TU TECHNISCHE
UNIVERSITÄT
WIEN

DIPLOMA THESIS

Impact of vacancies on the phase stability of the quasi-binary systems Ta-Al-N and Nb-Al-N

by

Ferdinand Pacher

0928386

carried out for the purpose of obtaining the degree Master of Science (MSc. or DI)
under supervision of

Univ. Prof. Dipl.-Ing. Dr.mont. Paul Heinz Mayrhofer

and

Univ. Lektor Dr. David Holec

E 308 - Institute of Material Science and Technology
special field Material Science

submitted at

Technische Universität Wien
Getreidemarkt 9/164-CT
1060 Wien

Vienna, March 23, 2017

Affidavit:
I declare in lieu of oath, that I wrote this thesis and performed the associated research myself,
using only literature cited in this volume.

Date	*Signature*

Acknowledgements

Contents

Contents

Abstract

Abstract

Within this work, the quasi-binary transition metal-aluminum-nitride alloys Ta-Al-N and Nb-Al-N were studied using *ab initio* methods based on density functional theory (DFT). The aim was to clarify the influence of vacancies on the stability of various phases known for these two systems. For the Ta-Al-N system, the hexagonal B4 (wurtzite), B_k, B_h, and ε as well as the cubic B1 (rock salt) structures were taken into account. The study of the Nb-Al-N system covered the hexagonal B4, B_k, and B_i and the cubic B1 structures.

Starting from fully occupied lattices, vacancy concentrations up to a maximum of 11 % (22 % empty sites on one sublattice) were introduced on either the metal or the nitrogen sublattice, and spanning the whole composition range of $Ta_{1-x}Al_x$/$Nb_{1-x}Al_x$ on the metal sublattice. The solid solutions were modeled using supercells with 54-72 for Ta- and 64-72 atoms for Nb-Al-N-structures. The atom positions were determined using the special quasi-random structure (SQS) approach.

All calculations were performed with the Vienna Ab-initio Simulation Package using a projector augmented wave function basis and GGA-PW91 exchange-correlation potentials.

The calculated energies were used to construct phase diagrams showing the stable structures as functions of vacancy and aluminum content. Furthermore, the structures were structurally characterized.

It is shown that cubic B1 structures containing either metal or nitrogen vacancies are more favorable than the perfect configurations with fully occupied lattices over a wide range of aluminum fractions. In all other phases considered in the calculations, however, vacancies have the opposite effect and destabilize the respective structures. As a result, the stability range of the cubic B1 phase is extended by the presence of vacancies in both the Ta-Al-N as well as in the Nb-Al-N system.

1

Kurzfassung

In dieser Arbeit wurden die quasi-binären Übergangsmetall-Aluminium-Nitridsysteme Ta-Al-N und Nb-Al-N mittels *ab initio* Methoden basierend auf der Dichtefunktionaltheorie (DFT) untersucht. Das Ziel war, den Einfluss von Fehlstellen auf die Stabilität der verschiedenen Kristallstrukturen, die für diese beiden Systeme bekannt sind, zu klären. Für das Ta-Al-N System wurden die hexagonalen B4- (Wurtzit), B_k-, B_h-, und ε-Strukturen sowie die kubische B1-Struktur (NaCl) berücksichtigt. Die Untersuchung des Nb-Al-N Systems umfasste die hexagonalen B4-, B_k- und B_i-Strukturen, und die kubische B1-Struktur.

Ausgehend von voll besetzten Gittern, wurden über die gesamte Kompositionsbreite $Ta_{1-x}Al_x/$ $Nb_{1-x}Al_x$ des Metallgitters, entweder auf dem Metall- oder dem Stickstoffgitter Fehlstellen bis zu einem Ausmaß von maximal $11\,\%$ eingebracht (bis zu $22\,\%$ der Gitterplätze eines Teilgitters unbesetzt). Die Legierungen wurden mittels Simulationszellen von 54-72 für das Ta- und 64-72 Atomen für das Nb-Al-N-System modelliert. Die Besetzung der Gitterplätze in den Mischkristallen wurde durch die sogenannte „special quasi-random structure" (SQS) Methode bestimmt.

Alle Berechnungen erfolgten mittels „Vienna Ab-initio Simulation Package". Es wurden Basisfunktionen, die durch den „projector augmented wave" Ansatz erweitert wurden und Austausch-Korrelations Potentiale vom Typ GGA-PW91 verwendet.

Mit den berechneten Energien wurden Phasendiagramme konstruiert, welche die stabilen Phasen der Systeme in Abhängigkeit von Aluminium- und Fehlstellengehalt darstellen. Zudem wurden die Phasen bezüglich ihrer Strukturparameter charakterisiert.

Es konnte gezeigt werden, dass für die kubische B1-Struktur über eine große Spanne an Zusammensetzungen sowohl Metall- als auch Stickstofffehlstellen energetisch günstiger sind als voll besetzte Kristallgitter. Alle anderen Strukturen reagieren mit einer Erhöhung ihrer Gesamtenergie auf die Defekte, werden also von diesen destabilisiert. Gemeinsam führen diese, durch die Fehlstellen hervorgerufenen Effekte, sowohl im Ta-Al-N-, als auch im Nb-Al-N-System zu einer Erweiterung des stabilen Bereichs der kubischen Phase.

Introduction

The aim of this work is to use *ab initio* methods to investigate the phase stability of the quasi-binary alloy systems Ta-Al-N and Nb-Al-N. Especially the role of vacancies in this context should be clarified, as they have been shown to strongly affect the structure and the properties of closely related materials [1–3]. Their explicit consideration in the density functional theory (DFT) simulations therefore promises new insights and more accurate predictions for improved design of materials. Both alloys belong to the material class of transition metal nitrides (TMNs), which will be shortly introduced in this chapter. Furthermore, the boundary systems Al-N, Ta-N, Nb-N are presented briefly in terms of their properties, crystallographic phases, which are important for the actual DFT calculations, and applications.

Chapter 2 provides the theoretical background of DFT in general, and discusses approaches to reduce computational costs of the simulations while preserving their accuracy. Additionally, difficulties in modeling random alloys are explained and an approach used to deal with them is shown.

Chapter 3 deals with technicalities of the Vienna Ab-initio Simulation Package (VASP), the software used to carry out the DFT calculations, and describes the results of the performed convergence tests, problems imposed by these results and their solutions.

Finally, Chapter 4 introduces the methodological concepts for the phase stability assessment carried out in this work, and presents and discusses the actual results.

1.1   Transition metal nitrides

As the name of this rather large class of materials already indicates, it comprises the compounds of nitrogen with the elements of groups IB-VIIIB of the periodic table. Based on the dominant bonding characteristics, electronic structure and properties, they can be classified into three groups, although this classification is rather vague for some of them. Nevertheless,

it allows a rough categorization, which is helpful for a preliminary assessment of the behavior of the individual materials.

The nitrides of the transition metals of groups IB-IIIB can be attributed to the group of ionic or salt-like nitrides [4, 5]. Besides them the alkali and alkaline-earth metals (groups IA and IIA) and some elements of groups IVA and VA form compounds associated with this group. Due to a relatively large difference in electro-negativity of $1.3 - 2.1$ of these elements to nitrogen, ionic bonding dominates. Their chemical stability is rather low and they are susceptible to hydrolysis [4–6]

Nitrides of elements from groups IVB-VIB are usually called interstitial nitrides [4, 6, 7]. The atomic size of those elements is much larger than that of nitrogen. For this reason, the nitrogen atoms can populate interstitial sites on the metal sublattice. The bonding of these compounds includes both metal-metal and metal-nitrogen interactions resulting in dominant metallic bonding with smaller covalent and ionic contributions [4, 8–11]. Many properties, such as their high electrical and thermal conductivity or magnetic and optical behavior, resemble those of the corresponding metallic parent materials, while their hardness and oxidation resistance benefit from the covalent and ionic contributions to the inter-atomic bonds [4, 8, 10–12]. Of the late transition metals in groups VIIB-VIIIB only Mn, Fe, Co, Ni, Te and Re form nitrides [4, 8]. They are referred to as intermediate nitrides by some authors [4]. Their bonding characteristics are similar to the interstitial nitrides. But the nitrogen content in them is lower and the resulting phases have more complicated crystal structures. Furthermore, they show lower chemical and thermal stability [4, 7, 13]. Fe_4N and FeN_{1-x} are, however, very important for strengthening of steels by nitriding, as they form hard precipitates [8, 14, 15].

Both Ta and Nb are group VB elements and therefore form interstitial nitrides. Interstitial transition metal nitrides of group IVB-VIB are industrially well established materials for various applications. They exhibit very high hardness and melting points, and are chemically very stable. Optical, magnetic and electric properties are essentially metallic, but the temperature dependence of the electrical conductivity is very low or even zero in many of them [8, 16].

In electronic applications, interstitial TMNs are deployed as diffusion barriers, adhesion promoters and thin film resistors in semiconductor devices because of their chemical stability and electric properties. Examples for materials in this field are TiN_x, crystalline and amorphous phases of WN, and different TaN structures [16, 17].

Their high hardness, thermal stability and oxidation resistance also make them common protective coating materials for machining and forming tools as well as for engine parts [4, 7, 18, 19]. Among the first coating systems used for this purpose were TiN and CrN, which are still commonly used nowadays in industry [19]. Extending the binary systems by additional elements such as aluminum or carbon allowed improving their properties and

thereby increasing the lifespan and performance of the coated parts [17, 20, 21]. The most prominent example, which has been intensively studied throughout the years is Ti-Al-N [1, 17, 22–28]. Nevertheless, the demand for even more capable thin film materials for these applications is not exhausted yet. Present research in this area explores multi-layered coatings, their architectural design, and a further expansion of the used alloy systems [10, 29–31]. For example, Koller *et al.* investigated coatings of the quaternary Ti-Al-Ta-N system and Ti-Al-N/Ta-Al-N multi-layers [32, 33].

One very important feature of transition metal nitrides of groups IVB-VIB is their defect structure. Many of their phases do not follow strict stoichiometries and very large vacancy concentrations can be found on both, metal and nitrogen sites. The defect structure has a significant effect on material properties and is strongly influenced by the used synthesis [8, 16, 17].

It is therefore impossible to ignore their influence in *ab inito* studies [34, 35]. Yu, Stampfl, Freeman *et al.* and Koutná *et al.* published a number of papers on the influence of different point defects on the stability and electrical properties of the binary Ta-N system [3, 36–38]. Regarding ternary systems, Alling *et al.* investigated the influence of nitrogen vacancies on the decomposition behavior of Ti-Al-N [1]. Euchner and Mayrhofer researched the effect of vacancies on its phase stability and structural and mechanical properties [2]. They showed that previous calculations correctly predicting the phase transition of the system despite neglecting the influence of defects were more or less a lucky chance, since Ti-Al-N is not very sensitive to them regarding its phase stability. Its physical properties are, however, affected by the presence of vacancies [2].

This work aims to expand the theoretical knowledge on the influence of vacancies in TMNs by investigating their impact on Ta-Al-N and Nb-Al-N.

1.1.1 Material systems and crystal structures

In the following, the binary boundary systems of both, Ta-Al-N and Nb-Al-N, are discussed shortly with an emphasis on metal-to-nitrogen compositions around 1. Also a short introduction into current research of the quasi-binary systems is given.

Al-N

AlN is the only compound of the Al-N system [4, 12]. It is a IIIA semiconductor and can be classified as covalent nitride due to its dominant bonding characteristics, although ionic bonding components are also present in the material [4, 39]. Applications are mostly based on its high thermal conductivity and high electric resistivity which are caused by its large band gap ($> 6\,\text{eV}$ [39]). These properties make AlN an attractive substrate for semiconductor devices and components in power electronic applications [12, 40, 41]. But

also its semiconducting nature is gaining importance for example as LED [39, 42].

The crystal structure of AlN under equilibrium conditions is the wurtzite B4 structure (space group 186, pearson symbol hP4, Fig. 1.1b) [4, 12]. At pressures above $\sim 16\,\mathrm{GPa}$, however, the metastable cubic B1 phase (space group 225, pearson symbol cF8, Fig. 1.1a) is also reported [43].

Ta-N

Binary Ta-N has a number of different applications due to its versatile properties [44]. It is used in electronic components as a diffusion barrier, because of its chemical stability and good adhesion to Cu as well as to common substrates, such as Si and InP [17, 45–48]. It is also used as a thin-film resistor, because of its small temperature coefficient of resistivity [7, 49–51]. Furthermore, its favorable mechanical properties make Ta-N materials interesting as new protective coatings and the system is gaining importance especially in the context of multi-layer arrangements and as components of ternary and quaternary alloys [52–55].

The binary Ta-N system exhibits a very large number of different stable and metastable phases [37, 54, 56, 57]. At Ta/N= 1 the hexagonal ε-phase (space group 189, pearson symbol hP6, Fig. 1.1d) is generally accepted to be the stable phase at ambient pressure and temperature [9, 56, 58, 59]. But the systems shows a large number of metastable phases not appearing in the equilibrium phase diagram and it is apparently very sensitive to the used synthesis techniques and conditions [17, 23, 37, 49, 60, 61]. For a strict Ta/N ratio of 1, the cubic B1 phase and the hexagonal B_h phase (space group 187, pearson symbol hP2, Fig. 1.1e) are known to be such metastable phases [9, 56, 58, 62].

Nb-N

The superconducting properties and high critical temperature for superconductivity of Nb-N have long been the main topic of interest in research on this system [9, 12, 63–65]. Especially the possible application in tunneling junctions received much attention [12, 66–70]. Next to the high electrical conductivity, the high melting point and chemical inertness of Nb-N have motivated to use them as cathode material for field emission displays [71]. Additionally, the possible application as diffusion barrier is reported [72, 73]. Lately, more attention has been paid to a possible application of Nb-N as protective coatings due to its high hardness, high melting point and wear resistance [74–77].

Similarly to the Ta-N system, also the Nb-N system shows a large number of different phases. For NbN with a Nb/N ratio ~ 1, the hexagonal B_i structure (space group 194, pearson symbol hP8, Fig. 1.1f) is found to be most stable at ambient conditions [9, 78–80]. But also several metastable phases have been found for the near-stoichiometric compositions [9, 79, 80]. The cubic B1 phase is generally accepted to be a high temperature phase occurring at $T > 1230 - 1270\,^{\circ}\mathrm{C}$ [9, 79, 80]. But it is commonly found in thin films deposited by

physical vapor deposition (PVD) [74, 76, 77, 81]. Another metastable phase, the hexagonal $B8_1$ structure (space group 194, pearson symbol hP4, Fig. 1.1g), was originally found during the transformation from the cubic B1 to the hexagonal B_i [78, 79], but can also be stabilized with PVD methods [81–83].

Ta-Al-N

This alloy system is a subject of intensive research as a possible material for protective hard coatings, either alone or as part of a multilayer arrangement [33]. A recent study of Koller *et al.* was concerned with cathodic arc evaporated thin films with a composition of $Ta_{0.89}Al_{0.11}N_{1-1.2}$. In the as deposited state, the cubic B1 structure was found to be stable [84]. Chen *et al.* deposited a series of four Ta-Al-N films by reactive magnetron co-sputtering, three of which which had Ta-rich compositions up to $Ta_{0.7}Al_{0.3}N$ and crystallized in the B1 phase. The Al-rich material showed blurred XRD-patterns, which they ascribed to a hexagonal B4 texture [85].
Zhang reported a single phase B1 structure in $Ta_{1-x}Al_xN$ films synthesized by DC reactive magnetron sputtering up to $x = 0.36$ and in a dual phase arrangement together with the hexagonal B4 phase up to $x = 0.65$ [86].
These results are quite interesting, because neither of the two boundary systems Ta-N and Al-N occurs in the cubic B1 structure at equilibrium conditions. Since this phase is only metastable in the system it would undergo decomposition upon thermal loading. Furthermore, *ab initio* simulations of Holec *et al.* predicted the hexagonal B_h phase to be stable for low aluminum contents up to $x = 0.2$ [87].

Based on the information on the binary systems Al-N and Ta-N as well as the experimental and theoretical work on the ternary Ta-Al-N summed up above, the calculations presented in this work considered the cubic B1 structure, and the hexagonal structures B4, ε and B_h. In addition to them, the hexagonal B_k structure (space group 194, pearson symbol hP4, Fig. 1.1c) was included, because of its similarity to both the cubic B1 and the hexagonal B4 structure [28, 88]. It is for this reason likely to be an intermediate phase during the transition between the two phases and its consideration thus promises a more accurate description of this transition [28]. The crystallographic structures included in the calculations for the Ta-Al-N system are illustrated in Figs. 1.1a to 1.1e and listed in Table 1.1.

Nb-Al-N

The available literature on this system is rather scarce. Selinder *et al.* were the first to investigate the influence of the aluminum content on the crystallographic structure of $Nb_{1-x}Al_xN$ materials [89]. They produced a film with continuously varying composition by reactive triode sputtering and characterized it in terms of crystal structure. A pure film of cubic B1

phase was found over a wide range of aluminum fractions x. Even for their lowest aluminum content $x \sim 0.09$, the hexagonal B_i phase did not occur. Barshila *et al.* also reported the B1 structure in a Nb-Al-N material with $x = 0.37$, but they noted that the oxygen content in their films was as high as $5.2\,\text{at.}\%$ [90]. The cubic phase in $Nb_{1-x}Al_xN$ was confirmed by experiments of Franz *et al.*, who reported it up to $x = 0.56$ [91]. Zhang investigated both binary Nb-N and quasi-binary Nb-Al-N films and found a mixture of the hexagonal B_i and the cubic B1 phases for Nb-N films with nitrogen over-stoichiometry, but a single-phase cubic structure in $Nb_{1-x}Al_xN$ materials up to $x = 0.44$ [86]. *Ab inito* studies by Holec *et al.*, however, predicted the hexagonal $B8_1$ phase to be the most stable one in $Nb_{1-x}Al_xN$ alloys up to $x = 0.14$, while for $0.14 \leq x \leq 0.7$ the cubic B1 phase was the most stable one before the B4 structure appears for $x \geq 0.7$ [92].

With this compiled information it was decided to include the cubic B1 and the hexagonal B_i, B4 and B_k structures in the present calculations regarding the Nb-Al-N system. The structures considered for the Nb-Al-N system are shown in Figs. 1.1a to 1.1c and Fig. 1.1f and listed in Table 1.1.

