.8 is E_{Li} = 2.9603 eV total for the two atoms. Notice that it is positive. The energy of dissolving two Li atom is 2.9 eV higher than dissolving two Al atoms.

5.2.3 C-phase parameters and relaxation

Table 5.9: Murnaghan and BM parameters from the partial c-phase curve fitting.

EOS	Energy (eV)	B_0 (GPa)	B_0' (-)	V $(Å^3)$	d (Å)
Murnaghan	−96.5140	236.0670	74.5486	244.8356	15.0074
Birch-Murnaghan	−96.5139	235.7802	123.3609	244.8251	15.0068

The results from the curve fitting of the data retrieved as explained in subsection 4.3.4 were plotted in figure 5.6. The parameters of the curve can be found in table 5.9, and the dataplots are found in table C.6. One point (3.5c) is off the graph and is not used in the curve fitting. It is howewer used in the interface energy calculation to subtract the correct value. The reason it is off could be that the internal structure of the cell is altered for some reason. We see in table C.6 that the calculation stopped when the energy

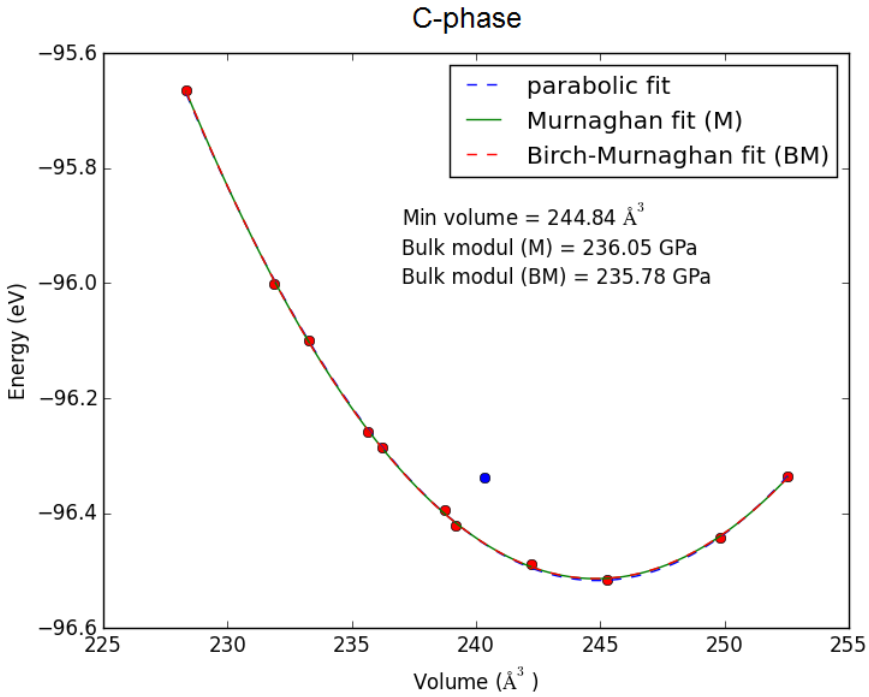


Figure 5.6: Free energy of the partial c-phase, with volume varying with extension in [001]-direction. The blue point is a bit off the graph and is not used in the curve fitting.

between the iterations were 10^{-4} eV and not 10^{-7} eV as for the others. The reason to that might be either the walltime (the calculation were aborted by the job-moderator computer), or the number of iteration exceded our maximum limit. The curve from the datafit show the strain energy in the phase as the interface thickness varies. We use the actual data points in the calculations to subtract the correct energy of the phase including strain energy. So we actually don't include the strain energy of the c-phase in the interfaceenergy, γ.

5.3 Lattice misfits

Here comes the structures after relaxation. The different interfaces are in separate tables under each subsection.

5.3.1 Theoretical unstrained misfit

The theoretical relationship between aluminium lattice constant and θ'-long-lattice constant is listed in table 5.10. It can help predict which thickness to expect for the precipitate. One see from the plot (5.7) that the misfit alternates between negative and

Table 5.10: Theoretical relationship between θ'-precipitate and al-matrix

θ' thickness	cu layers	n (al layers)	$\Delta l_{\theta'}$ (Å)	$\Delta l_{\theta'}$ (%)	$\bar{c}$ (Å)	$\Delta\bar{c}$ (%)
0.5c	1	1	1.104	27.3	8.078	37.6
1c	2	1	-1.831	-45.3	4.039	-31.2
1.5c	3	2	-0.727	-9.0	5.385	-8.3
2c	4	3	0.377	3.1	6.059	3.2
2.5c	5	4	1.481	9.2	6.463	10.1
3c	6	4	-1.454	-9.0	5.385	-8.3
3.5c	7	5	-0.350	-1.7	5.770	-1.7
4c	8	6	0.755	3.1	6.059	3.2
4.5c	9	7	1.859	6.6	6.283	7.0
5c	10	7	-1.076	-3.8	5.655	-3.7
5.5c	11	8	0.028	0.1	5.875	0.1
6c	12	9	1.132	3.1	6.059	3.2
6.5c	13	9	-1.803	-5.0	5.593	-4.7
7c	14	10	-0.699	-1.7	5.770	-1.7
7.5c	15	11	0.405	0.9	5.924	0.9
8c	16	12	1.509	3.1	6.059	3.2
8.5c	17	12	-1.425	-2.9	5.702	-2.9
9c	18	13	-0.322	-0.6	5.834	-0.6
9.5c	19	14	0.782	1.4	5.952	1.4
10c	20	15	1.887	3.1	6.059	3.2
10.5c	21	15	-1.049	-1.7	5.770	-1.7
11c	22	16	0.056	0.1	5.875	0.1

positive values. One can expect the points 3.5c, 5.5c, 7.5c and 11c to have good fit, and have a lower interface energy and better stability. One might find more precipitates of these sizes than of other sizes. For thicker precipitates we se the relative misfit is lower and that is because the precipitate have more room. The more cells the less strain each cell has to make up for.

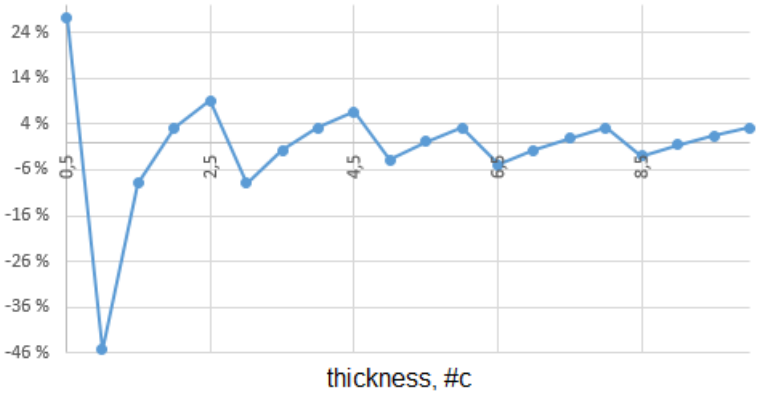


Figure 5.7: Theoretical relationship between θ'-precipitate and al-matrix. It's the relative misfit with respect to a_{Al}.

We notice the lower misfit for 3.5c opposed to 3c, and this is supporting the findings of Stobbs et al. [42] where they found that there is a tendency to form precipitates of 3.5 rather than 3 cells in thickness. It is possible to compare these theoretical values to the DFT calculated ones.

5.3.2 Pure interface (no cu)

Table 5.11: Calculated distances l in Al-Cu with *no* atoms at interface.

θ' thickness	cu	n (al layers)	$l_{\theta'}$ [Å]	$\Delta l_{\theta'}$ [Å]	$\Delta l_{\theta'}$[%]	$\bar{c}$[Å]	$\Delta\bar{c}$[%]	τ[Å]	l_s [Å]
0.5c	1	1	3,073	-0,966	-23,9	6,146	4,7	2,960	24
1c	2	1	5,657	1,618	40,0	5,657	-3,6	2,814	47
1.5c	3	2	8,633	0,555	6,9	5,755	-2,0	2,854	28
2c	4	3	11,810	-0,307	-2,6	5,905	0,6	2,975	8
2.5c	5	4	15,198	-0,958	-5,9	6,079	3,6	2,973	20
3c	6	4	17,038	0,882	5,5	5,679	-3,2	2,833	36
3.5c	7	5	20,360	0,165	0,81	5,817	-0,9	2,886	30
4c	8	6	23,745	-0,490	-2,0	5,936	1,1	2,925	16
4.5c	9	7	27,364	-0,910	-3,2	6,081	3,6	2,963	24
5c	10	7	28,659	0,385	1,4	5,732	-2,4	2,863	32
5.5c	11	8	32,250	-0,063	-0,19	5,864	-0,1	2,898	20
6c	12	9	35,941	-0,411	-1,1	5,990	2,0	2,930	14

The relaxed structures of precipitates with no copper-interface is found in table 5.11. We see the points 2c, 3.5c and 5.5c have low misfit as expected by the theoretical predictions. We expect these to have low interface energy as well. We see a good connection with the strainlegth as well. It is low for good fit and high for bad fit. It is interesting that not only is the misfit low for these thicknesses. The c-distortion is also low. That means almost no strain energy is absorbed by the lattice or the precipitate. We can expect low interface values for these points.