Table 1.1: Table of crystal structures considered in this work.

structure	prototype	space group	pearson symbol	nr. of atoms in conventional cell	Ta-Al-N	Nb-Al-N
B1	NaCl	225	cF8	8	✓	✓
B4	ZnS	186	hP4	4	✓	✓
B_k	BN	184	hP4	4	✓	✓
ε	TaN	189	hP6	6	✓	
B_h	WC	187	hP2	2	✓	
B_i	TiP	194	hP8	8		✓

(a) cubic B1   (b) hexagonal B4   (c) hexagonal B_k

(d) hexagonal ε (e) hexagonal B_h (f) hexagonal B_i

(g) hexagonal $B8_1$
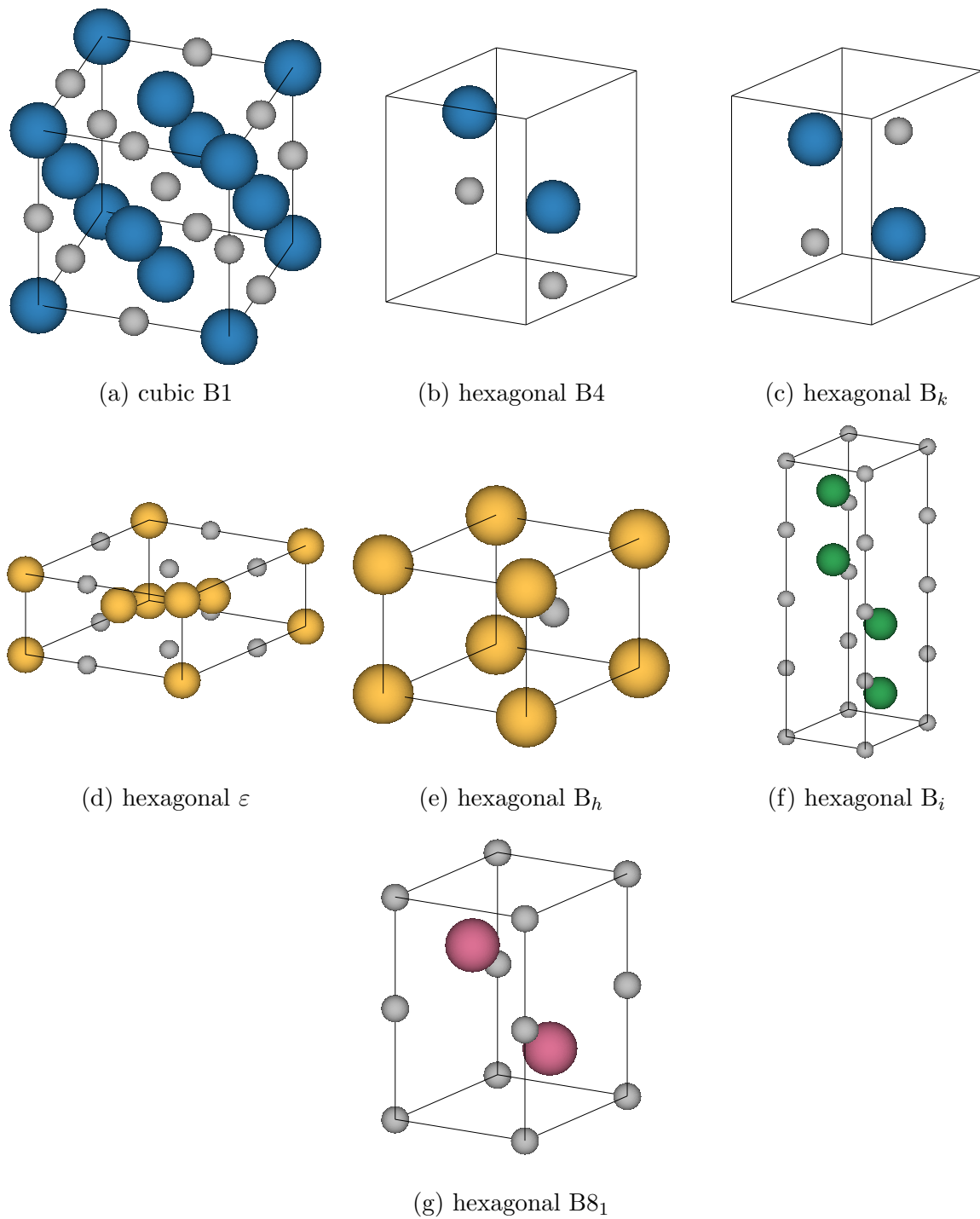
Figure 1.1: Crystal structures considered in this thesis. The nitrogen atoms are represented by the smaller gray spheres, while the larger colored spheres illustrate the metal atoms. Structures with yellow metal atoms are incorporated in the Ta-Al-N system, structures with green ones in the Nb-Al-N. The structures with blue atoms are included in both systems. The hexagonal $B8_1$ structure, which is found in the Nb-Al-N system, is not considered in the calculations and is therefore identified by violet spheres. The images were created with VESTA [93].

Theoretical background

In this chapter, the background of this work from the simulation perspective will be explained. Section 2.1 deals with the approach of density functional theory (DFT) to the quantum mechanical many-body problem and describes the basic principles used in its practical applications. Section 2.2 furthermore discusses concepts used in the treatment of crystalline solids with DFT as well as some important aspects and methods used to choose an adequate model system for the calculations.

2.1 Density functional theory

This section will give a brief introduction to the basics of DFT methods. The assumptions and approximations made to tame the complicated equations describing quantum mechanics are explained in a nutshell. Its approach to the topic is following the books by J. G. Lee [94], S. Cottenier [95] and a course on atomistic material modeling given at Vienna UT [96].

2.1.1 The underlying problem

At the beginning of the 20th century, the great minds such as Bohr, Schrödinger, De Broglie, and many others, have succeeded in uncovering the principles of quantum mechanics, and have given it an accurate mathematical description. The result of this effort is the many-body wave equation, or Schrödinger equation [97]:

$$\hat{H}\psi(\mathbf{R}_i, \mathbf{r}_i) = E\psi(\mathbf{R}_i, \mathbf{r}_i). \qquad (2.1)$$

In this equation, ψ is the unknown wave function, E is the system energy, and $\hat{H}$ is the Hamilton operator. The variables of the function, $\mathbf{R}_i$ and $\mathbf{r}_i$, are the coordinates of the nuclei and the electrons, respectively. Any atomic system can, in principle, be fully described by correctly formulating its Hamiltonian, and solving the resulting partial differential equation 2.1.

The first step to simplify the problem is to confine it to the solution for the ground state at $0\,\mathrm{K}$. If, in addition to that, gravitational and relativistic effects are omitted, the defining term of Eq. 2.1, the Hamiltonian, takes the following form [95]:

$$\hat{H} = -\overbrace{\frac{\hbar^2}{2}\sum_i \frac{\nabla^2_{\mathbf{R}_i}}{M_i}}^{\hat{T}_n} -\overbrace{\frac{\hbar^2}{2}\sum_i \frac{\nabla^2_{\mathbf{r}_i}}{m_e}}^{\hat{T}_e} -$$
$$-\underbrace{\frac{1}{4\pi\varepsilon_0}\sum_{i,j}\frac{e^2 Z_i}{|\mathbf{R}_i - \mathbf{r}_j|}}_{\hat{V}_{n\text{-}e}} +\underbrace{\frac{1}{8\pi\varepsilon_0}\sum_{i\neq j}\frac{e^2 Z_i Z_j}{|\mathbf{R}_i - \mathbf{R}_j|}}_{\hat{V}_{n\text{-}n}} +\underbrace{\frac{1}{8\pi\varepsilon_0}\sum_{i\neq j}\frac{e^2}{|\mathbf{r}_i - \mathbf{r}_j|}}_{\hat{V}_{e\text{-}e}} . \quad (2.2)$$

$\hat{T}_n$ and $\hat{T}_e$ are operators giving the kinetic energies of the nuclei and the electrons, respectively. $\hat{V}_{n\text{-}e}$ gives the potential energy for nucleus-electron interactions, $\hat{V}_{n\text{-}n}$ for nucleus-nucleus interactions, and $\hat{V}_{e\text{-}e}$ for electron-electron interactions.

Solving Eq. 2.1 with this Hamiltonian is, however, not possible even for simple di-atomic systems because of its complexity. A further simplification of the problem is therefore needed.

Max Born and Robert Oppenheimer suggested to separate the movement of the atomic nuclei and the electrons, and to treat the nuclear and the electronic problem successively [98]. This is justified by the fact that the mass of the neutrons and protons forming the atomic core is much higher than that of the electrons ($m_p = 1.673 \cdot 10^{-27}\,\mathrm{kg}$, $m_n = 1.675 \cdot 10^{-27}\,\mathrm{kg}$, $m_e = 9.11 \cdot 10^{-31}\,\mathrm{kg}$). Thus, the electrons react much faster to forces acting upon them and follow the motion of the nuclei immediately. In the decoupled electronic problem, the wave equation is therefore solved with fixed coordinates of the nuclei. The kinetic energy of the nuclei $\hat{T}_n$ drops out of the Hamiltonian and the nucleus-nucleus interaction $\hat{V}_{n\text{-}n}$ becomes a constant. The nuclei's influence on the electrons $\hat{V}_{n\text{-}e}$ is described by an external potential $\hat{V}_{ext}$. The terms for the kinetic energy of the electrons and their mutual interaction $\hat{T}_e$ remain unchanged and the effective Hamiltonian becomes [94]:

$$\hat{H} = \hat{T}_e + \hat{V}_{e\text{-}e} + \hat{V}_{ext} . \quad (2.3)$$

The solution of 2.1 with this $\hat{H}$ describes the electronic ground state for a specific configuration of the nuclei's positions. Born and Oppenheimer showed that the result of the electronic problem acts as the potential energy for the nuclear problem, thus providing the forces acting on the atomic cores [98].

One important feature about Eq. 2.3 is, that the terms for the kinetic energy, $\hat{T}_e$, and the interaction between the electrons, $\hat{V}_{e\text{-}e}$, are universal and take the same form in different systems. This means that all material-specific information to be gained from solving Eq. 2.1 with 2.3 comes from the external potential, $\hat{V}_{ext}$, of the nuclei [99].

Although the above described approximation reduces the complexity of the problem, a solution for Eq. 2.1 is still out of reach for practically relevant systems.

2.1.2 Theorems of Hohenberg and Kohn

Picking up an approach that Llewellyn Hilleth Thomas and Enrico Fermi came up with in the 1920s, Pierre Hohenberg and Walter Kohn shifted their focus away from the electronic wave function towards the electron density [100–103]. They were able to proof that the electron density describes a system equally well as the wave function and can give information about any physical property of the system related to an observable. This is nowadays known as the (Hohenberg-Kohn) density functional theory [102]

If the wave function is already known, the electron density can be written as [95]:

$$n(\mathbf{r}) = \sum_{i=1}^{N} \int \psi^*(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_i \equiv \mathbf{r}, \ldots, \mathbf{r}_N) \cdot$$

$$\cdot \psi(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_i \equiv \mathbf{r}, \ldots, \mathbf{r}_N) d\mathbf{r}_1 d\mathbf{r}_2 \ldots d\mathbf{r}_i \ldots d\mathbf{r}_N. \quad (2.4)$$

Since the wave function is determined by the external potential, the electron density is also fully defined by V_{ext}. The first theorem of Hohenberg and Kohn states that the external
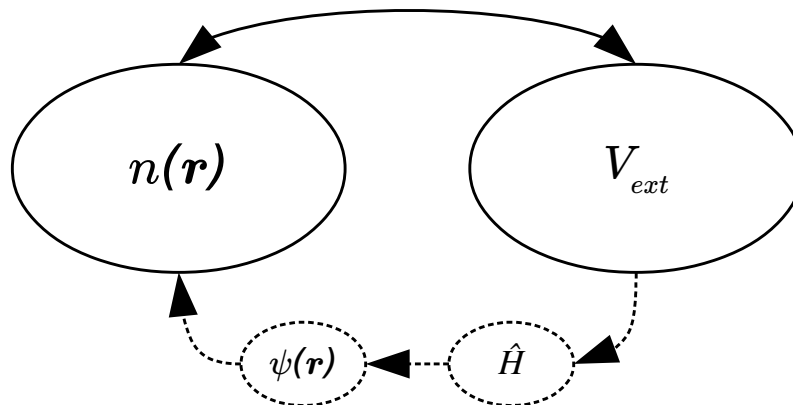
Figure 2.1: The first theorem of Hohenberg and Kohn states a one two one correspondence between the external potential V_{ext} and the ground state electron density $n(\mathbf{r})$.

potential V_{ext} is a unique functional up to a constant of the ground state electron density $n(\mathbf{r})$ of a system [102]. V_{ext} in turn defines the Hamiltonian, and therefore $\psi(\mathbf{r})$ and $n(\mathbf{r})$ resulting in a one-to-one correspondence [95]. This theorem thus warrants that no information is lost when searching directly for the electron density instead of the wave function. All properties of a system in a ground state can be attained from its ground state electron density. Or more formally speaking: The ground state expectation value of any observable is a unique functional of the ground state electron density [95, 99].

In their second theorem, Hohenberg and Kohn defined a total energy functional of a system as [102]:

$$E[n] = \int V_{ext} \cdot n(\mathbf{r}) d\mathbf{r} + F[n]. \quad (2.5)$$

While the first term in Eq. 2.5 represents the potential energy arising from the interaction between the electrons and the external potential, $F[n]$ is a functional describing the kinetic energy of the electrons and the energy due to their interaction with each other. Since this term depends only on $n(\mathbf{r})$, it is universal and valid for any external potential [102]. The theorem further states that the electron density n, for which $E[n]$ reaches minimum value, is the ground state electron density and $E[n]$ is the corresponding ground state total energy. If $F[n]$ can be determined, the problem of finding the ground state electron density (and therefore the ground state energy) can be solved by minimizing $E[n]$ using the variational principle [102].

2.1.3 Kohn-Sham-approach

Walter Kohn and Li Sham endeavored to find such a formulation for $F[n]$ and suggested its approximate form, and thereby presented a way how to practically deal with DFT [104]. Hohenberg and Kohn pointed out that the electrostatic Coulomb energy of the electrons is one part of $F[n]$ and that it can be written as [102]:

$$F[n] = \frac{1}{4\pi\varepsilon_0} \int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r} d\mathbf{r}' + G[n] \,. \tag{2.6}$$

$G[n]$ is again a unique functional of n. It accounts for the kinetic energy and all energetic terms of quantum mechanical electron-electron interactions not included in the classical Coulomb term.

Starting from this expression, Kohn and Sham formulated $G[n]$ as [104]:

$$G[n] = T_S[n] + E_{xc}[n] \,, \tag{2.7}$$

with T_S being the kinetic energy of a system of non-interacting electrons. E_{xc} is called exchange-correlation energy. It covers the effects of Pauli's exclusion principle (exchange) and of interactions between electrons of the same spin (correlation) as well as corrections for the error made by treating a non-interacting system [94, 105]. By inserting Eqs. 2.7 and 2.6 into 2.5, and using the variational principle, a Schrödinger-like equation for non-interacting particles, called Kohn-Sham equation, can be derived [94]:

$$\hat{H}_{KS}\phi_m = \varepsilon_m \phi_m \,. \tag{2.8}$$

$\hat{H}_{KS}$ is the Kohn-Sham Hamiltonian [95]:

$$\hat{H}_{KS} = -\frac{\hbar^2}{2m_e}\nabla^2 - \frac{e^2}{4\pi\varepsilon_0} \int \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d\mathbf{r}' + \hat{V}_{ext} + \hat{V}_{xc} \,. \tag{2.9}$$

The solutions ϕ_m of Eq. 2.8, the Kohn-Sham orbitals, cannot be used to obtain the real wave function of a system, nor do they have any particular physical meaning. But for a general

system of N electrons, the N solutions, ϕ_m, corresponding to the lowest energies, ε_m, can be used to construct the exact ground state electron density using [95]:

$$n(\mathbf{r}) = \sum_{m=1}^{N} \phi_m^*(\mathbf{r}) \cdot \phi_m(\mathbf{r}) \,. \tag{2.10}$$

Since the KS Hamiltonian (Eq. 2.9) depends on the electron density, Eq. 2.8 presents a self-consistent problem that must be solved iteratively [95]. Starting from an initially guessed $n(\mathbf{r})$, $\hat{H}_{KS}$ is constructed and Eq. 2.8 is solved. Using Eq. 2.10, a better approximation for $n(\mathbf{r})$ is calculated from the KS orbitals. This procedure is repeated until the difference between two successive iterations is sufficiently low and a convergence is reached (Fig. 3.1, electronic loop) [95, 96]. The Kohn-Sham equation 2.8 is significantly easier to solve than the problem for interacting electrons. For a system of N electrons, this would be a partial differential equation (PDE) depending on $3N$ coordinates. Using the Kohn-Sham approach, the problem is mapped on a single PDE of a function of coordinates. Nevertheless, the exact ground state electron density can be calculated only, if the exact expression for the exchange-correlation potential V_{xc} can be found.

2.1.4 Exchange-correlation functional

In practice, however, the exchange-correlation potential has to be approximated. This is usually done by defining an exchange-correlation functional whose derivative with respect to the electron density gives the potential [94, 95]:

$$\hat{V}_{xc} = \frac{\delta E_{xc}[n(\mathbf{r})]}{\delta n(\mathbf{r})} \,. \tag{2.11}$$

The first proposal how to construct E_{xc} was made by Kohn and Sham themselves in the very same paper presenting their approach [104].
In a so called local density approximation (LDA), the exchange-correlation energy functional takes a form:

$$E_{xc}^{\text{LDA}}[n(\mathbf{r})] = \int n(\mathbf{r}) \cdot \varepsilon_{xc}(n(\mathbf{r}))d\mathbf{r} \,, \tag{2.12}$$

where ε_{xc} is the exchange-correlation energy density (or energy per electron) approximated by a value of a homogeneous electron gas of the density $n(\mathbf{r})$. It is called local since ε_{xc} depends only on $n(\mathbf{r})$, making Eq. 2.12 a local functional.
By using an ε_{xc} depending on $n(\mathbf{r})$ and $\nabla n(\mathbf{r})$, the exchange-correlation energy becomes a semi-local functional. Potentials of the type:

$$E_{xc}^{\text{GGA}}[n(\mathbf{r})] = \int n(\mathbf{r}) \cdot \varepsilon_{xc}(n(\mathbf{r}), \nabla n(\mathbf{r}))d\mathbf{r} \,, \tag{2.13}$$

belong to a class called generalized gradient approximation (GGA) [94]. Within the GGA-family, many different variations exist. Prominent examples of the GGA-potentials are the

ones introduced by Perdew and Wang [106] and Perdew, Burke and Ernzerhof [107]. In addition to the above mentioned, more advanced potentials are also available, which take more variables into account in ε_{xc}. The increased accuracy they offer comes, however, at the cost of significantly higher computational demands.

In this work, GGA-potentials with the parametrization of Perdew and Wang are used [108, 109].

2.1.5 Finding a solution

To finally get a palpable solution to the KS-equation 2.8, a specific ansatz for the single-particle-wave functions is made. The KS-orbitals are expressed as a linear combination using a set of basis functions ϕ_p^{basis} and they therefore have the form [95]:

$$\tilde{\phi}_m(\mathbf{r}) = \sum_{p=1}^{P} c_p^m \phi_p^{\mathrm{basis}}(\mathbf{r}) \,. \tag{2.14}$$

To make this solution exact (within the previous approximations) the basis set would have to be infinitely large, which of course is practically impossible. A sufficiently accurate basis set of a finite size P has to be used instead [95]. The larger P is, the more accurate the expansions 2.14 become, but at the same time the computational costs increase. Additionally, the type of the basis set is an important factor impacting the efficiency of the calculations. Depending on the type of functions used, the size P necessary to get an accurate result can be significantly reduced. At the same time, the basis set must not influence the results by its own characteristics and should be able to describe any solution accurately [95].

Once the type and the size of the basis set is chosen, problem 2.8 transforms to an eigenvalue problem:

$$\begin{bmatrix} \cdots & \cdots & \cdots \\ \vdots & \int (\phi_i^{\mathrm{basis}})^* \cdot \hat{H}_{KS} \cdot \phi_j^{\mathrm{basis}} d\mathbf{r} - \varepsilon_m \cdot \int (\phi_i^{\mathrm{basis}})^* \cdot \phi_j^{\mathrm{basis}} d\mathbf{r} & \vdots \\ \cdots & \cdots & \cdots \end{bmatrix} \begin{bmatrix} c_1^m \\ \vdots \\ c_P^m \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \,. \tag{2.15}$$

Index m in Eqs. 2.8, 2.14 and 2.15 labels the solutions. Part of the index m is $\mathbf{k}$, a reciprocal vector. In molecules, only $\mathbf{k} = \mathbf{0}$ is necessary to attain the system properties. The case is more complicated for crystalline solids, and m includes a so-called band index n and a quasi-continuous $\mathbf{k}$-vector [95]. This topic will be addressed in a greater detail in Sections 2.2.1 and 3.1.1.

Vienna Ab-initio Simulation Package uses a plane wave basis set, which is very efficient for describing the smooth behavior of valence electron wave functions far away from the atomic nuclei. But both core and valence electrons have strongly oscillating wave functions in the vicinity of the nuclei [94, 95]. A very large number of plane waves is needed to describe

this behavior, resulting in a large number of equations in the system Eq. 2.15, and hence very long calculation times and high memory demands. There are several approaches how to overcome this problem. In VASP, the so called the pseudo-potential method (PP) is implemented, which will be explained shortly in the next section.

2.1.6 The pseudo-potential method

Depending on their position in the atomic configuration, electrons can be classified as either core or valence electrons. The core electrons are very localized at the positions of the atoms and their wave functions oscillate rapidly near the nucleus, but quickly decay with increasing distance. When using a plane wave basis set, this oscillatory behavior can only be described if the basis set is extremely large, which is not economical in computational terms. Since the involvement of the core electrons in the actual bonding interactions determining most physical properties is minimal, it is possible to limit their contribution to screening the potential of the nuclei and neglect them for the rest of the calculations. This approach is known as the frozen-core approximation, which is the first simplification step used by the pseudo-potential method [94].

The wave functions of the valence electrons usually oscillate heavily close to the atom cores as well, in order to remain orthogonal to the wave functions of the core electrons. With the frozen core approximation this is no longer necessary, since the core electrons are essentially removed. New effective potentials are constructed that incorporate the influence of the nuclei and the core electrons. These pseudo-potentials are designed in such a way, that beyond a certain radius r_c from the atomic cores, both the potential and the valence electron wave functions are exactly the same as in a full-potential all electron calculation. Within the regions around the cores defined by r_c, however, they smoothen the pseudo-wave functions and make them thereby representable with a plane wave basis set [94, 99, 110].

While the efficiency of a DFT calculation can be extremely increased by using a plane wave basis set in combination with pseudo-potentials, their creation is rather complex. Ideally, they should be able to describe all possible quantum-mechanical setups correctly, and reduce the necessary basis size to a minimum [95, 99]. Also, there are different types of pseudo-potentials. Some of them reconstruct the electron density in the spheres around the atomic nuclei by introducing so-called augmentation charges [111, 112]. The calculations carried out in this work are projector augmented wave method enabled potentials, described in the next section.

2.1.7 The projector augmented wave method

The PAW method was first introduced by Peter Blöchl [113] and bases on a transformation of the physical single-particle wave functions onto smooth pseudo-wave functions, which are

numerically easier to describe. The following should only sketch the method briefly and is based on two papers by Blöchl [113, 114] and one by C. Rostgaard [115], which offer a more detailed explanation of the method.

The basic concept of the PAW method is, that the true single-particle wave function ϕ_m is constructed from the so-called pseudo-wave function $\tilde{\phi}_m$. $\tilde{\phi}_m$ is nothing but the approximation of ϕ_m by a smooth basis set (e.g., plane waves) as described by Eq. 2.14. Due to the pseudo-potential approach, $\tilde{\phi}_m$ differs from ϕ_m close to the nuclei. To reconstruct ϕ_m there as well, a transformation is introduced, which alters the pseudo-wave function locally in a way that it adopts the correct, oscillating characteristics. The transformation can be written as

$$\phi_m(\mathbf{r}) = \hat{\mathcal{T}}\tilde{\phi}_m(\mathbf{r}) . \tag{2.16}$$

$\hat{\mathcal{T}}$ only acts in the direct vicinity of the atomic nuclei leaving most of the pseudo-wave function as it is, and can therefore be written as the unity plus a sum of local contributions, $S^{\mathbf{R}}$:

$$\hat{\mathcal{T}} = 1 + \sum_{\mathbf{R}} \hat{S}^{\mathbf{R}} , \tag{2.17}$$

where index $\mathbf{R}$ marks the atomic sites. Within a certain region, sometimes called an augmentation region, and outlined by a certain radius $r_c^{\mathbf{R}}$, the local contribution $\hat{S}^{\mathbf{R}}$ is defined using two sets of basis functions. The first set, the so-called pseudo partial waves $\tilde{\varphi}_i^{\mathbf{R}}$, is a complete basis of smooth functions inside the augmentation region. This is generally a different one than the basis set in Eq. 2.14 in the previous section. Using this the pseudo-wave function can be expanded within $r_c^{\mathbf{R}}$:

$$\tilde{\phi}_m(\mathbf{r}) = \sum_i a_i^{m,\mathbf{R}}\tilde{\varphi}_i^{\mathbf{R}}(\mathbf{r}) \quad \text{for} \quad |\mathbf{r} - \mathbf{R}| < r_c^{\mathbf{R}} . \tag{2.18}$$

The second set also forms a complete basis and consists only of functions orthogonal to the core wave functions. The so-called partial waves, $\varphi_i^{\mathbf{R}}$, are the target functions of the transformation. $\hat{S}^{\mathbf{R}}$ transforms every pseudo partial wave function $\tilde{\varphi}_i^{\mathbf{R}}$ of the augmentation region $\mathbf{R}$ on a partial wave function $\varphi_i^{\mathbf{R}}$ according to:

$$\varphi_i^{\mathbf{R}}(\mathbf{r}) = (1 + \hat{S}^{\mathbf{R}})\tilde{\varphi}_i^{\mathbf{R}}(\mathbf{r}) . \tag{2.19}$$

It therefore can be expressed as:

$$\hat{S}^{\mathbf{R}}\tilde{\varphi}_i^{\mathbf{R}}(\mathbf{r}) = \varphi_i^{\mathbf{R}}(\mathbf{r}) - \tilde{\varphi}_i^{\mathbf{R}}(\mathbf{r}) . \tag{2.20}$$

Within the augmentation region of site $\mathbf{R}$, $\hat{\mathcal{T}} = (1 + \hat{S}^{\mathbf{R}})$ holds. Combining this with Eqs. 2.16, 2.18 and 2.19 shows that the true wave function can be described as an expansion of the partial waves:

$$\phi_m(\mathbf{r}) = \hat{\mathcal{T}}\tilde{\phi}_m(\mathbf{r}) = \sum_i a_i^{m,\mathbf{R}}\varphi_i^{\mathbf{R}}(\mathbf{r}) \quad \text{for} \quad |\mathbf{r} - \mathbf{R}| < r_c^{\mathbf{R}} , \tag{2.21}$$

with exactly the same coefficients $a_i^{m,\mathbf{R}}$ as the pseudo-wave function is described by the pseudo partial waves. To restrict $\hat{S}^{\mathbf{R}}$ to the augmentation region, the true and the pseudo partial waves have to be identical outside of it:

$$\phi_m(\mathbf{r}) = \tilde{\phi}_m(\mathbf{r}) \quad \text{for} \quad |\mathbf{r} - \mathbf{R}| > r_c^{\mathbf{R}} \,. \tag{2.22}$$

The coefficients $a_i^{m,\mathbf{R}}$ of the expansions 2.18 and 2.21 are determined by inner products of the pseudo-wave function with the functions $\tilde{p}(\mathbf{r})$ of a third predefined set called projector functions:

$$a_i^{m,\mathbf{R}} = \int \tilde{p}_i^{\mathbf{R}}(\mathbf{r}) \tilde{\phi}_m(\mathbf{r}) d\mathbf{r} \,. \tag{2.23}$$

To make Eq. 2.18 hold, it must apply:

$$\int \tilde{p}_i^{\mathbf{R}}(\mathbf{r}) \tilde{\varphi}_j^{\mathbf{R}}(\mathbf{r}) d\mathbf{r} = \delta_{i,j} \,. \tag{2.24}$$

Finally the wave function is composed of three different components:

$$\phi_m(\mathbf{r}) = \underbrace{\tilde{\phi}_m(\mathbf{r})}_{1} + \sum_{\mathbf{R}} \sum_i (\underbrace{\varphi_i^{\mathbf{R}}(\mathbf{r})}_{2} - \underbrace{\tilde{\varphi}_i^{\mathbf{R}}(\mathbf{r})}_{3}) \cdot a_i^{m,\mathbf{R}} \,. \tag{2.25}$$

Term 1 is the pseudo-wave function of Eq. 2.14, which describes the smooth behavior of the true wave function correctly in regions further away from atomic nuclei. The partial wave expansions 2 describe the oscillations of the true wave functions close to the nuclei, within the augmentation spheres. The pseudo partial wave expansions 3 are effectively correction terms. They are identical to the contribution of 1 inside the augmentation region and equal to their counterpart in 2 outside their mutual augmentation region. They therefore cancel out the inadequate terms in each region, and assure the transition between the terms that properly describe the true wave function.

The description of the core states in PAW methods usually follows the frozen core approximation, although the method can, in principle, be modified to dismiss it [113]. For practical aspects on the implementation of PAW-method, the reader is referred to the literature already mentioned above [113–115]. The PAW data sets of VASP used in this work were generated by Kresse and Joubert [108].

2.2 Structural models of crystalline solids

A crucial part to every simulation is the design of the model system. One of the big advantages of the *ab initio* methods is, that they explicitly consider the very basic principles of physics and chemistry, and therefore only very few assumptions and simplifications are needed to describe the system behavior accurately. Moreover, the largest part of this task is already covered by the steps described in Section 2.1 and by choosing density functional

theory to solve the quantum-mechanical problem. But since the computational cost of the method is very high, the size of systems that can be treated with it is rather limited. This fact makes it necessary to find a seizable representation of setups that do not exceed the available resources. The following section describes some common techniques used for DFT calculations of crystalline solids and the the difficulties they impose. Also modeling approaches adapted in this work to overcome these difficulties and to give an adequate representation of the alloy-systems treated are explained.

2.2.1   Supercell approach and Bloch's theorem

Like any other differential equation, also the Kohn-Sham problem needs boundary conditions to be well defined. The most commonly applied ones within the density functional theory are periodic boundary conditions (PBCs). For crystalline materials, which have per definition recurrently arranged components, the periodic continuation of the model does reflect the reality extremely well. Thus, the PBCs make it possible to simulate bulk materials by considering only a handful of atoms that lead to the same result as millions of atoms that form up a real solid [94]. The periodically repeated simulation cell, called supercell, needs to be chosen sufficiently large to adequately reflect the studied material within the PBCs framework, but kept as small as possible to reduce computational costs. This approach has some important consequences to the solution of the quantum mechanical problem and the methods used to attain it.

From the idealized problem of a single electron in a potential well, it is known that the allowed $\mathbf{k}$-vectors of the wave function depend on the dimensions of the well:

$$ k_x = \frac{2\pi n_x}{L_x}, \quad k_y = \frac{2\pi n_y}{L_y}, \quad k_z = \frac{2\pi n_z}{L_z}, \qquad n_{x,y,z} = 0, \pm 1, \pm 2, \dots . \qquad (2.26) $$

For macroscopic crystals, the effective dimensions of the system L_x, L_y and L_z become extremely large and for the repeated arrangement of the supercell in all directions infinite. Thus, the allowed $\mathbf{k}$-vectors become quasi-continuous. In the discrete case, the electron density, the total energy and other properties of a system are calculated as a sum of the respective values over all $\mathbf{k}$-vectors. In the limit of quasi-continuous $\mathbf{k}$-vectors, this sum becomes an integral [116].

Moreover, Felix Bloch showed in 1928 that the wave functions of electrons in a periodic potential of a crystal lattice:

$$ V_{ext}(\mathbf{r}) = V_{ext}(\mathbf{r} + \mathbf{R}) , \qquad (2.27) $$

where $\mathbf{R}$ is a crystal lattice vector, can be described as a periodic modulation of plane waves [116, 117]. They can therefore be expressed as a product of a plane wave and a function $u(\mathbf{r})$ with the same periodicity as the lattice [95, 96]:

$$ \psi_{\mathbf{k}}^{n}(\mathbf{r}) = u_{\mathbf{k}}^{n}(\mathbf{r})e^{i\mathbf{k}\cdot\mathbf{r}} . \qquad (2.28) $$

The same applies to the Kohn-Sham orbitals $\phi_{\mathbf{k}}^n(\mathbf{r})$ [96]. $\mathbf{k}$ in Eq. 2.28 is a wave vector spanning the reciprocal space, where $\phi_{\mathbf{k}}^n(\mathbf{r})$ solves the eigenvalue problem Eq. 2.15 and is therefore one part of the index m in Eq. 2.14.

Bloch's theorem reduces the unknown part of $\phi_{\mathbf{k}}^n(\mathbf{r})$ (or $\psi_{\mathbf{k}}^n(\mathbf{r})$) to a periodic function $u_{\mathbf{k}}^n(\mathbf{r})$. The expansion in the basis set of Eq. 2.14 is now limited to this function [95]:

$$u_{\mathbf{k}}^n(\mathbf{r}) = \sum_{p=1}^{P} c_p^{n,\mathbf{k}} \phi_p^{\text{basis}}(\mathbf{r}) \,. \tag{2.29}$$

Because of the periodic nature of $u_{\mathbf{k}}^n(\mathbf{r})$, a plane wave basis set is a very convenient choice, as it makes the expansion 2.29 equivalent to a Fourier series.

Solving the eigenvalue problem Eq. 2.15 with this ansatz for a given $\mathbf{k}$ yields P eigenvalues, $\varepsilon_{\mathbf{k}}^n$, and P sets of coefficients, $c_p^{n,\mathbf{k}}$, defining the corresponding eigenvectors. This P solutions differ in general in energy and are distinguished by the so-called band index n, the second part of the original index m in 2.15 [95].

As mentioned above, getting the system properties requires solving Eq. 2.15 continuously in the $\mathbf{k}$-space and calculating the integral, which is computationally not possible. Fortunately $\phi_{\mathbf{k}}^n(\mathbf{r})$ and $\varepsilon_{\mathbf{k}}^n$ usually vary smoothly in the $\mathbf{k}$-space. Therefore, the system properties can be found by solving the Kohn-Sham equation on a discrete grid of $\mathbf{k}$-points, and the integral is determined numerically as a weighted sum over those points. The density of this grid must be high enough to capture the $\mathbf{k}$-dependence of the solutions, and is thus another factor influencing the accuracy of the calculations.

2.2.2 Special quasi-random structures

Many material properties sensitively depend on a specific arrangement of its constituting elements on the atomic scale. When doing *ab initio* calculations for disordered systems, it is therefore important to carefully consider how to occupy the limited number of sites available in the supercell to fully capture the features of electronic interactions in a real material. There are two principle approaches to tackle this problem. The so-called non-structural theories rely on "average occupations" of the atomic sites in the cell to reproduce these interactions [87, 118]. In contrast to them, structural methods approach the issue by considering specific atom configurations. In order to deduce a physical property from the calculations of a certain system state, the ensemble average of that property over all possible configurations of that state has to be used. If Ω is the set of all possible configurations of the state m, the total energy, for example, is obtained as:

$$E = \langle E \rangle = \sum_{\sigma \in \Omega_m} \rho(\sigma) E(\sigma) \,, \tag{2.30}$$

with $\rho(\sigma)$ being the probability of the system being in configuration σ [87]. The problem with this procedure is, that usually it is impossible to do calculations on all possible configurations.

The so-called special quasi-random structures (SQS) approach, proposed by A. Zunger *et al.*, tries to overcome this obstacle by constructing a single supercell that is representative for a given state [118, 119]. By introducing geometrical correlation functions, the conditions of an average neighborhood for a given arrangement can be quantified. These correlation functions correspond to different "figures" (pairs, triplets, etc.) into which the vicinity of a site can be discretized. Thereby, the configuration of a supercell can be compared to a statistical ensemble. Instead of evaluating the properties of a system by statistical methods, special structures can be designed which reproduce the ensemble characteristics as good as possible [118]. If the considered neighborhood interactions are kept confined to pairs, the assessment procedure of the atom configuration can be realized by using Warren-Cowley short-range order parameters as shown in the following adopted after Holec *et al* [87, 120]. In a supercell of a binary alloy of A and B atoms with N atom sites, the number of A-B pairs separated by the jth distance l^j is referred to as N_{AB}^j. Every site in the cell has M^j neighboring sites at distance j. The composition of the alloy is given by the concentrations x_A and $x_B = 1 - x_A$. In an arrangement with statistically random occupation, $x_A N$ sites are populated by A-atoms and every A atom is neighbored by an average of $x_B M^j$ B-atoms. The total number of A-B pairs in the random alloy is therefore $x_A N x_B M^j$. The Warren-Cowley short-range order parameter is defined as [87]:

$$\alpha^j = 1 - \frac{N_{AB}^j}{x_A x_B N M^j} \, . \tag{2.31}$$

It is a measure of the ordering characteristics of the supercell. For a statistically random arrangement on distance l^j, $\alpha^j = 0$, while a trend to ordering results in $\alpha^j > 0$, and clustering produces $\alpha^j < 0$ [87].

If the alloy is altered to three components A, B, and C, one short-range order parameter for each type of mixed pairs A-B, B-C and C-A is needed to evaluate the supercell configuration. In analogy to Eq. 2.31, they are expressed for as:

$$\begin{aligned}
\alpha_{AB}^j &= 1 - \frac{N_{AB}^j}{x_A x_B N M^j} \, , \\
\alpha_{BC}^j &= 1 - \frac{N_{BC}^j}{x_B x_C N M^j} \, , \\
\alpha_{CA}^j &= 1 - \frac{N_{CA}^j}{x_C x_A N M^j} \, ,
\end{aligned} \tag{2.32}$$

with N_{AB}^j, N_{BC}^j, and N_{CA}^j being the respective number of neighbor-pairs separated by a distance l^j [87].

The cases treated in this thesis, are statistically random solid solutions. Therefore, all SROs should ideally be 0. Unfortunately, this is not achievable for all combinations of supercell sizes and concentrations. Instead the weighted sum of all $|\alpha^j|$ was minimized. A more detailed account on the supercell creation procedure is given in the next section.