5.3.3 Normal interface (with cu)

The results for the case with interstitial copper is found in table 5.12. It shows that 1c, 3c, 3.5c, 4.5c, 5c and 6c has low misfit with respect to the lattice. However the thinnest one is only one cell and that one cell has a big c-distortion. That means much strain in the precipitate. Of the two 3c and 3.5c the the smallest one has the lowest c-distortion, and 4.5c have the same distortion. That leaves 5c and 6c. Of the two 5c have almost no

Table 5.12: Calculated distances l in Al-Cu with Cu at interface

θ' thickness	cu	n	l_δ[Å]	Δl_δ[Å]	Δl_δ[%]	$\bar{c}$[Å]	$\Delta\bar{c}$[%]	$l_{\theta'}$[Å]	τ_{Cu}[Å]	l_s [Å]
0.5c	3	2	9,303	1,225	15,2	5,650	-3,7	2,825	3,239	>56
1c	4	3	12,294	0,177	1,5	5,742	-2,2	5,742	3,276	30
1.5c	5	4	15,514	-0,642	-4,0	5,881	0,2	8,822	3,346	26
2c	6	5	19,067	-1,129	-5,6	6,089	3,7	12,177	3,445	21
2.5c	7	5	20,559	0,364	1,8	5,652	-3,7	14,134	3,212	36
3c	8	6	24,081	-0,154	-0,63	5,824	-0,8	17,472	3,305	8
3.5c	9	7	27,709	-0,564	-2,0	5,988	2,0	20,958	3,374	20
4c	10	7	28,969	0,695	2,5	5,640	-3,9	22,560	3,205	24
4.5c	11	8	30,711	-1,602	-4,9	5,454	-7,0	24,543	3,084	39
5c	12	9	35,916	-0,436	-1,2	5,855	-0,2	29,277	3,320	40
5.5c	13	10	40,077	-0,314	-0,8	6,062	3,3	33,343	3,367	8
6c	14	10	40,900	0,509	1,3	5,733	-2,3	34,396	3,252	52

c-distortion and the strainlength is short. 5c with copper interface corresponds to six θ'-cells.

5.3.4 Ag at interface

Table 5.13: Calculated distances l in Al-Cu-Ag with Ag (and Cu) at interface.

θ' thickness	n	l_δ[Å]	Δl_δ [Å]	Δl_δ[%]	$\bar{c}$[Å]	$\Delta\bar{c}$[%]	$l_{\theta'}$[Å]	τ_{Ag}[Å]	l_s [Å]
0.5c	2	21,072	0,877	4,3	5,574	-5,0	2,787	6,319	36
1c	3	24,374	0,140	0,57	5,694	-3,0	5,694	6,471	16
1.5c	4	27,767	-0,507	-1,8	5,787	-1,4	8,680	6,588	53
2c	5	31,070	-1,243	-3,9	5,881	0,2	11,175	6,660	65
2.5c	5	32,689	0,376	1,2	5,677	-3,3	14,192	6,399	28
3c	6	36,496	0,144	0,4	5,837	-0,6	17,513	6,551	24
3.5c	7	40,201	-0,190	-0,5	5,967	1,7	20,886	6,647	20
4c	8	43,158	-1,272	-2,9	5,937	1,1	23,746	6,691	65
4.5c	8	44,743	0,313	0,7	5,770	-1,7	25,967	6,486	56
5c	9	48,464	-0,005	0,01	5,881	0,2	29,406	6,573	13
5.5c	9	49,001	0,532	1,1	5,593	-4,7	30,766	6,212	24
6c	10	52,722	0,214	0,4	5,707	-2,8	34,242	6,393	24

The results for the silver case is shown in table 5.13. 3c, 3.5c, 4.5c, 5c and 6c all have good latticefit. But only 3c and 5c have low c-distortion. They also have low strainlength, and caan be expected to have low interface energy.

Table 5.14: Calculated distances l in Al-Cu-Li with Li at interface.

θ' thickness	n	l_δ[Å]	Δl_δ [Å]	Δl_δ[%]	$\bar{c}$[Å]	$\Delta\bar{c}$[%]	$l_{\theta'}$[Å]	τ_{Li}[Å]	l_s [Å]
0.5c	1	19,514	-0,682	-3,4	6,076	3,5	3,038	8,238	45
1c	1	21,253	1,058	5,2	5,729	-2,4	5,729	7,762	>60
1.5c	2	24,359	0,124	0,5	5,773	-1,7	8,660	7,849	24
2c	3	27,984	-0,290	-1,0	5,918	0,8	11,835	8,074	28
2.5c	4	31,861	-0,452	-1,4	6,124	4,3	15,310	8,276	28
3c	4	32,684	0,371	1,1	5,722	-2,5	17,167	7,759	34
3.5c	5	36,267	-0,085	-0,2	5,821	-0,8	20,375	7,946	22
4c	6	40,090	-0,301	-0,7	5,971	1,7	23,885	8,102	28
4.5c	6	40,680	0,289	0,7	5,636	-4,0	25,361	7,660	24
5c	7	44,401	-0,029	-0,07	5,736	-2,3	28,682	7,859	24
5.5c	8	48,325	-0,144	-0,3	5,87	0	32,318	8,003	24
6c	9	52,192	-0,316	-0,6	5,997	2,2	35,979	8,106	30

5.3.5 Li at interface

Results for the lithium interface relaxations are listed in table 5.14. Here 1.5c, 3.5c and 5.5c stands out with low misfit and distortion. This interface does not contain copper at outermost layer of θ' as in Cu- and Ag-interface-cases. 3.5c corresponds to the same number of cells as 3.5c in the pure form with no interface. It is also the predicted theoretical value and the one found by Stobbs [42]. 5.5c has the same lattice constant, c as the theoretical one and can be expected to hav every low interface energy.

5.3.6 Mg and Si at interface

Table 5.15: Calculated distances l in Al-Cu-Mg-Si with partial c-phase at interface.

θ' thickness	n	l_δ[Å]	Δl_δ[Å]	Δl_δ[%]	$\bar{c}$[Å]	$\Delta\bar{c}$[%]	$l_{\theta'}$[Å]	τ_C[Å]	l_s [Å]
0.5c	5	18,609	-1,587	-7,9	6,324	7,7	3,162	7,740	39
1c	5	20,488	0,293	1,4	5,732	-2,4	5,732	7,330	33?
1.5c	6	23,927	-0,307	-1,27	5,965	1,6	8,948	7,517	26
2c	7	27,470	-0,804	-2,8	6,035	2,8	12,069	7,657	29
2.5c	7	28,710	0,436	1,5	5,716	-2,6	14,291	7,222	33
3c	8	32,189	-0,124	-0,38	5,789	-1,4	17,359	7,366	45
3.5c	9	35,713	-0,639	-1,7	5,921	0,9	20,723	7,495	>30
4c	10	38,841	-1,550	-3,8	5,909	0,66	23,635	7,553	>35
4.5c	10	40,647	0,256	0,6	5,78	-1,5	26,020	7,317	32
5c	11	44,135	-0,300	-0,68	5,836	-0,58	29,179	7,424	26
5.5c	11	45,001	0,571	1,28	5,601	-4,6	30,805	7,107	26
6c	12	48,593	0,124	0,26	5,670	-3,4	34,019	7,240	10

Results of the Mg-Si slabs relaxations are found in table 5.15. We can find 3c, 4.5c,

5c and 6c to have low misfit, but only 5c have significally lower c-distortion and can be expected to be the lowest point on the interface energy-plot.

5.4 Interface energies

5.4.1 Energy of no interface Al-θ'

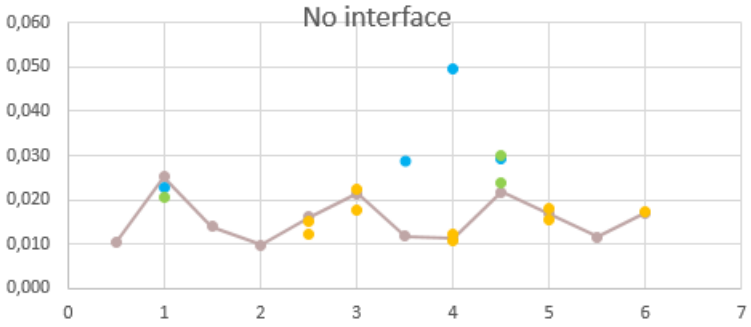


Figure 5.8: The energy of $Al \parallel \theta'$ interface with *no* interstitial Cu at the interface.

Table 5.16: Al-θ' interfac energy for different θ' thickness no cu at interface.