2.2.3 System setup

In both of the treated alloy systems, Ta-Al-N and Nb-Al-N, the metallic components constitute a solid solution with statistically random arrangement on one sublattice while the second sublattice is fully populated by nitrogen atoms. Supercells with dimensions listed in table 2.1 were used to model the two pseudo-binary alloys. To capture the system behavior

Table 2.1: Dimensions of the supercells used for the calculations in this work.

structure	supercell size	max. nr. of atoms
	Ta-Al-N	
B1	$3 \times 3 \times 3$	64
B4	$3 \times 3 \times 2$	72
B_h	$3 \times 3 \times 3$	54
Bk	$3 \times 3 \times 2$	72
SG189	$2 \times 2 \times 3$	72
	Nb-Al-N	
B1	$3 \times 3 \times 3$	64
B_i	$3 \times 3 \times 1$	72
B4	$3 \times 3 \times 2$	72
Bk	$3 \times 3 \times 2$	72

as a function of the aluminum fractions x on the metal sublattice, calculations for different metallic compositions were conducted. The aimed compositional step was $\Delta x = 0.125$. Due to the limited atom numbers in the supercells, the specific concentrations of the different setups deviate slightly from this rule, and thus the discrete compositions were distributed around the individual target values.

The investigation of the influence of vacancies requires construction of supercells containing defects. For this purpose, supercells with 1,2,4, and 6 vacancies on either the metal or the nitrogen sublattice were created for every crystal structure, again aiming for the same values in aluminum concentrations as in the perfect cases. The figures 2.2 and 2.3 give an overview of the configurations for which calculations were performed. For the perfect compositions without any vacancies, the atom configurations on the metal sublattice were optimized following the SQS-approach for binary alloys using pair-figures (Eq. 2.31) up to the 7th coordination shell. The structure generation was done using a self written python script (a more general version of which for ternary systems can be found in Appendix A) and was carried out as follows: In the first step, parameters controlling the creation procedure were read in from several input files, including the supercell dimensions and the lattice sites. Sticking to the notation of Section 2.2.2 and Eq. 2.31, these parameters are the maximum number J of distances j to be considered for the pair figures, and a tolerance range l_{tol} for the distances. All neighbors at distances $l^j \pm l_{\text{tol}}$ are attributed to distance l^j, i.e., the jth coordination shell of each site. Next are the number of atoms of type A in the supercell, and
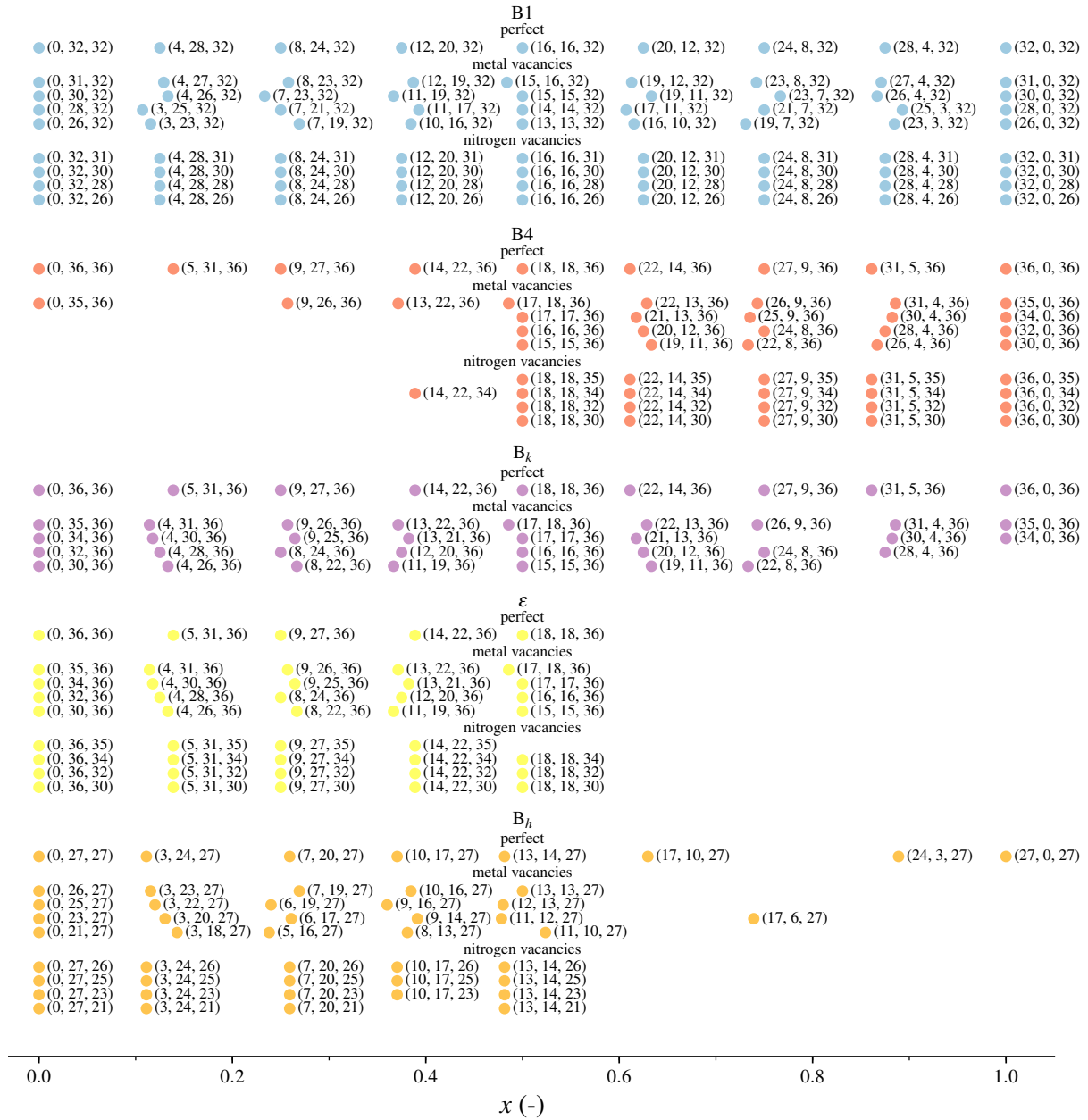
Figure 2.2: Supercell configurations of the various phases over aluminum concentration x in the Ta-Al-N system. The parentheses give the number of atoms in the supercell (Al, Ta, N).

how many different SQS-supercells should be generated. The last parameter determines the maximum number of configurations produced and evaluated. After reading the input information, and evaluating some parameters needed for the further process, the script produces a supercell with randomly arranged atoms on the available sites. Subsequently the binary Warren-Crowley SROs of the cell are evaluated according to Eq. 2.31 for all distances l^j taking into account the periodic boundary conditions. Finally both the supercell setup and its SROs are stored. This is repeated until the maximum number of configurations defined
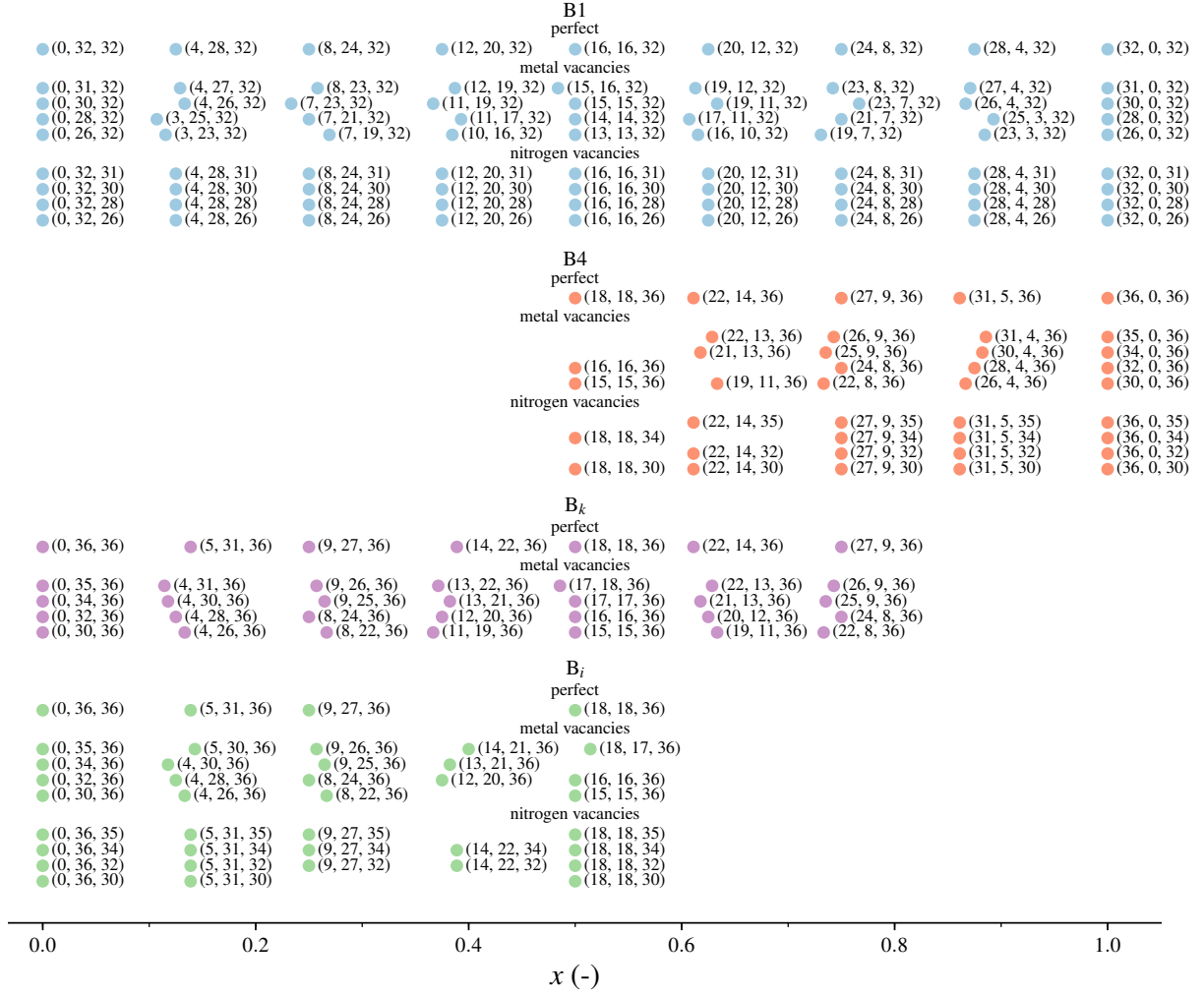
Figure 2.3: Supercell configurations of the various phases over aluminum concentration x in the Nb-Al-N system. The number of atoms in the supercell is given in the parentheses (Al, Nb, N).

in the input files is reached.

The generated supercells are then compared using the weighted sum of the SROs:

$$\alpha_{\text{total}} = \sum_{j=1}^{J=7} w^j \left| \alpha^j \right|, \tag{2.33}$$

with each weight w^j depending on the coordination distance nummer j and the maximum number of distances to be considered J:

$$w^j = 1 + 2 \frac{J-j}{J-1}. \tag{2.34}$$

Since ideally all SROs α^j should be zero for a totally random configuration, the supercell with the lowest α_{total} is the most suitable model of a solid solution for the given composition, and is therefore used for the *ab initio* calculations.

As already pointed out, this procedure was used for all perfect structures to determine the arrangement of the different metal atoms on the corresponding sublattice. The very same idea was applied to choose the positions of the vacancies on the nitrogen sublattice for the various structures, treating the point defects like an alloying element.

To create supercells with defects on the metal sublattice, the procedure was adapted for ternary alloys. Again the population of the supercell sites is randomly generated and their SROs are evaluated according to Eq. 2.32. The criteria for the overall assessment of the cells is now the sum of the weighted sums of Eq. 2.33:

$$\alpha_{\text{total}} = \sum_i \sum_{j=1}^{J=7} w^j |\alpha_i^j| , \qquad i = AB, BC, CA . \tag{2.35}$$

Usually, more than one equally good configurations could be found, in which case the eventually used one was chosen randomly from the best identified structures. The script used to create, assess and select the supercells for ternary compositions is attached in Appendix A. It should be noted that the quantification procedure of the supercells using SROs described above does not consider any possible influence of higher order geometrical correlations, and therefore possible effects resulting from them cannot be reproduced by calculations using such supercells. Also the random search applied does not guarantee to produce the overall optimum supercell for the SRO-approach. Finally, it is noted that the SRO-based generation described above is a purely geometrical procedure. A more physical approach would consider weights w^j based on the system's chemistry.

Computational Setup

This section gives insight into the technicalities of the calculations. Section 3.1 contains remarks on VASP and deals with some of the many parameters it offers to control numerical precision and algorithmic details of the computations. The performed convergence tests and the problems arising from them are discussed in section 3.2. Section 3.3 addresses these problems and depicts the way chosen to resort them as well as the findings which justify this alternative approach.

3.1 Software and numerical parameters

As mentioned earlier, the calculations presented in this work were carried out using the Vienna Ab-Initio Simulation Package (VASP) in version 5.12. VASP bases on a program created by Mike Payne at the MIT and was developed by Georg Kresse, Jürgen Hafner and Jürgen Furthmüller. It is a commercial program for density functional theory and *ab initio* molecular dynamics computations which uses a plane wave basis set together with pseudo-potentials and the projector augmented wave method [108, 121–123]. While the main parameters determining the accuracy of these methods were already described in sections 2.1 and 2.2, this section discusses how they can be controlled in the context of VASP and it is therefore mainly based on the official user guide [121], the workshop lectures [124] and chapters 6 and 7 of Lee's book [94].

VASP, however, is a rather extensive software package, which allows its users to adjust many more settings which in turn influence the calculated predictions. A detailed account of all of them can be found in the official user guide [121].

3.1.1 k-Points

In Sections 2.1.5 and 2.2.1 it was pointed out, that a solution of the Kohn-Sham eigenvalue-problem depends on the wave vector $\mathbf{k}$ of the reciprocal space. In crystalline solids the

allowed **k**-vectors are quasi-continuous, and their unique values span the first Brillouin zone. To capture the **k**-dependence of the solution, the reciprocal space has to be discretized using a mesh of fixed **k**-points. This so-called **k**-point sampling is important, since it directly influences the accuracy of a numerical integration needed for estimation of various system properties.

In VASP, the **k**-mesh generation can be done either by hand or automatically. The process is controlled through an input file called KPOINTS. For the present calculations, the fully automatic **k**-point sampling method was chosen. This method produces a Γ-centered Monkhorst-Pack grid, the density of which is defined by a parameter l in the KPOINTS file [125]. l controls the number of subdivisions N of the Brillouin zone along the three reciprocal vectors $\mathbf{b}_1$, $\mathbf{b}_2$ and $\mathbf{b}_3$ according to:

$$N_i = \max(1, l \cdot |\mathbf{b}_i| + 0.5), \qquad i = 1, 2, 3. \tag{3.1}$$

The corresponding mesh consists of **k**-points with coordinates:

$$\mathbf{k} = \sum_{i=1,2,3} \frac{n_i}{N_i} \mathbf{b}_i, \qquad n_i = 0, \dots, N_i - 1. \tag{3.2}$$

The value of the parameter l has to be determined by a convergence test for a representative quantity, typically the total energy [121].

3.1.2   Basis Set

The importance of the type and size of the basis set used to construct the KS-orbitals was discussed in Section 2.1.5. VASP uses a plane wave basis set. Thus, the only remaining parameter related to this topic concerns the size P in Eq. 2.14.

Using Bloch's theorem (Eq. 2.28) and a plane wave basis set, Eq. 2.14 is written as [95]:

$$\tilde{\phi}_{\mathbf{k}}^n(\mathbf{r}) = \sum_{\mathbf{K}} c_{\mathbf{K}}^{n,\mathbf{k}} e^{i(\mathbf{k}+\mathbf{K})\cdot\mathbf{r}}. \tag{3.3}$$

As mentioned in Section 2.2.1, index m of a KS-orbital in Eq. 2.14 includes a vector **k** and a band index n, with **k** being a point of the **k**-mesh, at which the orbital $\tilde{\phi}_{\mathbf{k}}^n(\mathbf{r})$ is an eigenstate, and n labels the solutions at the same **k**. The effectively used basis functions are therefore the plane waves:

$$\phi_{\mathbf{K}}^{\text{basis}}(\mathbf{r}) = e^{i(\mathbf{k}+\mathbf{K})\cdot\mathbf{r}}. \tag{3.4}$$

The size P of the basis set is controlled by limiting the maximum kinetic energy of the plane waves. With a so-called plane wave cut-off energy, E_{cut}, the basis consists of plane waves (Eq. 3.4) satisfying a condition [121]:

$$|\mathbf{k} + \mathbf{K}| < G_{\text{cut}} \qquad \text{with} \qquad E_{\text{cut}} = \frac{\hbar^2 G_{\text{cut}}^2}{2m_e}. \tag{3.5}$$

Consequently, the basis set is different at every $\mathbf{k}$-point. E_{cut} is usually set by the user in an input file called INCAR using the so-called ENCUT-tag.

Similarly to the $\mathbf{k}$-mesh density, the cut-off energy has to be determined by convergence tests. More details on the convergence tests for both, $\mathbf{k}$-points and energy cut-off are given in Section 3.2.

3.1.3 Additional Settings

This section shortly describes some other parameters of VASP that can be controlled by the user through the INCAR file, and which influence the course of the calculations. Giving a full account on those parameters goes beyond the scope of this work and often even the user guide, from which most of the following information originates [121].

PREC

PREC is a master parameter controlling the overall accuracy and computational demands by setting automatically energy cut-off, FFT-grid and real space projector optimization accuracy, unless they are explicitly defined by the user in the INCAR file. While the energy cut-off has already been discussed in Section 3.1.2, the other two matters will be shortly addressed in the following. In this work, PREC was set to *Medium*. Other available options are *Low*, *Hight*, *Single*, *Normal* and *Accurate* and their exact effects on the calculations can be found in the user guide [121].

Since in all calculations concerning this work the ENCUT tags were set manually, the basis set was not affected by the PREC tag.

In every cycle of the electronic loop, the individual terms of the KS-Hamiltonian in Eq. 2.9 have to be evaluated using the actual charge density. While some of the terms, such as the Coulomb interactions between the electrons or the kinetic energy, are easier to calculate in the reciprocal space, others such as the exchange-correlation term and the external potentials are evaluated in real space. Hence, the electron density $n(\mathbf{r})$ has to be accessible in both spaces, i.e., it must be Fourier transformed, which is done using fast Fourier transformation (FFT). However, since $n(\mathbf{r})$ is calculated as a sum of the squared KS-orbitals (Eq. 2.10), to be complete this transformation requires even more plane waves than included in the basis set for expansion of the orbitals (Eq. 3.3) [94]. To be exact, the transformation must include all plane waves with wave vectors up to $2 \cdot G_{\mathrm{cut}}$. For this reason, an additional FFT-grid is introduced, defining the plane waves used for the transformation. The number of the grid points is controlled manually by the parameters NGX, NGY and NGZ. Their default values are determined by the chosen PREC-tag. For PREC=*Medium*, the FFT uses plane waves up to $\frac{3}{2}G_{\mathrm{cut}}$ introducing so called wrap-around errors [121].

For pseudo-potentials which require augmentation charges, another FFT-grid is used to

expand them, which has even more grid points than the one transforming the charge density. Also here, the number of the grid points can be controlled manually via the tags NGXF, NGFY and NGFZ, or automatically using the PREC-tag. This FFT-basis is defined in terms of a cut-off value G_{aug} related to a parameter ENAUG according to:

$$\frac{\hbar^2}{2m_e}G_{\text{aug}}^2 = \text{ENAUG}\,, \tag{3.6}$$

where ENAUG is usually read from the used pseudo-potential input files, but can also be specified manually. For PREC=$Medium$, the FFT-basis uses all plane waves with wave vectors up to $4 \cdot G_{\text{aug}}$ [121].

The last parameter influenced by the PREC-tag is the so-called ROPT-value. The impact of the pseudo-potentials on an electron, described in section 2.1.6, depends on the angular momentum of its wave function. This so-called non-locality of the pseudo-potentials make it necessary to project the orbitals onto different angular momentum states, so that the respective parts of the pseudo-potential can act on each of them [110, 126]. To do so, special projector operators have to be evaluated, which can be either done in real or in reciprocal space [121]. For large systems, it is more efficient to perform this projection in the real space, but in this case the projector operators have to be optimized to prevent errors [121, 126]. The ROPT parameter can be used to influence the accuracy of such optimization. For the calculations presented here, the projection was carried out in the real space and the optimization scheme used is proprietary to VASP (LREAL=Auto) [121]. In the context of PREC=$Medium$ and this optimization scheme, the parameter ROPT is set to a value of $2 \cdot 10^{-3}$, meaning that the *real-space operators will be optimized for an accuracy of approximately* $2 \cdot 10^{-3}\,\text{eV/atom}$ [121].

ALGO

The ALGO-tag determines the algorithm used for the iterative solution of the KS-equations 2.8 in the electronic optimization loop. It determines the numerical details on how the eigenvalue problem is preconditioned and subsequently solved. Despite their internal differences, all of the algorithms implemented in VASP minimize the same residual vector, using the same subroutines. They therefore should yield the same result and only differ in their convergence behavior, although it the outcome can actually differ due to numerics [121].

Within this work, the ALGO-tags *Fast* and *Normal* were used. When ALGO is set to *Fast*, the first few cycles of the electronic loop use the blocked Davidson (DAV) algorithm to solve the KS-equations [127, 128]. DAV is always stable and is therefore more suitable for orbitals far from the solution. It is thus used at the beginning of the loop, to pre-converge the orbitals [121, 129]. In the subsequent cycles, a variant of the *"residual minimization scheme, direct inversion in the iterative subspace"* (RMM-DIIS) is used [123, 130, 131]. This algorithm is up to two times faster than DAV but its convergence is sensitive to the initial guess

[121, 129]. The default of ALGO, *Normal*, uses of the blocked Davidson iteration scheme for all electronic updates within the loop.

For the calculations of the Ta-Al-N system, ALGO=*Fast* was applied. In the Nb-Al-N system, however, the RMM-DIIS algorithm repeatedly failed, which led to longer overall calculation times. To prevent this, many of the cases in this system were treated using only the Davidson algorithm by setting ALGO=*Normal*.

IBRION and ISIF

Sections 2.1.1 to 2.1.5 introduced the Born-Oppenheimer theorem, implying that the KS-equations are solved for fixed positions of the nuclei by a minimization of the total energy with respect to the electron density. In order to find the absolute minimum of a system, the total energy has to be optimized also with respect to the nuclear positions. This involves repeatedly solving the electronic problem, extracting forces acting upon the atomic nuclei, and updating their positions according to them (see Fig. 3.1). The forces are attained using the Hellmann-Feynman theorem, by calculating the derivative of the total energy with respect to the ion positions [121, 132, 133]. Based on these forces the positions of the nuclei can be subsequently updated in different ways.

In VASP, there are three algorithms implemented for the ionic optimization procedure in the context of the pure density functional theory, namely, a quasi-Newton-Raphson algorithm originally suggested by Peter Pulay (RMM-DIIS, a variant of which is also used for the electronic minimization), a conjugate-gradient algorithm, and a damped molecular dynamics algorithm [121, 123, 130, 131]. A particular method is chosen using the IBRION-tag in the INCAR file. For all calculations performed for this work, the conjugate gradient method was used which corresponds to IBRION=2.

The ISIF-tag restricts the nuclear movement and the degrees of freedom involved in the optimization process. Depending on the purpose of a calculation, the ionic relaxation is conducted allowing additionally for a change of the supercell's shape and/or volume [121]. Since the main objective of this work is the investigation of the phase stability in the Ta-Al-N and the Nb-Al-N systems, the goal of the calculations is to fully optimize the ion positions, which is achieved by setting the ISIF-tag to 3.

EDIFF and EDIFFG

As mentioned in the previous subsection a DFT-calculation usually includes two optimization loops. On the one hand, the inner cycle determines the self consistent solution of the KS-eigenvalue problem sketched in Section 2.1.3, and leads to the electron density that gives the lowest total energy for a set of given fixed ion positions. The outer loop, on the other hand, involves the relaxation of the ion positions by calculating forces on the nuclei from the

charge density and updating their positions according to the algorithm used. The overall
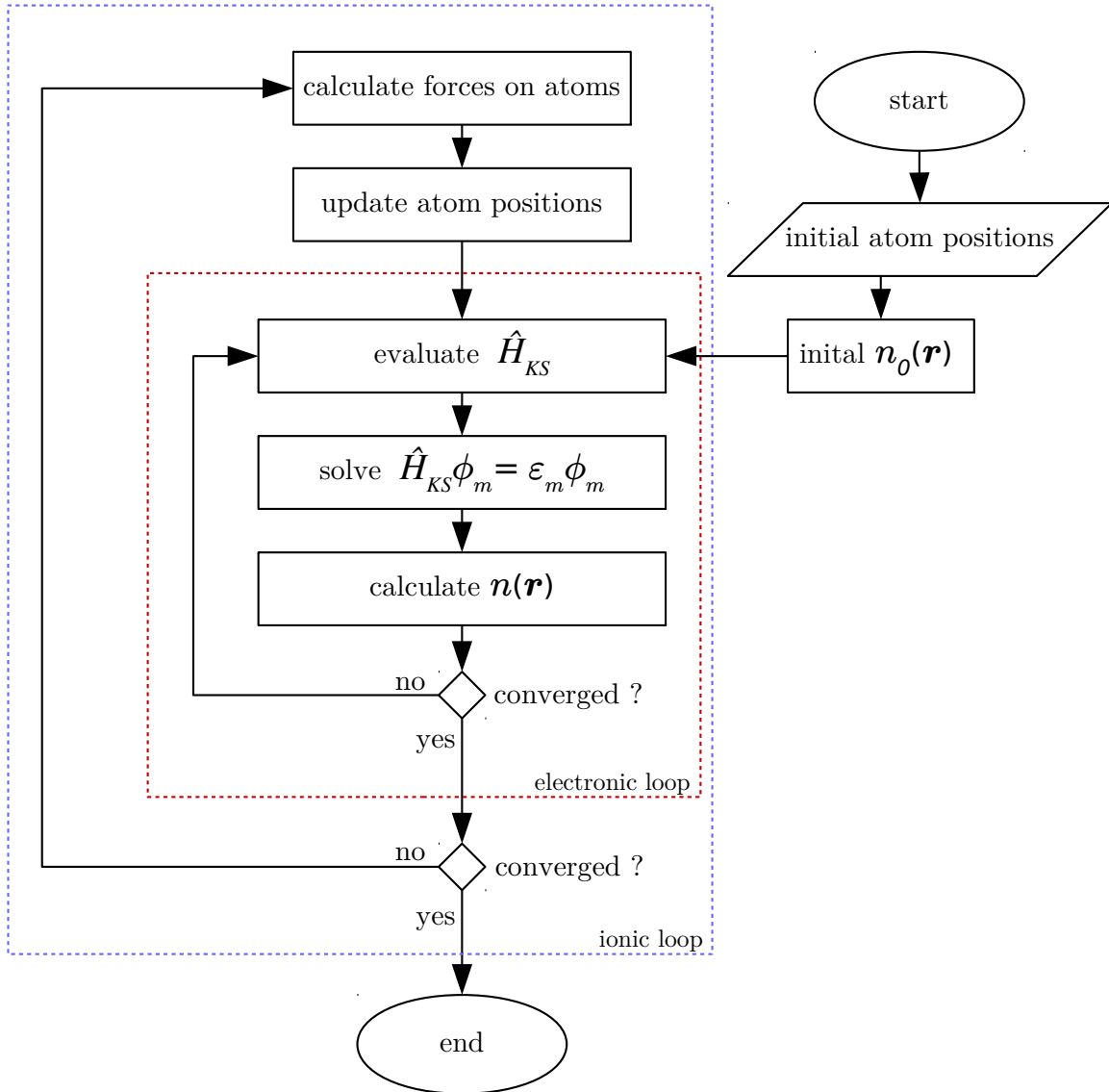

Figure 3.1: Flowchart of the overall relaxation process.

flow of these actions is visualized in Fig. 3.1. Both loops are executed until convergence is reached. Whether a loop has converged or not is assessed by break conditions.

VASP allows the user to manipulate these conditions within certain limits. For the electronic system, the stop criterion of the self consistent optimization involves the change in total energy in two succeeding steps. If this change lies below a certain threshold value, convergence is reached and the loop is left [121]. This is controlled by the EDIFF-tag in the INCAR file, with a default value of 10^{-4} eV.

On the contrary there are two options for the stopping condition of the ionic relaxation loop (EDIFFG tag in the INCAR file). It can either also be that the change of the total energy

in two consecutive steps is smaller than a critical value, or that the absolute value of all forces on each individual nucleus in the system is below a certain limit. By default, the first condition is applied and the total energy change has to be less than ten times the stopping condition of the electronic loop (EDIFFG= 10·EDIFF).

In all of the calculations presented here, the electronic break condition of EDIFF= 10^{-5} was used. The ionic relaxation was stopped either at EDIFFG= 10^{-3} eV or EDIFFG= -10^{-1} eV/Å.

It is noted at this point, that in the context of the other settings applied to the computations, the total energies calculated with EDIFFG= 10^{-3} eV and EDIFFG= -10^{-1} eV for an otherwise identical calculation setup differed less than 0.0011 eV/atom. The differences arising from those two criteria are therefore negligible.

ISMEAR and SIGMA

In the ground state at 0 K, the energy bands of solids are filled with electrons only up to the Fermi energy level, E_F, and no states are occupied for higher energies [116]. The band occupation drops sharply within partly filled bands at the Fermi level. This fact, however, imposes a problem for the numerical integration over the reciprocal space, because to capture a sudden change of the occupation, a very dense mesh of **k**-points is needed [94, 121]. Unfortunately, a higher density leads to much longer calculation times. To limit the necessary increase in number of **k**-points, so-called smearing methods are applied, which replace the exact sharp step function with some other, more smoothly decaying function. In the Fermi-smearing method, for example, the Fermi-Dirac function is used, resulting in an occupation profile found for finite temperatures [116, 121, 134]. Though they do qualitatively the same, other methods do not have such underlying physical pictures, and are simply a tool to speed up the calculations [121].

For the calculations of this work, the method of Methfessel and Paxton was applied. It replaces the step function with the error function plus a set of Hermite polynomials multiplied with e^{-x^2} [135]. The smearing methods are controlled by the ISMEAR-tag in the INCAR file. The Methfessel and Paxton method is applied when ISMEAR=N for $N > 0$, where the parameter N determines the number of Hermite polynomials employed for the approximation of the step function. The used parameter was ISMEAR=1 and consequently polynomials up to the first order were included in the expansion [121]. The width of the smearing procedure is defined using the SIGMA-tag which was set to 0.1 eV.

3.2 Convergence tests

To identify the cut-off energy and **k**-mesh density needed for the desired accuracy of our calculations, convergence tests with respect to these parameters have to be carried out. The

general practice is to perform multiple static calculations with increasing cut-off energies and **k**-mesh densities, while leaving all other input parameters constant. As soon as the properties of interest change only within a desired uncertainty range upon further increasing the cut-off energy and/or the **k**-mesh density, the system can be considered converged.

Typically the total energy per atom is chosen for the convergence tests in DFT studies focusing on structural properties with a threshold value of $1\,\mathrm{meV/atom}$. Table 3.1 lists the structures included in this work, together with the ENCUT values and the values of the parameter l of the KPOINTS file for which this convergence criterion is met. The convergence tests were carried out with PREC=*Accurate*. Based on these tests the proper parameters for

Table 3.1: Values for plane wave cut-off energy E_{cut} (basis set size), and l (**k**-mesh density) resulting from the convergence tests.

	structure	E_{cut} (eV)	l (-)
AlN	B1	700	49
	B4	700	33
	B_h	650	57
	B_i	700	25
	Bk	700	33
	SG189	650	49
TaN	B1	650	73
	B4	700	57
	B_h	650	41
	Bk	700	65
	SG189	700	65
NbN	B1	650	65
	B4	700	57
	B_i	650	57
	Bk	700	65

the calculations would be at least a cut-off energy of $700\,\mathrm{eV}$ and a parameter l of 73 for the automatic **k**-mesh generation. However, when running the calculations with these settings on the Vienna Scientific Cluster 2 (VSC2), several problem arose.

The first problem encountered was that the computations of solid solutions without crystal symmetries had memory demands that exceeded the limit of a single standard node of the VSC2. While increasing virtual memory failed and even led to a crash of the used nodes, the use of special nodes with higher memory succeeded in some cases. Yet the number of these high-memory nodes is small resulting in very long queuing times, which in turn drastically limited their use due to the large number of calculations needed for this work. Several attempts to parallelize the processes using more than one standard node resulted in

an increase of the time needed for the individual calculations to finish. The reason for this is most likely the use of non-optimized parallelization parameters. VASP gives the user control over a number of settings affecting the parallelization of the individual steps and, therefore, the communication between the used processor cores. The efficiency of the parallelization strongly depends on the adjustment of the settings used to the hardware and requires intensive testing [121]. Our attempts to overcome the problems of the calculations' high memory demands using parallelization were not successful, as they lead to an unacceptable consumption of the computational resources, which is why this approach was finally abandoned.

As an alternative option it was decided to overcome this problem by reducing the numerical accuracy and, thereby, the calculational load. The cut-off energy was finally reduced to a value of 450 eV and the parameter l determining the $\mathbf{k}$-mesh density to 57. To justify these choices and to assess the extend of the errors made, a number of tests were performed which are discussed in the next section.

3.3 Auxiliary accuracy tests

As outlined above, the two problems that had to be solved were the long calculation times needed for relaxing the majority of the structures on the one hand, and the high memory demands on the other hand. Both of these problems are linked to the high values for cut-off energy and $\mathbf{k}$-mesh density needed according to the convergence tests.

To overcome them it was decided to first perform the calculations with lower values for E_{cut} and l and then gradually increase them to sufficiently accurate settings. The results (ion positions and KS-orbitals) of the calculations performed with less accurate numerical parameters were used as inputs for the increased settings of the next step to speed up the convergence. As for some structures and compositions calculations with numerical parameters satisfying the values of the convergence tests could be successfully finished, their results were used to assess the error introduced by the reduced values in E_{cut} and l. The quantity used for this purpose was the energy of formation, E_f, defined in Eq. 4.2 (more details on its physical meaning are given in the next chapter).

In the following *Accurate* refers to the settings $E_{\mathrm{cut}} = 750$ eV, $l = 73$ with PREC set to *Accurate*. *Low* denotes $E_{\mathrm{cut}} = 300$ eV, $l = 40$ and PREC = *Low*, which were the settings used for the first step of the constitutive computations. The results presented as *Medium* were achieved by pre-converging the systems with the *Low* parameters and using the output as a starting point for calculations with $E_{\mathrm{cut}} = 450$ eV, $l = 57$ and PREC = *Medium*.

Figures 3.2 and 3.3 show a comparison between the E_f values of the phases B1, B4 and B$_h$ without vacancies resulting from calculations with the parameter sets *Low*, *Medium* and *Accurate*. While the results from the calculations with the parameters of *Low* differ significantly, the E_f values obtained with the settings of *Medium* are reliable. The difference in E_f