#Cu (at.)	#c	#a	E_{tot} (eV)	E_s (eV)	γ (eVÅ^{-2})	#Al
1	0,5	1	−610.769	0.341	0.010	20
2	1	1	−622.021	0.823	0.025	20
3	1,5	2	−424.341	0.456	0.014	13
4	2	3	−226.426	0.323	0.010	6
5	2,5	4	−537.645	0.526	0.016	16
6	3	4	−669.082	0.698	0.021	20
7	3,5	5	−531.283	0.387	0.012	15
8	4	6	−692.882	0.367	0.011	20
9	4,5	7	−524.459	0.711	0.022	14
10	5	7	−716.167	0.549	0.017	20
11	5,5	8	−578.225	0.382	0.012	15
12	6	9	−499.875	0.560	0.017	12

The interface energy is calculated in table 5.16 and ploted in figure 5.8. The points 0.5c, 2c, 3.5c, 4c and 5.5c are the lowest points. We see 3c is a top-point.

5.4.2 Energy of Al-θ' with Cu at interface

The interface energy for the Cu-interface-case is found in table 5.17 and ploted in figure 5.9. The points 1c, 1.5c, 3c, 4.5c and 5c are minima. 5c was the point expected to be

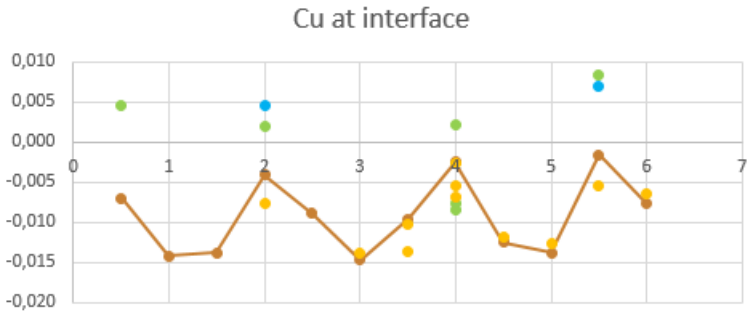


Figure 5.9: The energy of $Al \parallel \theta'$ interface *with* interstitial Cu at the interface.

Table 5.17: Al-θ' interfac energy for different θ' thickness with interstitial Cu atoms at interface.

#Cu (at.)	#c	#a	E_{tot} (eV)	E_s (eV)	γ (eVÅ^{-2})	#Al
3	0,5	2	−462.233	−0.228	−0.007	14
4	1	3	−414.263	−0.462	−0.014	12
5	1,5	4	−306.110	−0.449	−0.014	8
6	2	5	−497.340	−0.133	−0.004	14
7	2,5	5	−509.231	−0.290	−0.009	14
8	3	6	−521.154	−0.478	−0.015	14
9	3,5	7	−592.659	−0.312	−0.010	16
10	4	7	−724.037	−0.081	−0.002	20
11	4,5	8	−496.346	−0.407	−0.012	12
12	5	9	−508.123	−0.449	−0.014	12
13	5,5	10	−579.395	−0.050	−0.002	14
14	6	10	−591.331	−0.251	−0.008	14

lowest.

5.4.3 Energy of Al-θ' with Ag at interface

The interface energy for the Ag-case are found in table 5.18 and ploted in 5.10. Points 1.5c, 3c, 4.5c and 5c are the lowest ones. 3c and 5c were predicted from the misfit table. The point 5.5c might be explained by that it was held to the wrong n, that is number of aluminium cells across the precipitate thickness. It might give a high value if it contains much strain. In the misfit table we see it has a lot of c-distortion.

5.4.4 Energy of Al-θ' with Li at interface

The lithium interface-case is in table 5.19 and ploted in 5.11. The points 1.5c, 2c, 3.5c, 5c and 5.5c are minima. We predicted 5.5 to be the lowest point. And the point 3.5 keeps coming up as one of the most stable points for all interfaces.

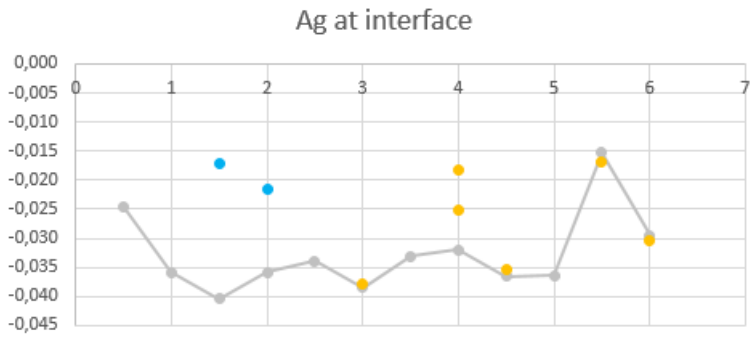


Figure 5.10: The energy of $Al \parallel \theta'$ interface in Al-Cu-Ag.

Table 5.18: Al-θ' interfac energy for different θ' thickness in Al-Cu-Ag.

#Cu (at.)	#c	#a	E_{tot} (eV)	E_s (eV)	γ (eVÅ^{-2})	#Al
3	0,5	2	−485.370	−0.806	−0.025	14
4	1	3	−497.467	−1.170	−0.036	14
5	1,5	4	−599.256	−1.318	−0.040	17
6	2	5	−610.840	−1.168	−0.036	17
7	2,5	5	−412.733	−1.109	−0.034	10
8	3	6	−544.490	−1.257	−0.039	14
9	3,5	7	−376.236	−1.082	−0.033	8
10	4	8	−657.653	−1.045	−0.032	17
11	4,5	8	−639.566	−1.192	−0.037	16
12	5	9	−411.545	−1.188	−0.036	8
13	5,5	9	−542.465	−0.499	−0.015	12
14	6	10	−674.537	−0.961	−0.029	16

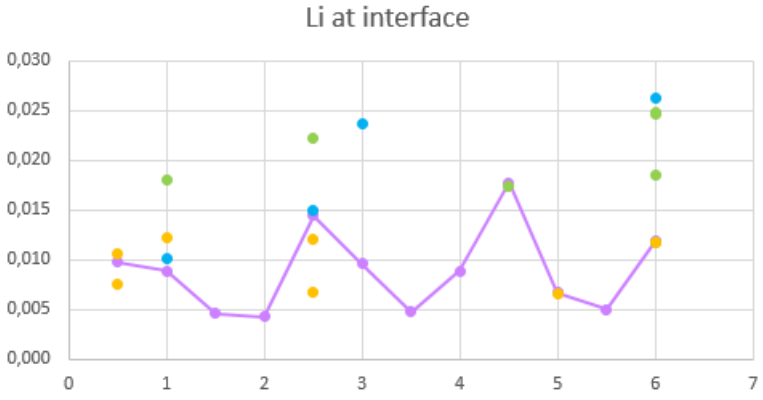


Figure 5.11: The energy of $Al \parallel \theta'$ interface in Al-Cu-Li.

Table 5.19: Al-θ' interfac energy for different θ' thickness in Al-Cu-Li.

#Cu (at.)	#c	#a	E_{tot} (eV)	E_s (eV)	γ (eVÅ^{-2})	#Al
1	1	0,5	−634.840	0.318	0.010	21
2	1	1	−466.791	0.289	0.009	15
3	2	1,5	−298.850	0.150	0.005	9
4	3	2	−310.596	0.139	0.004	9
5	4	2,5	−681.624	0.470	0.014	21
6	4	3	−333.891	0.312	0.010	9
7	5	3,5	−345.782	0.154	0.005	9
8	6	4	−357.381	0.290	0.009	9
9	6	4,5	−668.512	0.580	0.018	19
10	7	5	−620.671	0.218	0.007	17
11	8	5,5	−392.711	0.162	0.005	9
12	9	6	−404.222	0.385	0.012	9

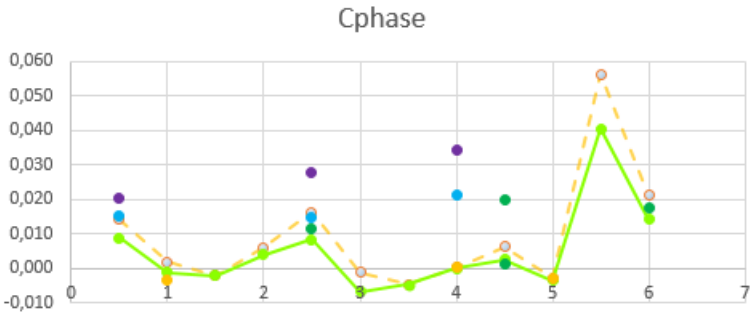


Figure 5.12: The energy of $Al \parallel \theta'$ interface in Al-Cu-Mg-Si.

5.4.5 Energy of Al-θ' with Mg and Si at interface

Results from the Mg-Si (partial c-phase) case are given in table 5.20 and plotted in figure 5.12. The datapoint marked in blue, did only converge to within 10^{-2} eV but is still representable for the 2c thickness. The points 1.5c, 3c and 5c are the lowest ones, or the local minima at least. Points 3c and 5c were predicted to be low points.