resulting from the calculations with the *Medium* parameters and the *Accurate* settings was at most 0.01 eV/atom. While this magnitude is non-negligible, the error is rather uniform as the *Medium* and *Accurate* E_f-plots are almost parallel. The assessment of the stability of the individual phases, which is based on the relative comparison of the energies of formation, is, however, only weakly affected by this error. Figure 3.4 shows, that the crossover between the phases B1 and B4, and B1 and B$_h$, respectively, is practically the same for the results coming from the *Medium* and *Accurate* calculations. As the main goal of this work is to predict the stability range of the individual phases as functions of aluminum- and vacancy content, the accuracy resulting from the calculations performed with the *Medium* parameters is sufficient.
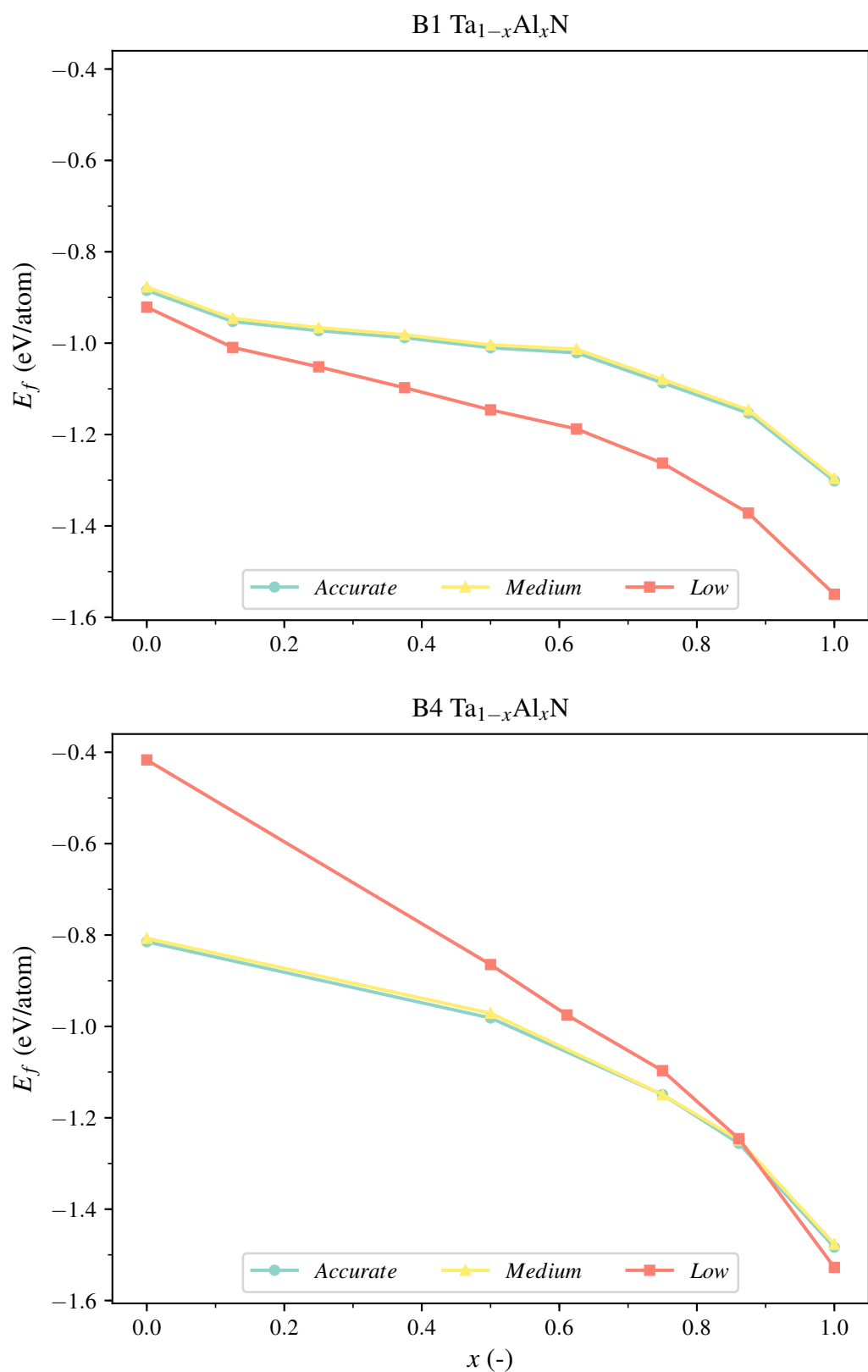
Figure 3.2: E_f against x plots from different numerical settings for the B1 and B4 phases of the $Ta_{1-x}Al_xN$ system without vacancies. The symbols mark the DFT data, while the solid lines are a linear interpolation between the adjacent points.
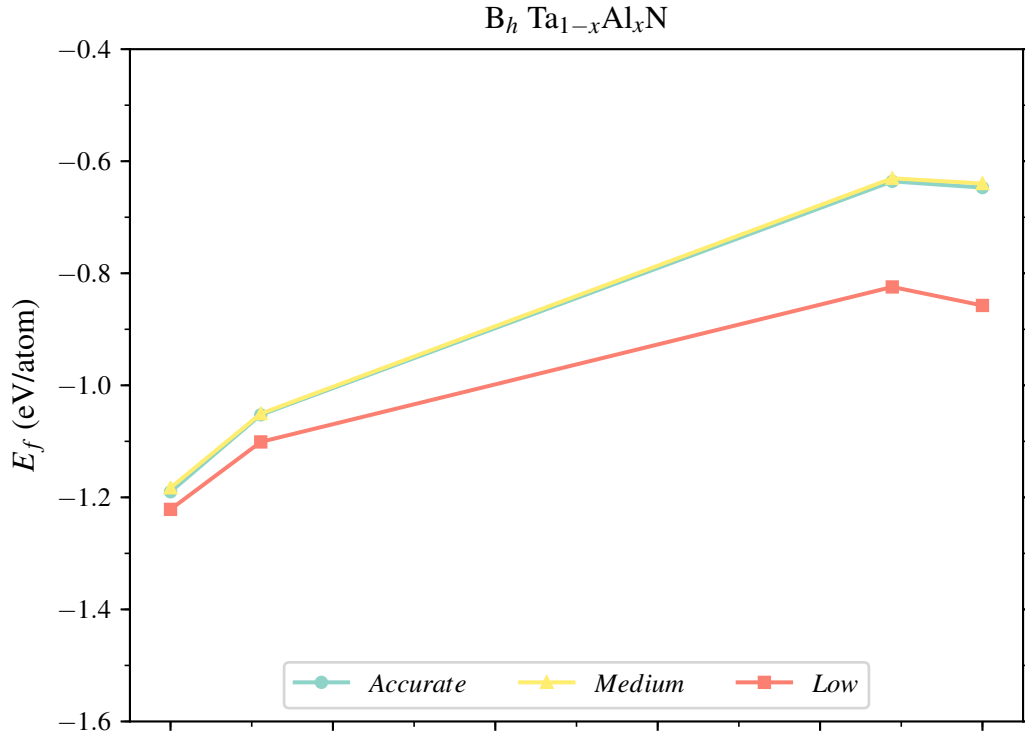
Figure 3.3: E_f data and linear interpolation for different numerical settings for the perfect B_h phase of the $Ta_{1-x}Al_xN$.
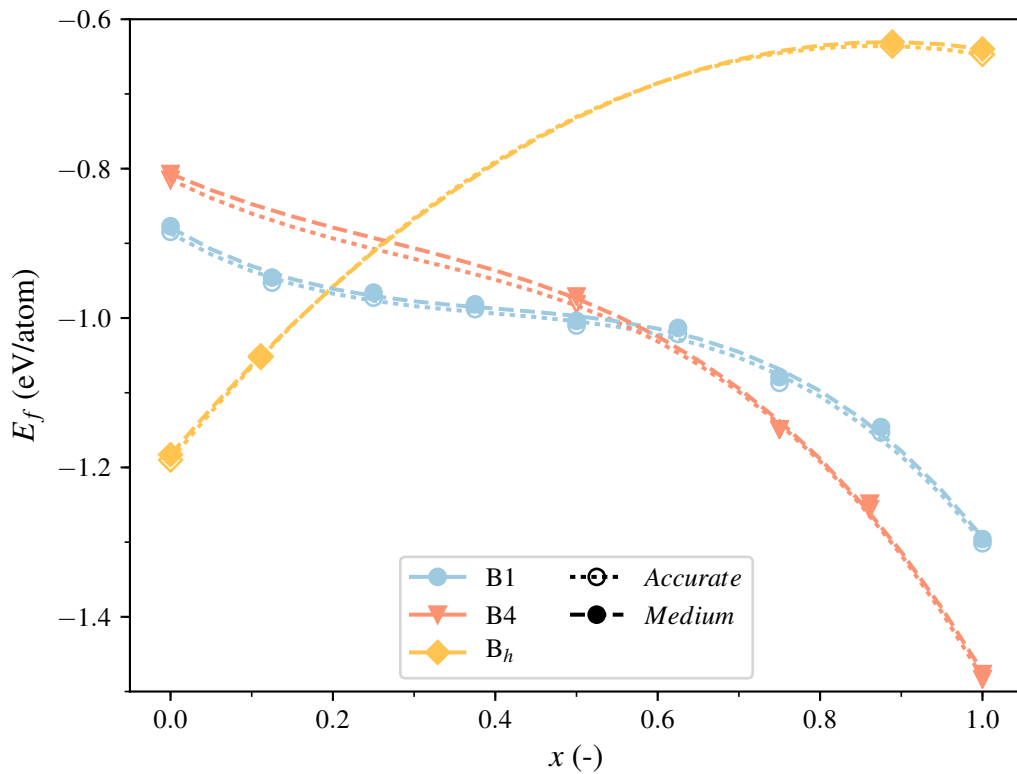

Figure 3.4: Comparison between the E_f against x plots resulting from *Medium* and *Accurate* settings. The symbols mark the DFT data and the dashed and dotted lines are third order polynomials.

Methodology and Results

The first two Sections of this chapter describe quantities used to obtain the actual results which are then presented and discussed in Sections 4.3 and 4.4.

4.1 Energy of formation

The very basic physical principle that all systems try to minimize their overall energy, is useful when discussing relative stability of phases. From classical thermodynamics it is known that the equilibrium state of a system is characterized by a minimum of the Gibbs free energy:

$$G = H - TS = U + pV - TS \,. \tag{4.1}$$

$H = U + pV$ is the enthalpy, U is the internal energy, S the entropy and p, V and T are pressure, volume and temperature, respectively. For the the ground state at $0\,\mathrm{K}$ and no external pressure, G reduces to the internal energy U, which is one of the system properties delivered by the VASP computations. It is important to keep in mind that when using the pseudo-potential method, the internal energy resulting directly from the calculations comprises only the valence electrons not removed by the frozen-core approximation. Consequently, these absolute energy values are hard to discuss on their own, since they can be compared neither with experiment nor with other DFT predictions. Instead, the energy of formation, E_f, is commonly used in DFT studies [2, 28, 87]:

$$E_f = \frac{1}{\sum\limits_i n_i} \cdot \left(E_{\mathrm{tot}} - \sum_i n_i E_i \right) . \tag{4.2}$$

It describes the difference in energy between a system in a particular structure, E_{tot}, and the sum of its pure constituents in their respective equilibrium phases, E_i. In this sense, it is nothing more than the change in internal energy ΔU upon formation of the structure. n_i are the respective numbers of atoms of species i. When one is interested in the stability of

39

competing phases of a system with a certain composition, E_f can therefore be used to rank the different structures. The one yielding the lowest E_f is energetically the most favorable and thus the most stable at $0\,\mathrm{K}$. The reference states and their values, E_i, used for the following considerations are listed in Table 4.1.
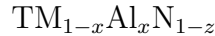
Table 4.1: Reference states and their respective energies, E_i, of species considered in this work.

structure	energy (eV/atom)
fcc Al	-3.697
bcc Ta	-11.735
bcc Nb	-10.195
molecular N_2	-8.339

In the following, the chemical compositions and vacancy concentrations will be described as

$$(\mathrm{TM}_{1-x}\mathrm{Al}_x)_{1-y}\mathrm{N}$$
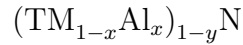
for metal vacancies and

$$\mathrm{TM}_{1-x}\mathrm{Al}_x\mathrm{N}_{1-z}$$

for nitrogen vacancies. TM is a placeholder for either Ta or Nb depending on the system discussed. x denotes the aluminum fraction of metal species, while y and z indicate the vacancy content on the metal and the nitrogen sublattice, respectively.

The E_f values were calculated for the sets of discrete atomic compositions listed in Figures 2.2 and 2.3. To create a phase diagram mapping the stable structures in the space of aluminum fraction, x, and vacancy contents, y and z, a relationship between these parameters and the energy of formation, E_f, is needed for every phase. Thus, functions $E_f(x,y)$ and $E_f(x,z)$ were constructed for each phase by fitting the discrete data points from the calculations in multiple steps using the least squares method.

A function $E_f(x,y)$ has the overall form

$$E_f(x,y) = a_0(y) + a_1(y)x + a_2(y)x^2 + a_3(y)x^3 \,. \tag{4.3}$$

It is therefore a third order polynomial in the aluminum fraction x with coefficients $a_i(y)$ depending on the metal vacancy content, y. The functions $E_f(x,z)$ capturing the influence of nitrogen vacancies are described equivalently to Eq. 4.3 with coefficients $a_i(z)$.

In the first step, the calculated data is used to fit coefficients b_i of several third order polynomials in the aluminum concentration $p(x)$:

$$p(x)\big|_{y=\text{const.}} = b_0 + b_1 x + b_2 x^2 + b_3 x^3 \,. \tag{4.4}$$

For each vacancy concentration $y = $ const. considered in the calculations one polynomial is created.

In the second step, these coefficients b_i are used to determine functions

$$a_i(y) = a_i^0 + a_i^1 y + a_i^2 y^2 \tag{4.5}$$

for every term i in Eq. 4.3. The parameters a_i^0 are set to be identical to the coefficients b_i of the polynomial fitted to the structure data with $y = 0$:

$$a_i^0 = b_i \big|_{y=0} . \tag{4.6}$$

The other two parameters, a_i^1 and a_i^2, are then determined by minimizing the sum of the squared residuals of the function in Eq. 4.5, and the coefficients of the polynomials at vacancy contents $y \neq 0$.

This particular choice of the fitting procedure was motivated by maintaining the continuity between $E_f(x, y)$ and $E_f(x, z)$ while preventing a distortion of the fit by data with no causal contribution in each region. To be more specific: configurations containing metal vacancies do not provide any information about the effect of nitrogen vacancies on the structure and vice versa. Therefore, the terms describing the influence of y and z are separated, and are equal for $y = 0$ and $z = 0$. Fitting the two sub-regions completely independently would result in a discontinuous jump at $y = z = 0$. To prevent this, the terms a_i^0 are generated using only the data of structures without vacancies. The vacancy dependent parts of the E_f-functions, $a_i(y)$ and $a_i(z)$, are then determined separately from each other, but apart from the constrained $a_i(y = 0) = a_i(z = 0) = a_i^0$.

Once the functions $E_f(x, y)$ and $E_f(x, z)$ are found for every phase, the assessment of phase stability is carried out by identifying which structure yields the lowest E_f value for each $x, y/z$.

It is noted that the fitting procedure is critical for the stability assessment, since (slightly) different compositions are used for each phase (various unit cells and supercell sizes, see Fig. 2.2 and 2.3) and hence a direct comparison of DFT values is not possible.

4.2 Structural parameters

Structural properties of the relaxed structures such as specific volume or lattice parameters are easily accessible from the the calculations. As mentioned in Section 3.1.3, the algorithm chosen for the computations leads to a complete structural relaxation including cell volume, cell shape, and ion positions. The most readily available quantity is the specific volume, v_0, which is the relaxed volume V_0 of the used supercell divided by the number of lattice sites, N, it contains:

$$v_0 = \frac{V_0}{N} . \tag{4.7}$$

It can be used to compare the packing density of different phases, which is an indicator for the stability at higher pressures.

Regarding the lattice parameters, cubic structures represent the easiest case. For a $u \times v \times w$ supercell with relaxed volume V_0 and based on a conventional cubic structure, the cubic lattice parameter is defined as:

$$a = \sqrt[3]{\frac{V_0}{u \cdot v \cdot w}} \,. \tag{4.8}$$

4.3 Ta-Al-N

Besides the hexagonal ε phase the cubic B1 structure has been frequently found in pure TaN produced by various synthesis methods [54, 61, 86, 136–138]. Many authors furthermore reported the presence of vacancies in the cubic structure, which seems to influence both phase stability and physical properties [3, 8, 36, 61, 136, 139]. The presence of the B1 phase extends to the quasi-binary $Ta_{1-x}Al_xN$ system, where it was found as a single phase up to aluminum fractions $x = 0.36$ and in a two phase mixture with B4 up to $x = 0.65$ [84–86]. As mentioned in Section 1.1.1, unpublished DFT simulations by Holec et $al.$, regarding this alloy system predicted that the hexagonal B_h phase is stable for low aluminum fractions x up to 0.2, followed by the cubic B1 phase up to $x \sim 0.7$, where a multi-phase region of B1 and hexagonal B4 and B_k was expected [87]. In the aluminum rich regime, the simulations predicted the B4 structure. However, these calculations were performed assuming only perfect structures without defects. These results were reproduced in the course of this work and Fig. 4.1 shows the energies of formation of the considered phases as a function of the aluminum content x. The dashed lines are the third order polynomials $p(x)\big|_{y=z=0}$ introduced in Eq. 4.4. It can be seen that up to $x \sim 0.27$ the ε-structure is predicted to be the most stable one, while for x larger than ~ 0.58 the wurtzite B4 structure yields the lowest E_f values. For $0.27 < x < 0.58$, the E_f values of the B1 and B_k structures are almost the same which suggests co-existence of both phases.

This, however, contrasts the experimental findings in two ways. On the one hand, the hexagonal B_k structure has never been reported for this system and was included in this study as a distorted B4 structure appearing as a transition state between B4 and B1 [28, 88]. On the other hand, the cubic B1 phase was found to be stable even for aluminum content below 0.2 and up to values $x \sim 0.53$ [84–86].

By considering vacancies in the calculations, which are always present in real structures and particularly in physical vapor deposited materials ([8, 23, 140]), a better agreement with experiment is expected. The resulting phase diagram including the impact of metal vacancies is shown in Fig. 4.2. Obviously, the presence of vacancies on the metal sublattice changes the phase stability drastically. At $x = 0$ the ε phase is the most stable for vacancy contents $y < 0.075$. For higher y, the B_h structure becomes the most stable, which is
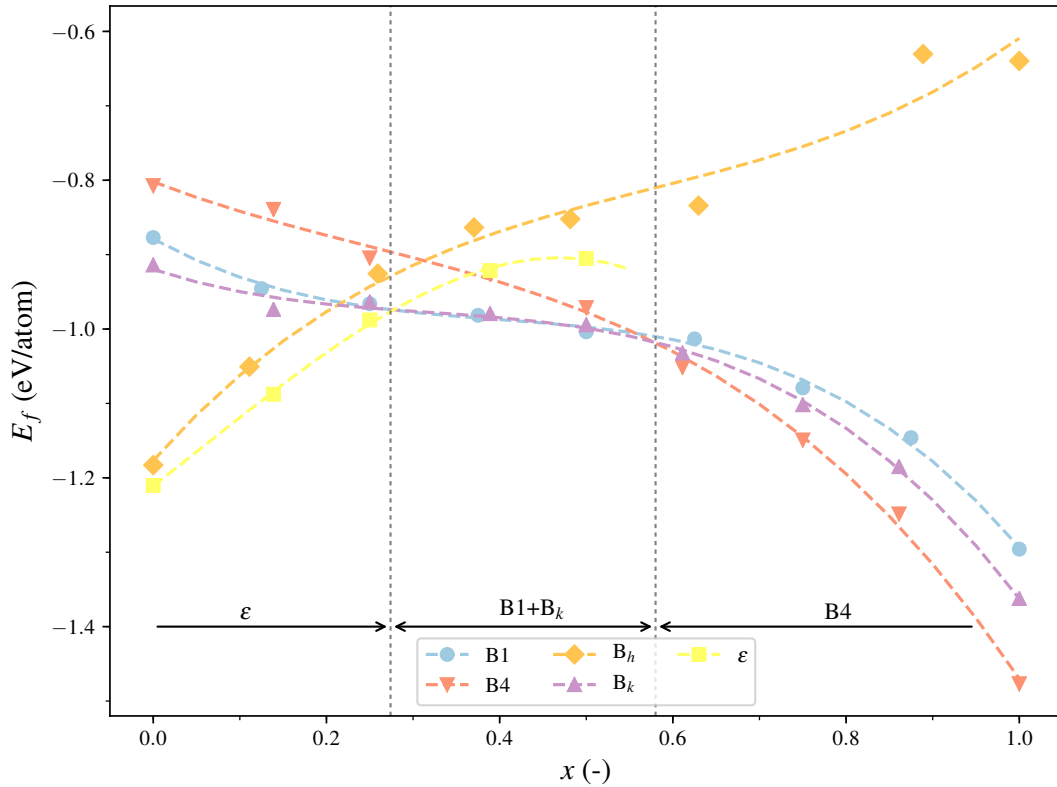
Figure 4.1: E_f as a function of x for $Ta_{1-x}Al_xN$ without vacancies. Symbols represent the actual DFT data, while the fitted polynomials $E_f(x, y = z = 0)$ (Eq. 4.3) are shown with dashed lines.

substituted by cubic B1 $Ta_{1-y}N$ at $y \sim 0.14$. The aluminum fraction at which the phase transition from ε to B1 takes place, originally at $x \sim 0.27$ for $y = 0$, shifts to lower x-values with an increase in y until at $x \sim 0.1$ and $y \sim 0.09$ the B_h structure appears. The B1 phase is stable at even lower x for higher metal vacancy content until B_h vanishes completely at $x = 0$ for $y \sim 0.14$. The switch from B1 to B_k, which is hard to determine from Fig. 4.1 due to the small difference in E_f of the two phases, apparently takes place at $x \sim 0.48$ for $y = 0$. The presence of metal vacancies, however, increases the difference in energy of formation between them in favor of the cubic phase. The B1 phase field extends at expense of the hexagonal phases B_k and B4 for increasing y up to $x \sim 0.7$ at $y \sim 0.07$. But for $y > 0.07$ this effect is revoked and the transition from cubic to hexagonal is moved back to lower x. In any case, the shifts of the phase transitions B1-B_k/B4 and, more importantly, ε-B1 result in a strongly extended B1-phase field.

In contrast to Fig. 4.1, the phase diagram in Fig. 4.2 cannot illustrate the differences in E_f between the individual phases. It was shown above, that they can be quite small for some compositions, thus it is difficult to explicitly identify one phase as the most stable one. If this is the case, a mixture of two or more phases is more likely than a homogeneous material
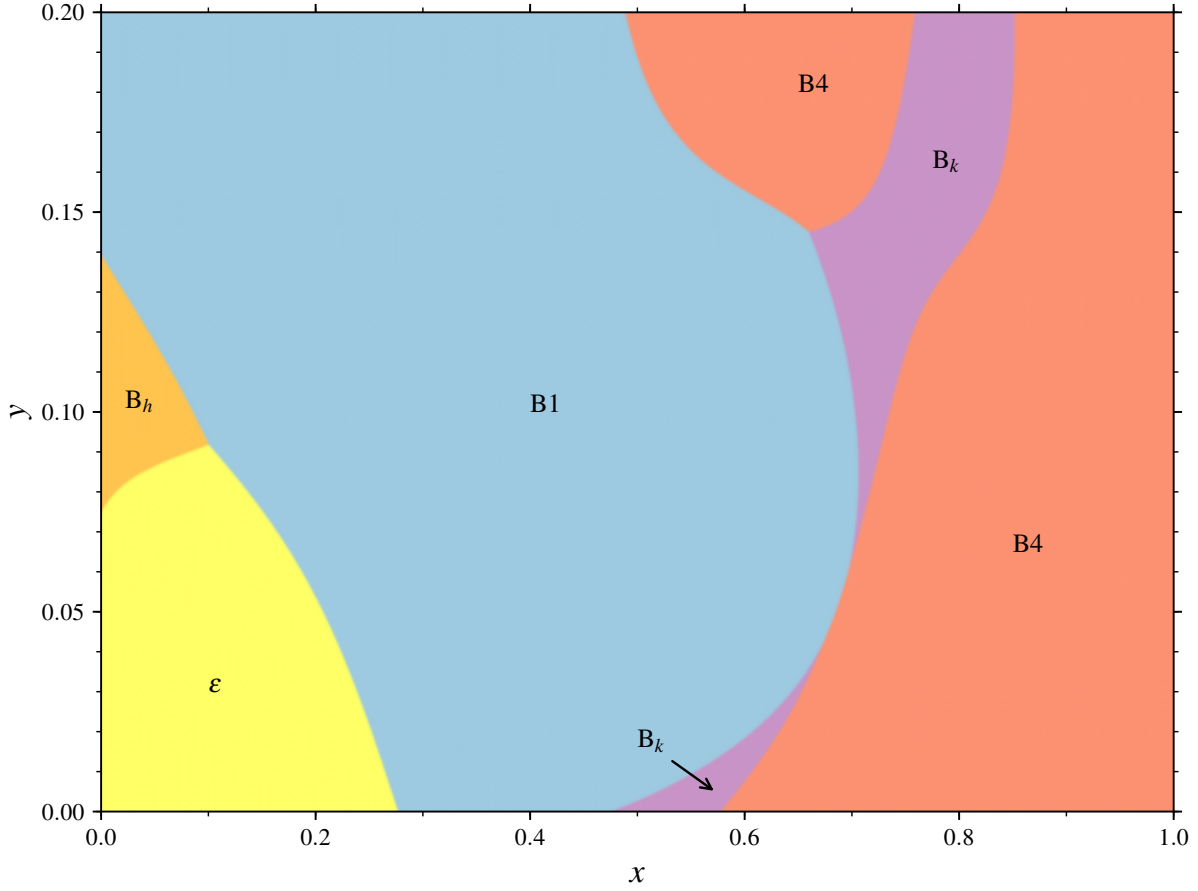
Figure 4.2: Stable phases of $(\mathrm{Ta}_{1-x}\mathrm{Al}_x)_{1-y}\mathrm{N}$ as a function of aluminum fraction, x, and the content of metal vacancies, y.

consisting of a single phase.

To account for this fact, Fig. 4.3 shows an extended phase diagram which identifies regions where two or more phases differ by less than $0.02\,\mathrm{eV/atom}$. As to be expected, several two- and three-phase fields arise from this approach near the transition regions of Fig. 4.2. Also, the small regions where B_h and B_k have the lowest E_f, vanish completely as the energetic difference to the first metastable phase is within $0.02\,\mathrm{eV/atom}$. Consequently, these two phases appear only in a combination with other phases and not independently. B_h is accompanied by ε in the largest part of the original single-phase field. The B_h-ε two-phase region is bounded by a thin three phase field including B1 at higher vacancy content. At $x = 0$, the B1 single-phase field is shifted to slightly higher y-values, $y \sim 0.155$, but in combination with B_h and ε the cubic phase appears already at $y \sim 0.125$.

In agreement with Fig. 4.1, B_k co-exists together with B1 for $0.27 < x < 0.58$ for low y. The two-phase field of B1 and B_k is surrounded by two three-phase fields, one additionally including ε at lower aluminum fractions, and the other one including B4 at $x \sim 0.5 - 0.6$. With increased number of metal vacancies, however, the energetic preference of B1 over B_k
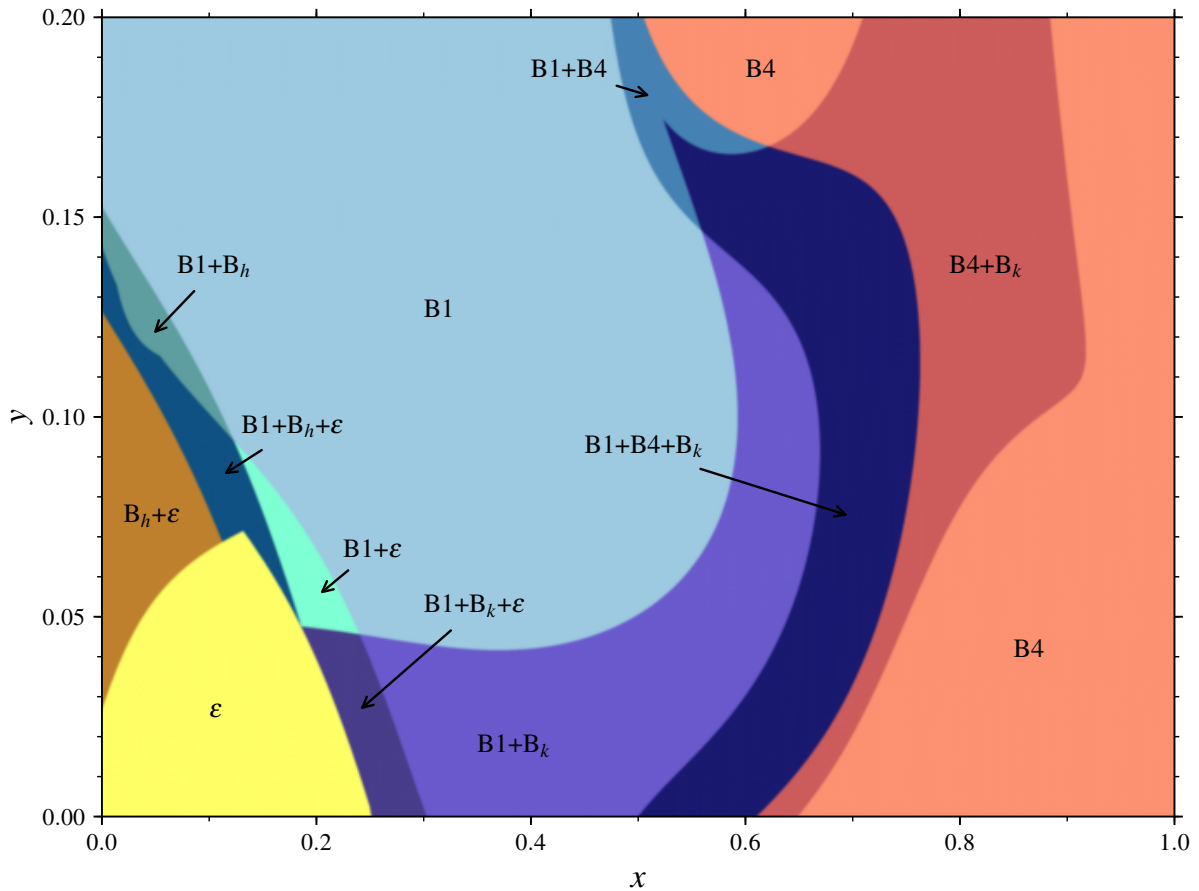
Figure 4.3: Extended phase diagram of $(Ta_{1-x}Al_x)_{1-y}N$ including regions where the $E_f(x,y)$ values of several phases differ less than $0.02\,\text{eV/atom}$ as multi phase fields.

increases, and B1 is found alone over a wide range of aluminum fractions x. Depending on the vacancy content y, the B1 phase is accompanied by the hexagonal phases B4 and/or B_k at $x = 0.48 - 0.58$ until it finally vanishes for higher aluminum fractions x.

The large B1 single-phase fields in both Figs. 4.2 and 4.3, as well as the various multi-phase fields containing it in Fig. 4.3, show that the region where the cubic phase is predicted to be stable, significantly expands by increasing the metal vacancy content.

The reason for the stabilization of B1 apparently caused by the metal vacancies can be found in Fig. 4.4. It depicts E_f of the individual phases for various metal vacancy contents. While B_h and ε experience a considerable increase in E_f for increasing y values at low aluminum fractions x, the formation energy of B1 is lowered, making the structure more favorable. This is in agreement with calculations by other authors of pure TaN, which showed that the cubic phase favors vacancies [3, 37, 141]. This decrease in E_f extends up to $x \sim 0.5$ where the metal vacancies eventually have an adverse effect. B_k, on the other hand, is affected by the metal vacancies less than the other phases, although a very small reduction in E_f can be observed for compositions up to $x \sim 0.9$. Similar to B_h and ε, the energy of formation
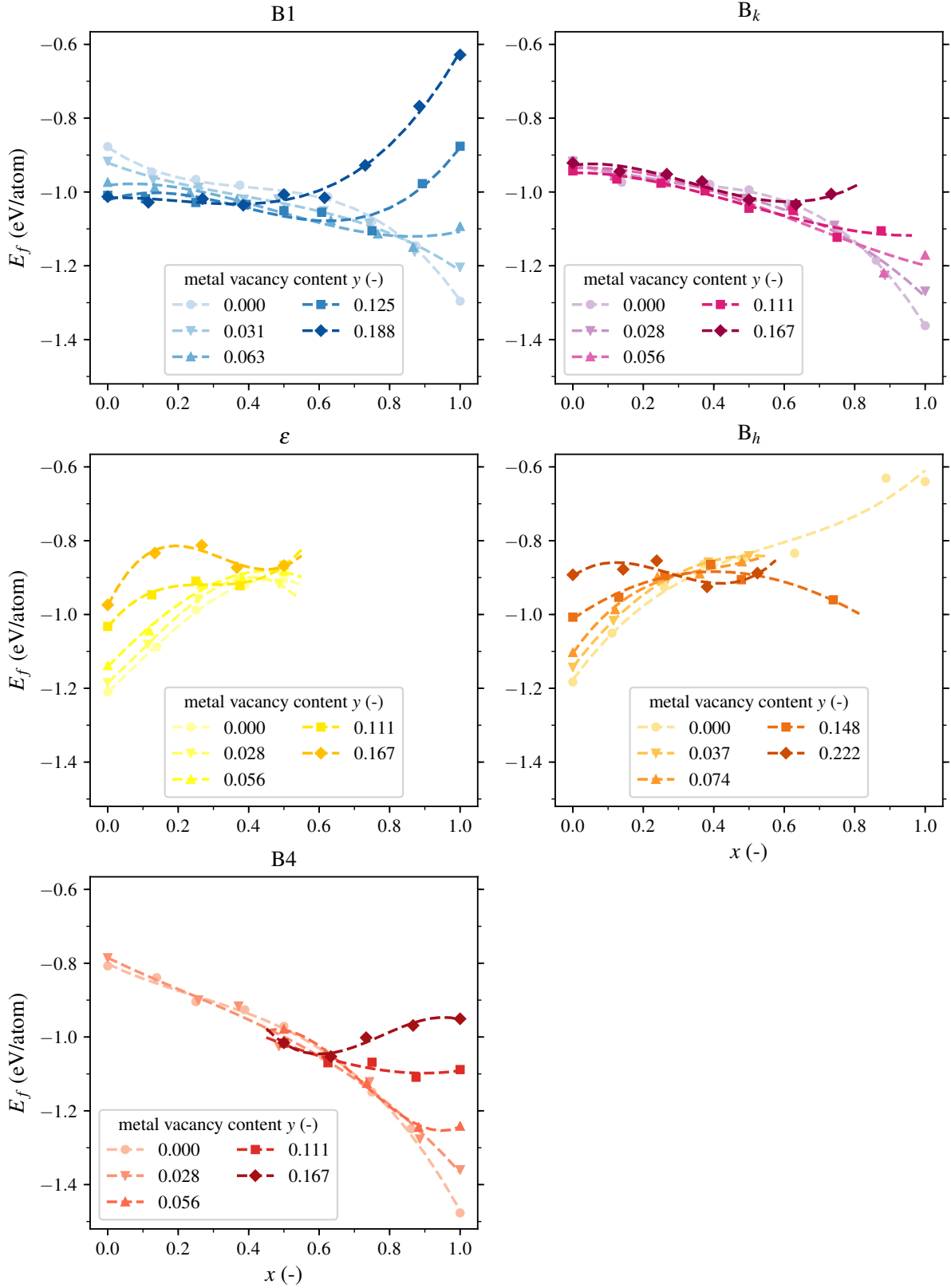
Figure 4.4: E_f against x plots at different metal vacancy contents y of all considered phases in the Ta-Al-N system. The symbols represent the discrete E_f data points while the dashed lines are the third order polynomials of Eq. 4.4 used to construct the coefficients $a_i(y)$ (Eq. 4.5).

of B4 increases for increasing y. Metal vacancies are undesirable for all structures when the aluminum content is high.

Interestingly, the curvature of the $E_f(x)$ functions of the B1 phase changes gradually from concave to convex with increasing y-values (Fig. 4.4). This means that the solid solution becomes chemically stable for aluminum fractions $0 < x < 1$, and loses its tendency for an iso-structural decomposition.

Some of the B4, B_h and ε cases were extremely difficult to stabilize (i.e., to converge the structural relaxation during the calculation) especially for higher vacancy concentrations. This, in addition to the high E_f values, indicates, that the respective configurations are not stable and try to change the atom position and cell dimensions during the ionic relaxation process. It also contributes to somewhat scattered data points found in the respective plots of Fig. 4.4.

Another factor that contributes to this scatter and the long calculation times is the procedure chosen for the SQS creation. As pointed out in section 2.2.2, only pair interactions were taken into account to assess the population of the supercells. This approach, however, seems to be insufficient for some phases and compositions, as SQS that were assessed to be equally good, yielded slightly different E_f values. If this was the case for a specific composition and vacancy arrangement, the SQS yielding the lowest E_f was considered to be the representative one. To reduce or, ideally, prevent this behavior completely, it is necessary to consider higher order geometric correlation functions in addition to the pairs used here.

As already mentioned in Section 1.1.1, the thin films investigated by Koller *et al.* had an overall chemical formula of $Ta_{0.89}Al_{0.11}N_{1.2}$ [84]. If the nitrogen over-stoichiometry is attributed to vacancies on the metal sublattice, this would translate to a vacancy content of $y = 0.167$ in the notation used here. This composition lies well within the B1 phase field according to the phase diagram in Fig. 4.3. The phase diagram also shows good agreement with the experimental study of Zhang, which reports a single phase cubic structure for $Ta_{1-x}Al_xN$ films with aluminum fractions x up to 0.36, and a mixture of cubic and hexagonal phases for x up to 0.65 [86].

The result is quite different for nitrogen vacancies. The resulting phase diagram is shown in Fig. 4.5. Similar to Fig. 4.3, it displays regions where the energy of formation of two or more phases differ less than $0.02\,\mathrm{eV/atom}$ as multi-phase fields. It shall be noted that no cases with nitrogen defects were considered for the B_k structure, and therefore it cannot appear in the phase diagram in Fig. 4.5. The results on the impact of metal vacancies presented above showed that the B_k phase does not occur as the single most stable structure throughout the probed configurations and thus confirmed its role as a transition state in the transformation between the cubic B1 and the hexagonal B4 phases. Moreover, B_k is most important in the regions with higher x where is the direct transition from B1 to B4. It plays only a minor role at low aluminum fractions, which are the center of interest for this work as the available
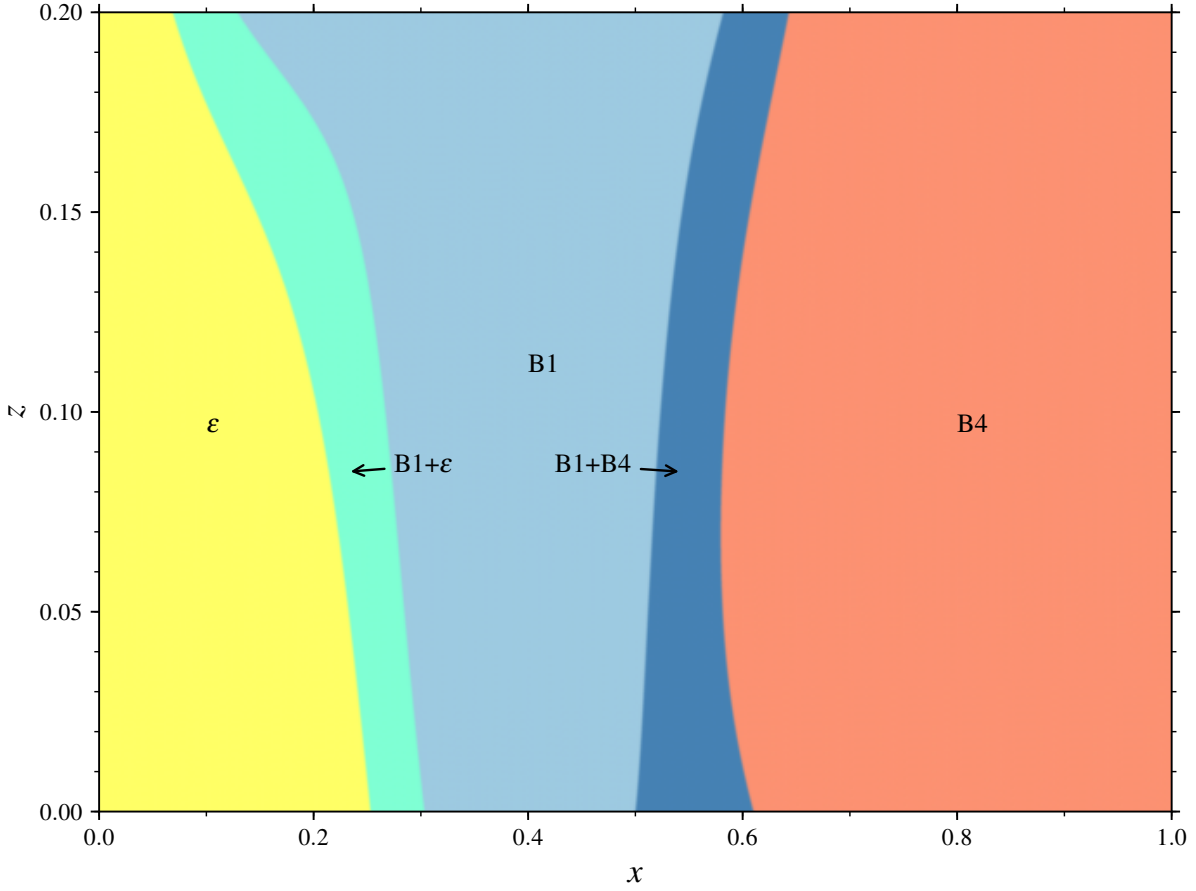
Figure 4.5: Extended phase diagram of $Ta_{1-x}Al_xN_{1-z}$ explicitly disclosing regions where the $E_f(x,z)$ values of several phases differ less than $0.02\,\mathrm{eV/atom}$ as multi-phase fields.

experimental findings lie in this region.