5.4.6 Comparing the interfacial energies

The interface energy for all the different configurations for the five cases are plotted together in figure 5.13. We see that the no-copper (no interface) configuration is the least beneficial of them all. Then comes the one with copper proving what Bourgeois found [4]. The most stable structure is in fact the on with silver on the outside of the copper enriched θ'-phase. The silver significantly lowers the misfit. The points 5.5c is not representative for the Ag and Mg-Si cases.

Table 5.20: Al-θ' interfac energy for different θ' thickness in Al-Cu-Mg-Si.

#Cu (at.)	#c	#a	E_{tot} (eV)	E_s (eV)	γ (eVÅ^{-2})	#Al	$E_{c-phase}$ (eV)
1	0,5	5	−958.643	0.287	0.009	14	−96.335
2	1	5	−982.518	−0.034	−0.001	14	−96.421
3	1,5	6	−1006.111	−0.064	−0.002	14	−96.517
4	2	7	−1029.315	0.127	0.004	14	−96.443
5	2,5	7	−1112.386	0.278	0.009	15	−96.259
6	3	8	−1076.489	−0.215	−0.007	14	−96.339
7	3,5	9	−740.442	−0.151	−0.005	8	−96.514
8	4	10	−823.685	0.004	0.000	9	−96.506
9	4,5	10	−1146.647	0.086	0.003	14	−96.395
10	5	11	−990.598	−0.115	−0.004	11	−96.488
11	5,5	11	−1012.141	1.323	0.041	11	−96.001
12	6	12	−976.811	0.467	0.014	10	−96.285

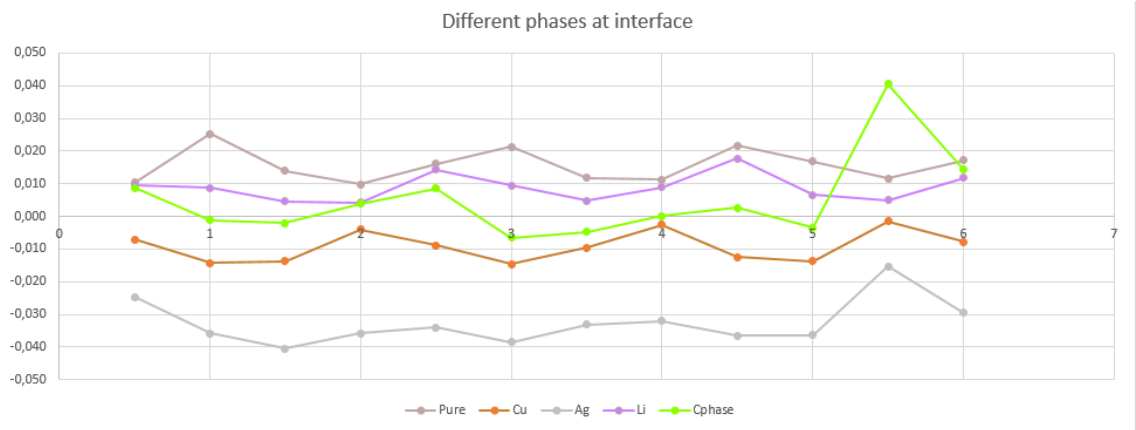


Figure 5.13: Interface energy [meV/Å^2] in the four cases.

5.5 Parameter relations

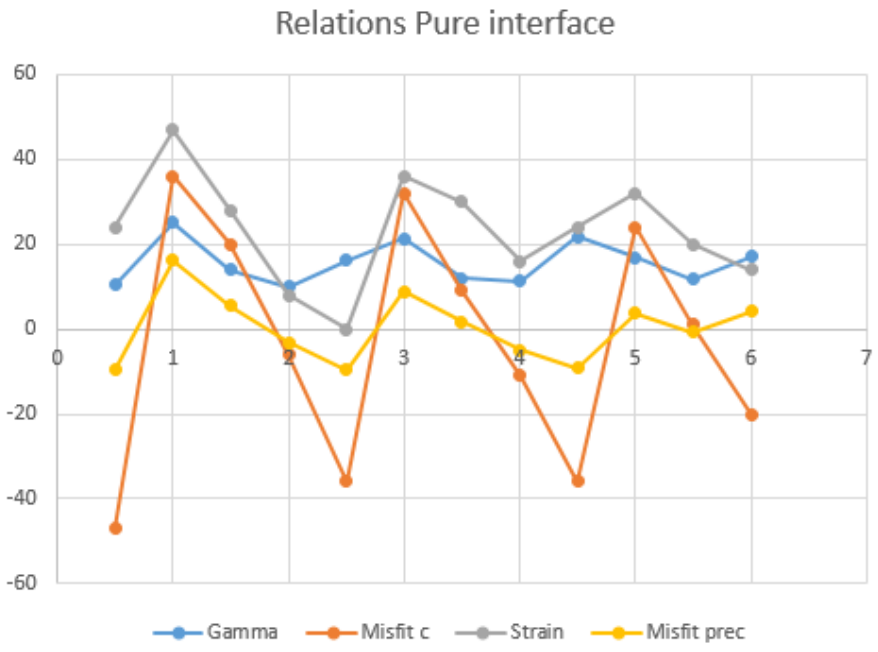


Figure 5.14: Comparing.

In figure 5.14 we see that some of the parameters follow each other. If there is big misfit, there is usually a big c-distortion (misfit in c), that gives a long strainlength that results in high interfacial energy.

5.5.1 Strain energy considerations

We observe that the bulk modulus (the double derivative of energy vs volume) that dictates the amont of enery absorbed in the distortion of bondings from their equilibrium length, is much larger when we constrain the phases to one direction of freedom. Silver has a bulkmodulus of $B_0^{silver} = 101.6\,\text{GPa}$ when separate, but it increases to $B_{0,constrained}^{silver} = 121.2\,\text{GPa}$ when held to the aluminium lattice constant in two directions. The same with θ', which we expect to have a bulk modulus intermediate between B_0^{Al} and B_0^{Cu}, $B_0^{\theta'} \in (76.9, 137)$. But when constrained to aluminium we get $B_{0,constrained}^{\theta'} = 216.5\,\text{GPa}$. This means that the θ' phase is very "hard" compared to the aluminium. Hard materials absorb more energy per volume than soft materials. It means that the precipitate will resist the forces that come from the matrix misfit at the expence of the aluminium. If the misfit is large, then there will be a longer length of distortion in the aluminium around, and vice versa.

We also notice that the aluminium is harder when expanded than when compressed. In figure 5.15 we have ploted the free energy of an aluminium cell held constant in two directions and relaxed in the third. This means that a precipitate that is too large for the aluminium lattice will give a higher interface energy than a corresponing too small precipitate with same absolute misfit. We see that this holds for the 1.5c vs 6c/4c case in the no-cu-case, 5c vs 2c in the same case. We also see this for the 6c vs 1.5c/3.5c case with copper-interface. Also comparing 5c/1.5c and 1c/4.5c for the cu-case.

5.6 Conclusions

We have found that the different misfits and strains are connected to the stability of the precipitate, in the sense that a low misfit gives a short strainlegth, which in turn lowers the strain energy part of the interface energy. The more low-energy thicknesses are stable for thin precipitates the more there are of them. This should also be found experimentally for the cases studied here to confirm it. As is done by Stobbs et al. We

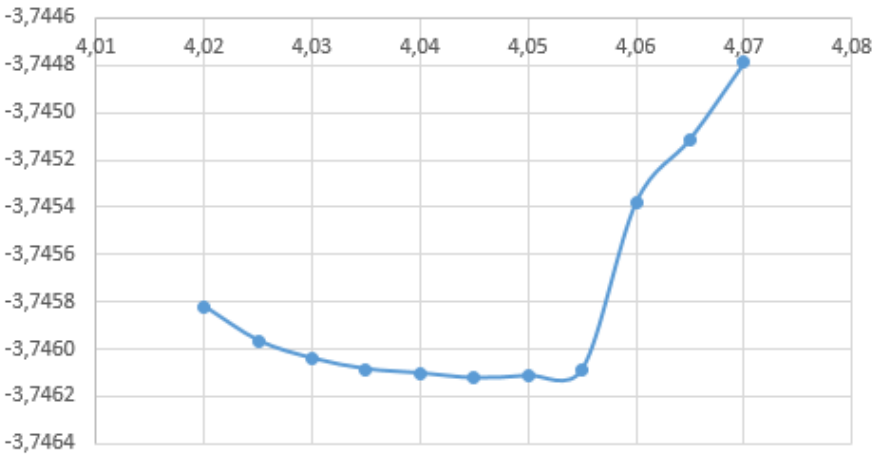


Figure 5.15: Plot of free energy of aluminium cell held constant in two directions and relaxed in the third. The curve is steeper when expanding the cell than when compressing.

also found the θ'-precipitate in the Al-Cu-Ag alloy to have the lowest chemical interface energy. And we found the Litium case to be not very beneficial.

A

Appendix A

Acronyms

DFT Density Functional Theory

VASP Vienna Abinitio Simulation Package

ASE Atomic Simulation Evironment

LDA Local Density Approximation

XC-energy Exchange-correlation energy

GGA Generalized Gradient Approximation

EOS Equation Of State

M Murnaghan EOS

BM Birch-Murnaghan EOS

QCC Quantum Chemistry Computing

PAW Projector Augmented Wave -method

LAPW Linear Augmented Plane Wave -method

Appendix B

The VASP input files (to be written)

INCAR for ionic relaxation

```
INCAR for ionic relaxation,

! Electronic relaxation
ALGO   = Fast       ! Algorithm for electronic relaxation
NELMIN = 4          ! Minimum # of electronic steps
EDIFF  = 1E-5       ! Accuracy for electronic groundstate
ENCUT  = 350        ! Cut-off energy for plane wave expansion
PREC   = High       ! Normal/Accurate/High
LREAL  = Auto       ! Projection in reciprocal space?
ISMEAR = 1          ! Smearing of partial occupancies. Metals: 1; else < 1.
SIGMA  = 0.2        ! Smearing width
ISPIN  = 1          ! Spin polarization?

! Ionic relaxation
EDIFFG = +0.0001    ! Tolerance for ions
NSW    = 200        ! Max # of ionic steps
MAXMIX = 80         ! Keep dielectric function between ionic movements
IBRION = 1          ! Algorithm for ions. 0: MD, 1: QN/DIIS, 2: CG, 6: elastic
ISIF   = 2          ! 7: vol relax 2: ions 3: ions+cell
! ADDGRID= .TRUE.   ! More accurate forces with PAW

! Output options
NWRITE = 1          ! Write electronic convergence at first step only
```

```
! Memory handling
NPAR    = 4
LPLANE  = .TRUE.
LSCALU  = .FALSE
NSIM    = 4
```

./INCAR1.fil

INCAR for energy calculation

```
INCAR for high-accuracy total energy calculation.

! Electronic relaxation
ALGO   = Fast       ! Algorithm for electronic relaxation
NELMIN = 4          ! Minimum # of electronic steps
EDIFF  = 1E-7       ! Accuracy for electronic groundstate
ENCUT  = 350        ! Cut-off energy for plane wave expansion
PREC   = High       ! Normal/Accurate/High
LREAL  = Auto       ! Projection in reciprocal space?
ISMEAR = -5         ! Smearing of partial occupancies. k-points >2: -5; else 0
SIGMA  = 0.2        ! Smearing width
ISPIN  = 1          ! Spin polarization?

! Ionic relaxation
NSW    = 0          ! Final high-accuracy calculation without relaxation

! Output options
LWAVE  = .FALSE.    ! Write WAVECAR?
LCHARG = .FALSE.    ! Write CHGCAR?
!NEDOS  = 2101       ! Number of grid points for DOS
!EMIN   = -9         ! Minimum energy for evaluation of DOS
!EMAX   =  12        ! Maximum energy for evaluation of DOS
! RWIGS  = 1.0 1.0   !  Wigner-Seitz radii; for LDOS
!LELF   = .TRUE.     !  Electron localization function
!LORBIT = 10         ! Write LDOS to DOSCAR + PROCAR

! Memory handling
NPAR    = 4
```

```
29 LPLANE   = .TRUE.
   LSCALU   = .FALSE
31 NSIM     = 4
```

./INCAR2.fil

B.1 KPOINTS file

```
1 Max k-point distance: 0.250000
  0
3 Gamma
   7 7 1
5  0   0   0
```

./KPOINTS.fil

Appendix C

Raw data tables

Table C.1: Computed free energy for varying lattice parameter of aluminium, with large interval.

a (Å)	E (eV)	dE (eV)	E (eV/at.)	V $(Å^3)$
3,7	−13.5675	-5.0×10^{-7}	−3.3919	50.6530
3,8	−14.3368	-1.2×10^{-6}	−3.5842	54.8720
3,9	−14.7806	-6.2×10^{-7}	−3.6952	59.3190
4,0	−14.9704	2.8×10^{-8}	−3.7426	64.0000
4,1	−14.9536	4.5×10^{-9}	−3.7384	68.9210
4,2	−14.7858	-5.5×10^{-7}	−3.6965	74.0880
4,3	−14.5108	-9.6×10^{-8}	−3.6277	79.5070
4,4	−14.1574	1.5×10^{-6}	−3.5393	85.1840

Table C.2: Computed free energy for varying lattice parameter for aluminium.

a (Å)	E (eV)	dE (eV)	E (eV/at.)	V $(Å^3)$
4,02	−14.9808	2.8×10^{-7}	−3.7452	64.9648
4,025	−14.9825	3.2×10^{-7}	−3.7456	65.2075
4,03	−14.9836	-1.3×10^{-5}	−3.7459	65.4508
4,035	−14.9843	3.9×10^{-7}	−3.7461	65.6947
4,04	−14.9844	4.3×10^{-7}	−3.7461	65.9393
4,045	−14.9842	4.5×10^{-7}	−3.7461	66.1844
4,05	−14.9836	4.9×10^{-7}	−3.7459	66.4301
4,055	−14.9824	5.2×10^{-7}	−3.7456	66.6765
4,06	−14.9809	5.4×10^{-7}	−3.7452	66.9234

Table C.3: Computed free energy of θ' for varying c, and keeping $b = a_{Al}$

c (Å)	E (eV)	dE (eV)	V $(Å^3)$
5,84	−23.4662	-1,80E-07	95.2757
5,85	−23.4672	-3,55E-08	95.4388
5,86	−23.4679	3,32E-05	95.6020
5,87	−23.4681	-1,76E-10	95.7651
5,88	−23.4680	5,33E-08	95.9283
5,89	−23.4674	-8,97E-10	96.0914
5,90	−23.4666	-8,99E-10	96.2545
5,91	−23.4655	1,46E-10	96.4177
5,92	−23.4642	1,76E-10	96.5808

Table C.4: Computed free energy of free bulk silver with fcc structure.

a (Å)	E (eV)	dE (eV)	V $(Å^3)$
4,135	−11.3048	4.9×10^{-5}	70.7012
4,140	−11.3075	-7.6×10^{-4}	70.9589
4,145	−11.3094	-3.2×10^{-6}	71.2153
4,150	−11.3106	-3.2×10^{-6}	71.4734
4,155	−11.3115	5.0×10^{-5}	71.7320
4,160	−11.3127	1.2×10^{-7}	71.9913
4,165	−11.3158	?	72.2512
4,170	−11.3012	7.1×10^{-7}	72.5127
4,175	−11.3208	?	72.7739
4,180	−11.3070	-2.4×10^{-7}	73.0356
4,185	−11.3049	-7.3×10^{-8}	73.2970
4,190	−11.3031	-2.9×10^{-6}	73.5601

Table C.5: Computed free energy of silver constrained to the matrix.

a (Å)	E (eV)	dE (eV)	V $(Å^3)$
4,0891	−11.1770	-4.0×10^{-6}	66.7109
4,1141	−11.1974	6.3×10^{-8}	67.1188
4,1391	−11.2155	8.2×10^{-8}	67.5266
4,1641	−11.2315	1.1×10^{-7}	67.9345
4,1891	−11.2450	1.2×10^{-7}	68.3424
4,2141	−11.2561	1.7×10^{-7}	68.7502
4,2391	−11.2655	1.9×10^{-7}	69.1581
4,2641	−11.2725	2.3×10^{-7}	69.5659
4,2891	−11.2691	-7.6×10^{-4}	69.9738
4,3141	−11.2778	-3.2×10^{-7}	70.3816
4,3391	−11.2788	-3.3×10^{-7}	70.7895
4,3641	−11.2779	-3.1×10^{-7}	71.1974
4,3891	−11.2754	-8.9×10^{-6}	71.6052
4,4141	−11.2710	-1.8×10^{-7}	72.0131
4,4391	−11.2651	-6.7×10^{-6}	72.4209
4,4641	−11.2580	-5.4×10^{-8}	72.8288
4,4891	−11.2490	-3.7×10^{-9}	73.2367

Table C.6: Computed free energy of the partial c-phase found at the interface of Al-Cu-Mg-Si the alloy.

d (Å)	E (eV)	dE (eV)	V $(Å^3)$
7,24	−96.2852	8.8×10^{-7}	236.2315
7,107	−96.0011	7.3×10^{-7}	231.8919
7,424	−96.4881	2.6×10^{-6}	242.2352
7,317	−96.3946	2.7×10^{-6}	238.7439
7,149	−96.0999	-7.8×10^{-7}	233.2623
6,998	−95.6646	-8.8×10^{-7}	228.3353
7,366	−96.3388	-6.7×10^{-7}	240.3427
7,222	−96.2589	-4.1×10^{-8}	235.6442
7,657	−96.4433	3.6×10^{-7}	249.8376
7,517	−96.5166	9.3×10^{-7}	245.2696
7,33	−96.4212	-4.8×10^{-7}	239.1681
7,74	−96.3352	-2.6×10^{-7}	252.5458

Appendix D

The python/ASE script

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

##############################################
### Script for building supercells of AL-Cu  ###
### with interfaces: Cu, Ag, C-phase (Mg,Si) ###
### and Li.                                  ###
###              Written by Philip Oestli    ###
### ======================================== ###

from __future__ import print_function
from __future__ import division

import numpy as np
import ase
import ase.io
from ase.utils import geometry
from ase.visualize import view
from ase.lattice.spacegroup import crystal
#np.set_printoptions(suppress=True)

def primitive_from_conventional_cell(atoms,spacegroup=1,setting=1):
    """Returns primitive cell given an Atoms object for a
    conventional cell and it's spacegroup."""
    from ase.lattice.spacegroup import Spacegroup
    from ase.utils.geometry  import cut
    sg = Spacegroup(spacegroup, setting)
```

```
    prim_cell = sg.scaled_primitive_cell
    return cut(atoms,a=prim_cell[0],b=prim_cell[1],c=prim_cell[2])

#Aluminium FCC unit cell
a = 4.0391

Al = crystal(symbols=['Al'], basis=(0.0, 0.0, 0.0), spacegroup=225,
             cellpar=a)
Al.write('Al.POSCAR')
#view(Al)

#Primitive aluminium cell:
alprim = primitive_from_conventional_cell(Al, 225)
alprim.write('Alprimitive.POSCAR')
#view(alprim)

#Copper unit cell
a_cu = 3.632
Cu = crystal(symbols=['Cu'],basis=(0.0, 0.0, 0.0),
                                  spacegroup=225,cellpar=a_cu)
Cu.write('Cu.POSCAR')
#view(Cu)

#Silver FCC unit cell
a_Ag = 4.145
Ag=crystal(symbols=['Ag'],basis=(0.0,0.0,0.0),
                                    spacegroup=225,cellpar=a_Ag)
Ag.write('Ag.POSCAR')
#view(Ag)

#Slab for Li substitusion:
longAl = Al.repeat((1,1,10))
longAl.numbers[0] = ase.data.atomic_numbers['Li']
longAl.numbers[4] = ase.data.atomic_numbers['Li']
longAl.write('Al38Li2.POSCAR')
#view(longAl)

# thetaprime unit cell
c = 5.87
```

```
thetap = crystal(symbols=['Al', 'Al', 'Cu'],
                 basis=[(0.0, 0.0, 0.0),
                        (0.0, 0.0, 0.5),
                        (0.0, 0.5, 0.25)],
                 spacegroup='I -4 m2 ',
                 cellpar=[a, a, c, 90.0, 90.0, 90.0])
thetap.write('thetap.POSCAR')
#view(thetap)

#Long theta' slab with one interstitial Cu atom
thetapcu = thetap.repeat((1, 1, 4))
Cu1 = ase.Atom('Cu', [0.5*a, 0, 0.25*c])
thetapcu = thetapcu + Cu1
thetapcu = geometry.sort(thetapcu)
thetapcu.write('thetapcu.POSCAR')
#view(thetapcu)

# C-phase (complete)
CphaseC = crystal(symbols='Si,Si,Si,Mg,Mg,Mg,Cu,Al,Mg'.split(','),
        basis=[ (0.5,   0.5, 0.25), # Si
                (0.83,  0.5, 0.08), # Si
                (0.83,  0.5, 0.58), # Si
                (0.61, 0.0, 0.03),  # Mg
                (0.61, 0.0, 0.53),  # Mg
                (0.94, 0.0, 0.36),  # Mg
                (0.68, 0.5, 0.31),  # Cu
                (0.68, 0.5, 0.81),  # Al
                (0.94, 0.0, 0.86)], # Mg
        spacegroup=4,
        cellpar=[a*0.5*np.sqrt(26),a, a*2,90.0,100.9,90.0])
CphaseC.write('CphaseC.POSCAR')
#view(CphaseC)

#Cphase (complete) after relaxation
Cphase = ase.io.read('Cphase.CONTCAR')
#view(Cphase)


##########################################
```

```
### Al/thetaprime interface structures ###
### ================================== ###

################## al - theta' ############################

#### parameters ####
alrows = 10          #number of aluminium cells outside interface
ntheta = 3           #integer number of theta's
nthetahalf = 0       #true if half-numbered theta's 0/1
n = 5                #precipitate cross section in number of a's
vw = 0               # 1 - view     2 - write     3 - view CONTCAR
####################

dc = n*a/(ntheta + nthetahalf/2)

thetap = crystal(symbols=['Al', 'Al', 'Cu'],
                 basis=[(0.0, 0.0, 0.0),
                        (0.0, 0.0, 0.5),
                        (0.0, 0.5, 0.25)],
                        spacegroup='I -4 m2 ',
                 cellpar=[a, a, dc, 90.0, 90.0, 90.0])

sthetap = crystal(symbols=['Al', 'Al', 'Cu'],
                 basis=[(0.0, 0.0, 0.0),
                        (0.0, 0.0, 0.5),
                        (0.0, 0.5, 0.25)],
                        spacegroup='I -4 m2 ',
                 cellpar=[a, a, dc, 90.0, 90.0, 90.0])
ind2remove = range(1,3,1)
ind2remove.append(5)
L = dc/2
sthetap.set_cell([a,a,L],scale_atoms=False)
del sthetap[ind2remove]

testslab = Al.repeat((1, 1, alrows))

if (ntheta>0):
    testslab = geometry.stack(
        testslab,
```

```
145         thetap.repeat((1, 1, ntheta)),
            axis=2)
147
if (nthetahalf):
149     testslab = geometry.stack(testslab,sthetap,axis=2)

151 testslab = geometry.stack(testslab,Al.repeat((1,1,alrows)),axis=2)
    testslab = geometry.sort(testslab)
153
if (vw == 1):
155     view(testslab)
elif (vw == 2):
157     testslab.write('POSCAR')
elif (vw == 3):
159     rfile = ase.io.read('CONTCAR')
        view(rfile)
161
############################## END al-theta'  ######################
163
################  slab 1 ### Al - Cu - Theta'  ####################
165
#### parameters ####
167 alrows = 20           #number of aluminium rows outside interface
    ntheta = 4            #integer number of theta's
169 nthetahalf = 1         #true if half-numbered theta's
    n = 8                 #precipitate cross section in number of a's
171 vw = 0                # 1 - view     2 - write     3 - view CONTCAR
    ####################
173
dc = n*a/(1+ntheta + nthetahalf/2)
175
thetap = crystal(symbols=['Al', 'Al', 'Cu'],
177                  basis=[(0.0, 0.0, 0.0),
                            (0.0, 0.0, 0.5),
179                         (0.0, 0.5, 0.25)],
                            spacegroup='I -4 m2 ',
181                 cellpar=[a, a, dc, 90.0, 90.0, 90.0])

183 sthetap = crystal(symbols=['Al', 'Al', 'Cu'],
```

```
                    basis=[(0.0, 0.0, 0.0),
185                        (0.0, 0.0, 0.5),
                           (0.0, 0.5, 0.25)],
187                        spacegroup='I -4 m2 ',
                    cellpar=[a, a, dc, 90.0, 90.0, 90.0])
189 ind2remove = range(1,3,1)
    ind2remove.append(5)
191 L = dc/2
    sthetap.set_cell([a,a,L],scale_atoms=False)
193 del sthetap[ind2remove]


195

    slab1 = Al.repeat((1, 1, alrows))
197

    slab1 = geometry.stack(
199     slab1,
        thetap.repeat((1, 1, ntheta+1)),
201     axis=2)

203 if (nthetahalf):
        slab1 = geometry.stack(slab1,sthetap,axis=2)
205


207 slab1 = geometry.stack(slab1,Al.repeat((1, 1, alrows)),axis=2)
    slab1 = geometry.sort(slab1)
209

    num = len(slab1)
211

    # Insert additional Cu atoms at interface
213 Cu1 = ase.Atom('Cu',
              [0.5*a,0.0,slab1.positions[num-2-2*ntheta-nthetahalf,2]])
215 if nthetahalf:
        Cu2 = ase.Atom('Cu', [0.5*a, 0.0 , slab1.positions[num-1, 2]])
217 else:
        Cu2 = ase.Atom('Cu', [0.0, 0.5*a, slab1.positions[num-1, 2]])
219 slab1 = slab1 + Cu1 + Cu2
    slab1 = geometry.sort(slab1)
221

    if (vw == 1):
```

```
    view(slab1)
elif (vw == 2):
    slab1.write('POSCAR')
#Al.repeat((1, 1, 12)).write('POSCAR', vasp5=True)
elif (vw == 3):
    rfile = ase.io.read('CONTCAR')
    view(rfile)

#################### END Al - Cu - Theta' ######################

################       slab 2  ###  Al - Ag - Theta' ##############

#### parameters ####
alrows = 12          #number of aluminium rows outside interface
ntheta = 5           #integer number of theta's
nthetahalf =  0      #true if half-numbered theta's
n = 9                #precipitate thickness in # of a's with Cu X2
vw = 0               # 1 - view     2 - write     3 - view CONTCAR
####################

ncs = (ntheta + nthetahalf/2)
dc = n*a/(ncs+1)
precithick = n*a
border1 = (alrows+1)*a
border2 = border1+precithick

thetap = crystal(symbols=['Al', 'Al', 'Cu'],
                 basis=[(0.0, 0.0, 0.0),
                        (0.0, 0.0, 0.5),
                        (0.0, 0.5, 0.25)],
                        spacegroup='I -4 m2 ',
                 cellpar=[a, a, dc, 90.0, 90.0, 90.0])

sthetap = crystal(symbols=['Al', 'Al', 'Cu'],
                 basis=[(0.0, 0.0, 0.0),
                        (0.0, 0.0, 0.5),
                        (0.0, 0.5, 0.25)],
                        spacegroup='I -4 m2 ',
                 cellpar=[a, a, dc, 90.0, 90.0, 90.0])
```

```
ind2remove = range(1,3,1)
ind2remove.append(5)
L = dc/2
sthetap.set_cell([a,a,L],scale_atoms=False)
del sthetap[ind2remove]

slab2 = Al.repeat((1, 1, alrows+1))

if (1):#ntheta>0):
    slab2 = geometry.stack(
        slab2,
        thetap.repeat((1, 1, ntheta+1)),axis=2)

if (nthetahalf):
    slab2 = geometry.stack(slab2,sthetap,axis=2)

slab2 = geometry.stack(slab2,Al.repeat((1, 1, alrows+1)),axis=2)
slab2 = geometry.sort(slab2)

# Replace some Al atoms with Ag
z = slab2.positions[:, 2]

mask = (z>border1-a-1)*(z < border1-1)
slab2.numbers[mask] = ase.data.atomic_numbers['Ag']

mask = (z > border2+1)*(z<border2+a+1)
slab2.numbers[mask] = ase.data.atomic_numbers['Ag']

slab2 = geometry.sort(slab2)

# Insert additional Cu atoms at interface
num = len(slab2)
Cu1 = ase.Atom('Cu',
        [0.5*a,0.0,slab2.positions[num-2-2*ntheta-nthetahalf,2]])
if nthetahalf:
    Cu2 = ase.Atom('Cu', [0.5*a, 0.0 , slab2.positions[num-1, 2]])
else:
    Cu2 = ase.Atom('Cu', [0.0, 0.5*a, slab2.positions[num-1, 2]])
slab2 = slab2 + Cu1 + Cu2
```

```
301
slab2 = geometry.sort(slab2)
303 if (vw == 1):
    view(slab2)
305 elif (vw == 2):
    slab2.write('POSCAR')
307 elif (vw == 3):
    rfile = ase.io.read('CONTCAR')
309 view(rfile)

311 ########################  END  Ag Ag Ag  ########################

313 ################      slab 3  ###  Al - Cphase - Theta' ###########

315 #### parameters ####
alrows = 9          #number of aluminium rows outside interface
317 ntheta = 5          #integer number of theta's
nthetahalf = 1      #true if half-numbered theta's
319 n = 12              #precipitate thickness in # of a's icl Clayer X2
vw = 0              # 1 - view     2 - write     3 - view CONTCAR
321 ####################

323 T_c = 5.7                   # Chase layer thickness
T_c2 = 7.72
325 N = (8*2+n)
L = N*a
327
# Make a partial cphase layer to ad to interface
329
# Remove the top/bottom Mg and Si atoms from the C-phase
331 Clayer = Cphase.copy()
scaled = Cphase.get_scaled_positions()
333 x = scaled[:, 0]
mask = (x < 0.2) + (x > 1.0 - 0.2)
335 del Clayer[mask]

337 # Rotate the structure such that the c_C // a_Al and b_C // b_Al
geometry.rotate(Clayer, Clayer.cell[2],
339                             [1, 0, 0], Clayer.cell[1], [0, 1, 0])
```

```
# Replace the cell with an orthorombic one fitting the Al lattice
Clayer.set_cell([[2 * a, 0, 0],
                 [0, a, 0],
                 [0, 0, T_c]])
Clayer.positions -= Clayer.positions[6] - (1, 1, 0.4)
Clayer.set_scaled_positions(Clayer.get_scaled_positions())
Clayer.positions -= (1, 1, 0.4)
Clayer.positions[0,0] = 0
Clayer.positions[0,1] = 0
Clayer.positions[9,0] = 0
Clayer.positions[3,1] = +0
Clayer.write('Clayer.POSCAR')
#view(Clayer)

# Make a second Cprime layer and adjust it in x- and y-direction
Clayer2 = Clayer.copy()
if (nthetahalf == 1):
    Clayer2.positions += [0, a/2 , 0]
else:
    Clayer2.positions += [a/2 , 0 , 0]
Clayer2.set_scaled_positions(Clayer2.get_scaled_positions())

# Make theta
thetathick = a*n - 2*T_c2
dc = thetathick/(ntheta + nthetahalf/2)


thetap = crystal(symbols=['Al', 'Al', 'Cu'],
                 basis=[(0.0, 0.0, 0.0),
                        (0.0, 0.0, 0.5),
                        (0.0, 0.5, 0.25)],
                        spacegroup='I -4 m2 ',
                 cellpar=[a, a, dc, 90.0, 90.0, 90.0])

sthetap = crystal(symbols=['Al', 'Al', 'Cu'],
                 basis=[(0.0, 0.0, 0.0),
                        (0.0, 0.0, 0.5),
                        (0.0, 0.5, 0.25)],
```

```
                          spacegroup='I -4 m2 ',
                    cellpar=[a, a, dc, 90.0, 90.0, 90.0])
ind2remove = range(1,3,1)
ind2remove.append(5)
L = dc/2
sthetap.set_cell([a,a,L],scale_atoms=False)
del sthetap[ind2remove]

#stack theta
if (ntheta>0):
    at2 = thetap.copy()
    at2 = at2.repeat((1, 1, ntheta))
    if (nthetahalf):
        at2 = geometry.stack(at2,sthetap,axis=2)
else:
    at2 = sthetap.copy()

#fit theta with Clayer
at2 = at2.repeat((2,1,1))
at2.positions += [0.5 * a, 0.5 * a, 0.0]
at2.set_scaled_positions(at2.get_scaled_positions())

#add al layer at end
at2 = geometry.stack(at2,
                  geometry.cut(Al,a=(2,0,0),b=(0,1,0),c=(0,0,0.5)))

      ## build ##

# Stack Al matrix
slab3 = Al.repeat((2, 1, alrows))

# Stack midslab containing theta and 2 clayers
midslab = geometry.stack(Clayer,at2)
midslab = geometry.stack(midslab,Clayer2)

slab3 = geometry.stack(slab3, midslab)

#add half al layer to fit
allayer = geometry.cut(Al, a=(2, 0, 0), b=(0, 1, 0), c=(0, 0, 0.5))
```

```
allayer.positions += [a/2,0,0]
allayer.set_scaled_positions(allayer.get_scaled_positions())
slab3 = geometry.stack(slab3, allayer)

#append aluminium at end
slab3 = geometry.stack(slab3, Al.repeat((2, 1, alrows)))

slab3 = geometry.sort(slab3)
if (vw == 1):
    view(slab3)
elif (vw == 2):
    slab3.write('POSCAR')
elif (vw == 3):
    rfile = ase.io.read('CONTCAR')
    view(rfile)

######################  END  C C C C C phase  ####################

###################  slab 4 ### Al - Li - Theta'  #################

#### parameters ####
alrows = 20          #number of aluminium rows ouside interface
ntheta = 0           #integer number of theta's
nthetahalf = 1       #true if half-numbered theta's
n = 1                #theta thickness in # of a's
vw = 3               # 1 - view   2 - write  3 - view CONTCAR
####################

dc = n*a/(ntheta + nthetahalf/2)
precithick = a*n
border1 = (alrows+1)*a
border2 = border1+precithick

thetap = crystal(symbols=['Al', 'Al', 'Cu'],
                 basis=[(0.0, 0.0, 0.0),
                        (0.0, 0.0, 0.5),
                        (0.0, 0.5, 0.25)],
                        spacegroup='I -4 m2 ',
                 cellpar=[a, a, dc, 90.0, 90.0, 90.0])
```

```
457
sthetap = crystal(symbols=['Al', 'Al', 'Cu'],
459                   basis=[(0.0, 0.0, 0.0),
                             (0.0, 0.0, 0.5),
461                          (0.0, 0.5, 0.25)],
                             spacegroup='I -4 m2 ',
463                   cellpar=[a, a, dc, 90.0, 90.0, 90.0])
ind2remove = range(1,3,1)
465 ind2remove.append(5)
L = dc/2
467 sthetap.set_cell([a,a,L],scale_atoms=False)
del sthetap[ind2remove]
469
slab4 = Al.repeat((1, 1, alrows+1))
471
if (ntheta>0):
473     slab4 = geometry.stack(
            slab4,
475         thetap.repeat((1, 1, ntheta)),axis=2)

477 if (nthetahalf):
        slab4 = geometry.stack(slab4,sthetap,axis=2)
479 slab4 = geometry.stack(slab4,Al.repeat((1, 1, alrows+1)),axis=2)
slab4 = geometry.sort(slab4)
481
num = len(slab4)
483
# Replace some Al atoms with Li
485 x = slab4.positions[:, 0]
z = slab4.positions[:, 2]
487 mask = (x<1)*( (z>border1-2*a+1) *
                  (z<border1-a-1) + (z>border1-a+1)*(z<border1-1))
489 slab4.numbers[mask] = ase.data.atomic_numbers['Li']

491 if (nthetahalf):
        mask = (x<1)*( (z > border2+1) *
493               (z<border2+a-1) + (z>border2+a+1)*(z<border2+2*a-1))
else:
495     mask = (x>1)*( (z > border2+1) *
```

```
            (z<border2+a-1) + (z>border2+a+1)*(z<border2+2*a-1))

slab4.numbers[mask] = ase.data.atomic_numbers['Li']

slab4 = geometry.sort(slab4)
if (vw == 1):
    view(slab4)
elif (vw == 2):
    slab4.write('POSCAR')
#Al.repeat((1, 1, 12)).write('POSCAR', vasp5=True)
elif (vw == 3):
    rfile = ase.io.read('CONTCAR')
    view(rfile)

########################      END LI     ############################
```

../mkposcar.py

Appendix E

Cphase/POSCAR file

```
Si Mg Cu Al Mg
   1.00000000000000
     10.2977248586763093     0.0000000000000000     0.0000000000000000
      0.0000000000000002     4.0391000000000004     0.0000000000000000
     -1.5275508075609407     0.0000000000000006     7.9324588729044114
   Si   Mg   Cu   Al   Mg
     6    6    2    2    2
Direct
  0.4880001572195170  0.5000000000000000  0.2479616815621668
  0.5119998427804830  0.0000000000000000  0.7520383184378332
  0.8289965254399192  0.5000000000000000  0.0729221947589878
  0.1710034745600808  0.0000000000000000  0.9270778052410122
  0.8342099448141411  0.5000000000000000  0.5913961402392403
  0.1657900551858589  0.0000000000000000  0.4086038597607597
  0.6347372648772236  0.0000000000000000  0.0810722486370068
  0.3652627351227764  0.5000000000000000  0.9189277513629861
  0.6384108533112567  0.0000000000000000  0.4875558923780687
  0.3615891466887504  0.5000000000000000  0.5124441076219313
  0.9022066367592743  0.0000000000000000  0.3414215157560818
  0.0977933632407257  0.5000000000000000  0.6585784842439182
  0.7217442349405516  0.5000000000000000  0.3066290234965976
  0.2782557650594484  0.0000000000000000  0.6933709765034024
  0.6702587545049425  0.5000000000000000  0.7919659205699929
  0.3297412454950575  0.0000000000000000  0.2080340794300071
  0.9031693624205914  0.0000000000000000  0.8569105459737116
  0.0968306375794086  0.5000000000000000  0.1430894540262884
```

```
28  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
30  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
32  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
34  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
36  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
38  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
40  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
42  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
44  0.00000000E+00  0.00000000E+00  0.00000000E+00
    0.00000000E+00  0.00000000E+00  0.00000000E+00
```

../cphase.CONTCAR

Bibliography

[1] J.M. Rosalie and L. Bourgeois. *Acta Materialia*, 60:6033–6041, 2012.

[2] S. Min and X. Daihong. *Science in China Series E: Technological Sciences*, 49:582–589, 2006.

[3] J. M. Silcock, T. J. Heal, and H. K. Hardy. *Journal of the Institute of Metals*, 82:24, 1953-54.

[4] L. Bourgeois, C. Dwyer, M. Weyland, J. Nie, and B. C. Muddle. *Acta Materialia*, 59:7043–7050, 2011.

[5] S. Wenner, C. D. Marioara, S. J. Andersen, M. Ervik, and R. Holmestad. *Materials Characterization*, 106:226–231, 2015.

[6] M. Torsæter, F. J. H. Ehlers, C.D. Marioara, S.J. Andersen, and R. Holmestad. *Philosophical Magazine*, 92:3833–3856, 2012.

[7] C. Wolverton and V. Ozolinš. *Physical Review Letters*, 86:5518, 2001.

[8] V. Vaithyanathan, C. Wolverton, and L. Q. Chen. *Acta Materialia*, 52:2973–2987, 2004.

[9] C. Wolverton, X.-Y. Yan, R. Vijayaraghavana, and V. Ozoli. *Acta Materialia*, 50:2187–2197, 2002.

[10] S.B. Dunkerton. *Procedure Development and Practice Considerations for Diffusion Welding, Welding, Brazing and Soldering*, volume 6. ASM International, 1993.

[11] A. Wilm. *Metallurgie : Zeitschrift für die gesamte Hüttenkunde*, 8:225–227, 1911.

[12] D. A. Porter and K. E. Easterling. *Phase Transformations in Metals and Alloys, 2nd Edition.* Chapman and Hall, 1992.

[13] A. Kelly and R. B. Nicholson. *Progress in Materials Science*, 10:151–391, 1963.

[14] G. Thomas and J. Nutting. *Acta Materialia*, 7:515, 1959.

[15] S. Sampath and R. Janisch. *Journal of Physics: Condensed Matter*, 25:355005, 2013.

[16] M. Born and R. Oppenheimer. *Zur Quantentheorie der Molekeln*, 84:457–486, 1927.

[17] D. R. Hartree. *Proceedings of the Royal Society of London*, A113:621, 1928.

[18] J. C. Slater. *Physical Review Letters*, 81:385, 1951.

[19] V. Fock. *Zeitschrift für Physik*, 61:126, 1930.

[20] W. Paulie. *Zeitschrift für Physik*, 31:765–783, 1925.

[21] PDF control of quantum many-body correlations. https://www.mpg.de/967764/CPT02Quantumbasetext.pdf. Accessed: 2016-06-01.

[22] P. Hohenberg and W. Kohn. *Physical Review Letters*, 136:B864, 1964.

[23] W. Kohn and L. J. Sham. *Physical Review Letters*, 140A:1133, 1954.

[24] J. Harris. *Physical Review Letters*, B31:1770, 1985.

[25] Slides comparison of exchange-correlation functionals: from lda to gga and beyond. http://helper.ipam.ucla.edu/publications/maws3/maws3_5880.pdf. Accessed: 2016-06-01.

[26] D. C. Langreth and M. J. Mehl. *Physical Review Letters*, B28:1809, 1983.

[27] D. D. Koelling and G. O. Arbman. *Journal of Physics F: Metal Physics*, F5:2041, 1975.

[28] J. P. Perdew and A. Zunger. *Physical Review Letters*, B23:5048, 1981.

[29] J. P. Perdew and Y. Wang. *Physical Review Letters*, B45:13244, 1992.

[30] Kieron Burke John P. Perdew and Matthias Ernzerhof. *Physical Review Letters*, A77:3865, 1996.

[31] P. Schwerdtfeger. *Physical Review Letters*, 12:3143–3155, 2011.

[32] E. Fermi. *Ricerca Scientifica*, 7:13–52, 1936.

[33] List of QCC programs wikipedia. https://en.wikipedia.org/wiki/List_of_quantum_chemistry_and_solid-state_physics_software. Accessed: 2016-05-30.

[34] J. C. Slater. *Physical Review Letters*, 51:846, 1937.

[35] P. E. Blöchl. *Physical Review Letters*, B50:17953, 1994.

[36] H. J. Monkhorst and J. D. Pack. *Physical Review Letters*, B13:5188, 1976.

[37] VASP vienna ab initio simulation package. https://www.vasp.at/. Accessed: 2016-04-19.

[38] G. Kresse and J. Furthmüller. *Physical Review Letters*, B54:11169, 1996.

[39] G. H. F. Diercksen and S. Wilson. *Methods in Computational Molecular Physics*. Springer, 1983.

[40] ASE atomic simulation environment. https://wiki.fysik.dtu.dk/ase/. Accessed: 2016-04-19.

[41] F. Birch. *Physical Review Letters*, 71:809, 1947.

[42] W. M. Stobbs and G.R. Purdy. *Acta Materialia*, 26:1069–1081, 1978.