It turns out that the impact of nitrogen vacancies on the individual phase fields is not as strong as the impact of metal vacancies. Although the B1 phase field is slightly expanded by an increased vacancy content z, the effect is much less pronounced than in the case of metal vacancies. The transition from the hexagonal ε to the cubic B1 phase is shifted from $x \sim 0.27$ in the perfect phases to lower aluminum fractions by increasing z. The transformation from B1 to B4, on the other hand, appears at slightly higher x for increased vacancy contents compared to the perfect configurations. Fig. 4.6 reveals the reason for this. The presence of nitrogen vacancies reduces the energy of formation of the B1 structure by a small amount for tantalum-rich compositions. This is in agreement with the findings of Koutná $et\ al.$ in pure TaN, who found a small decrease for nitrogen deficient compositions [3, 142]. Above $x \sim 0.2$, however, they have the opposite effect indicating that nitrogen vacancies are unfavorable for those compositions. In all other structures, B4, B_h, and ε, nitrogen vacancies increase the heat of formation to a similar extend as the metal vacancies.

Marihart $et\ al.$ synthesized pure TaN films by high power impulse magnetron DC and by
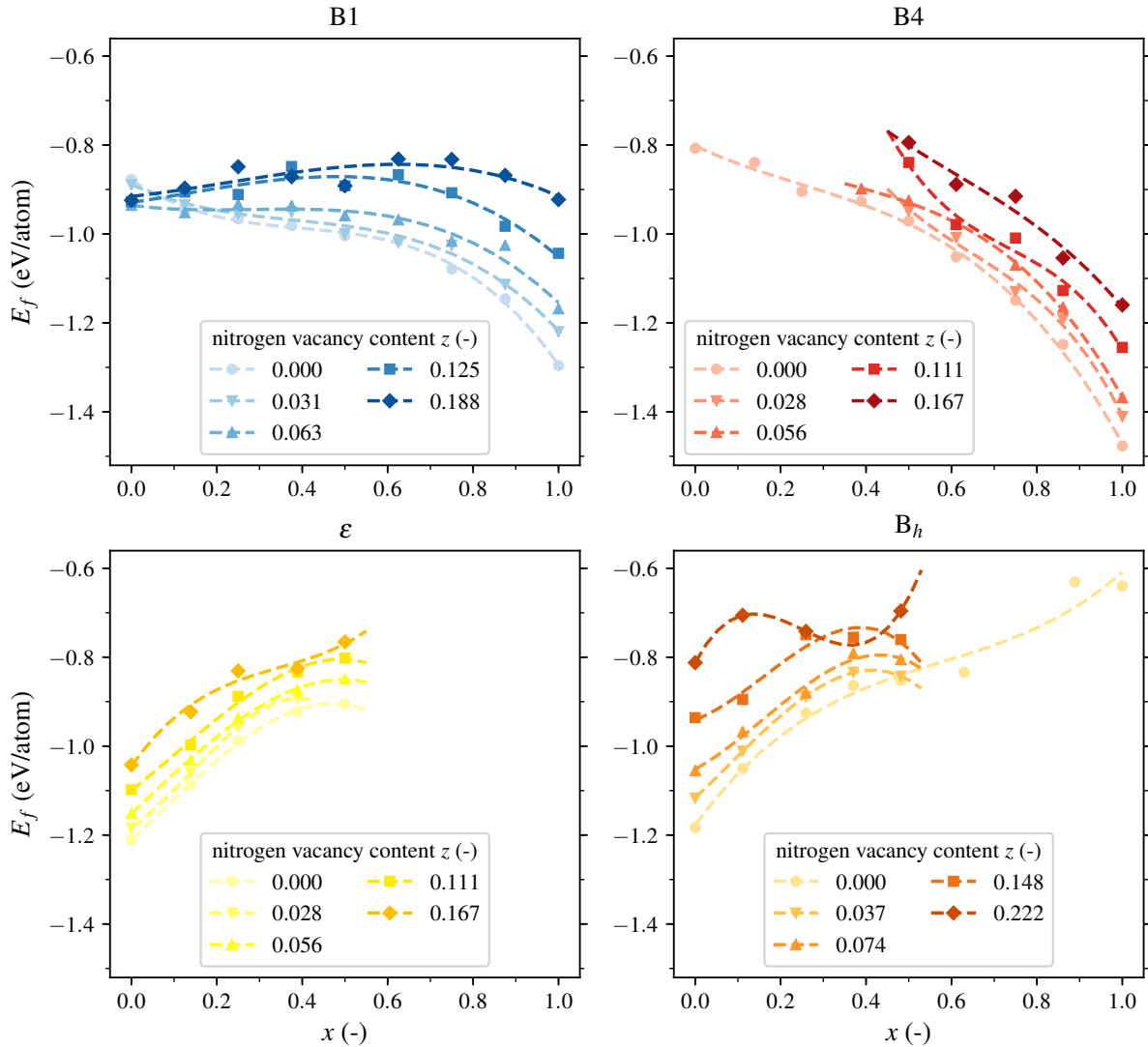
Figure 4.6: E_f over x plots at different nitrogen vacancy contents z of all considered phases in the Ta-Al-N system. The symbols represent the discrete E_f data points while the dashed lines are the third order polynomials (Eq. 4.4) used to construct the coefficients $a_i(z)$.

conventional DC sputtering. Materials deposited with nitrogen partial pressures $p_N/p_T > 38.1\%$ showed a mixture of cubic B1 and hexagonal ε phases with increased ε-fractions for increasing nitrogen partial pressure. Furthermore, the films were found to be nitrogen deficient and to have a high defect concentration. Comparing these findings with Fig. 4.5, it becomes clear that the B1+ε micro-structure cannot be solely explained by the influence of nitrogen vacancies, as the phase diagram predicts a single-phase ε material for $x = 0$. While their presence is, without a doubt, one factor, it must be concluded that other parameters such as other defects, limited diffusion kinetics, or intrinsic stresses in the coatings (all caused by the specific deposition conditions) are needed to allow the crystallization of the cubic B1 phase in pure TaN_{1-z}.

Figure 4.7 presents the specific volume of the perfect structures together with quadratic polynomial fits. It shows that in all structures, v_0 decreases with increasing aluminum content
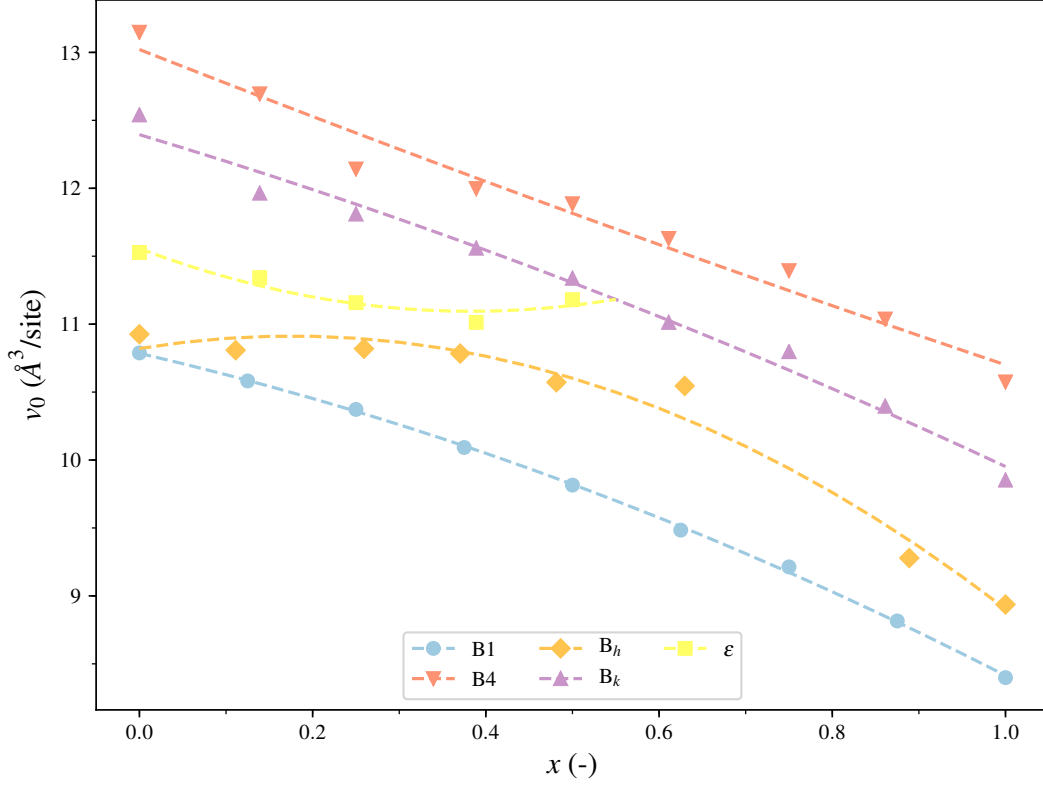

Figure 4.7: Specific volume, v_0, over x of the perfect phases of the Ta-Al-N system. The symbols represent the data from the DFT calculations and the dashed lines are fitted second order polynomials.

on the metal sublattice due to a smaller atomic radius of Al as compared to Ta. v_0 of B1, B4 and B_k change almost linearly with x, suggesting that v_0 of these phases follows Vergard's behavior [143, 144]. On the other hand, the specific volumes of B_h and ε deviate significantly from linearity. The B1 phase is the most densely packed one. This is important because the above considerations regarding the phase stability hold for external pressure of $p = 0\,\mathrm{GPa}$. Compressive intrinsic stresses, which are generally desirable in protective coatings, further extend the stability of structures with smaller specific volume [145, 146]. Therefore, PVD synthesized Ta-Al-N thin films are expected to crystallize in the cubic B1 structure for even larger compositional range than shown in Figs. 4.2, 4.3, and 4.5 for $p = 0\,\mathrm{GPa}$.

The impact of both, metal and nitrogen vacancies, on v_0 is illustrated in Fig. 4.8. The dashed lines are second order polynomials. Missing metal atoms result in a reduction of specific volume in all considered structures and the extend of this effect decreases with increasing aluminum content x. Nitrogen vacancies also reduce the specific volume, although the reduction is less than for metal vacancies, which can be explained by N atoms being
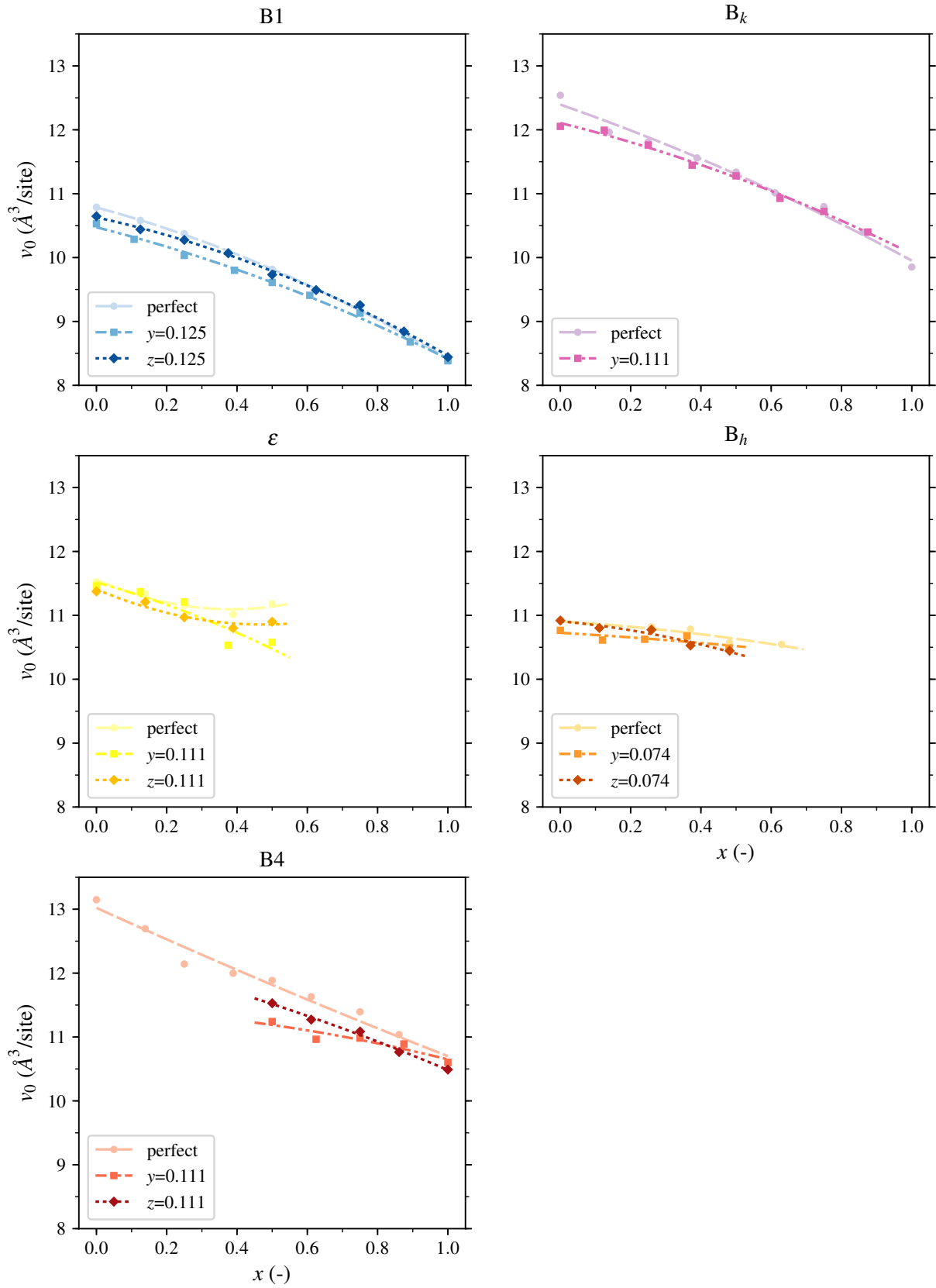
Figure 4.8: v_0 against x plots for the perfect configurations and different vacancy contents y and z of all considered phases in the Ta-Al-N system. The symbols represent the actually calculated data points while the dashed and dotted lines are second order polynomials.

smaller than Al and Ta. However, in some cases, such as the Ta-rich ε phase or the Al-rich B4 structure, the effect of nitrogen defects is larger than for metallic ones.

Fig. 4.9 shows the lattice constant a of the cubic B1 structure as a function of the aluminum content, including the influence of metal and nitrogen vacancies. The lattice constant was
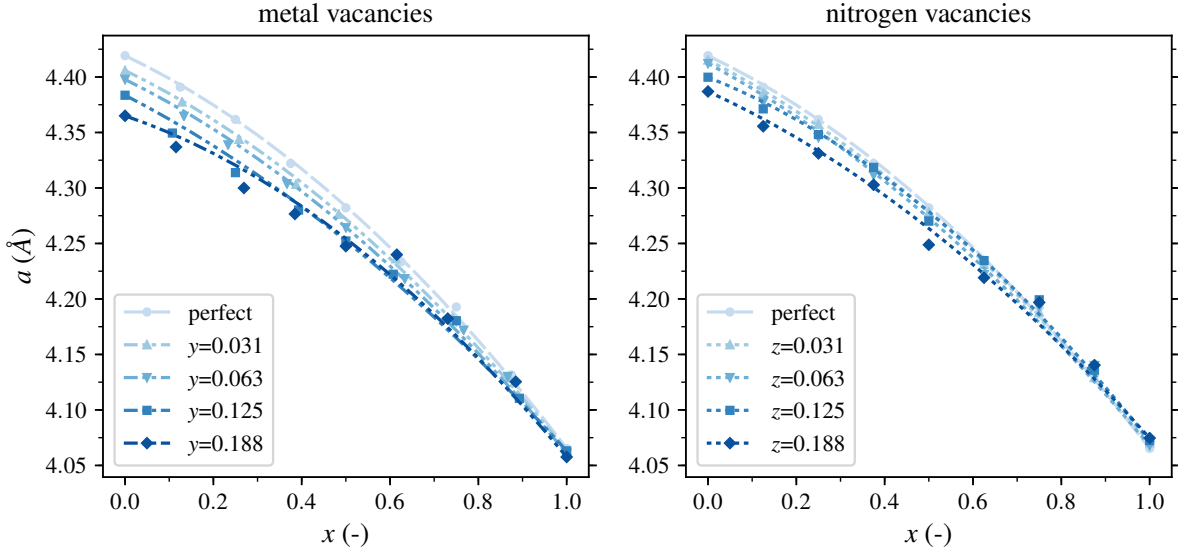

Figure 4.9: Lattice constants $a(x)$ including the influence of metal and nitrogen vacancies on them for the cubic B1 phase in the Ta-Al-N system. The symbols show the actual data and the dashed and dotted lines are fitted functions according to Eq. 4.9.

calculated from the relaxed volume according to Eq. 4.8 and fitted to a quadratic polynomial of the form:

$$a(x) = a_{\text{AlN}}x + a_{\text{TaN}}(1 - x) + bx(1 - x) \tag{4.9}$$

This is a quadratic modification of Vegard's law ([143]) with a_{AlN} and a_{TaN} being the lattice constants of the boundary systems AlN and TaN. b is a bowing parameter and quantifies the deviation of $a(x)$ from a linear, Vegard-like relation. The lattice constants and the bowing parameters of various configurations are listed in table 4.2. It is noted that DFT calculations using GGA exchange correlation potentials generally underestimate bonding, which leads to lattice constants and bond lengths larger than found experimentally. It follows from Fig. 4.9 and Table 4.2 that metal vacancies reduce the lattice constants and that missing Al atoms have a smaller impact than Ta vacancies. The same trend can be observed for nitrogen vacancies in TaN but the reduction of a_{TaN} is much smaller than the one resulting from metal defects. For AlN, however, an increase in nitrogen vacancy content seems to increase the lattice constant a_{AlN}. The same peculiar behavior was also reported by Koutná [142]. In all cases, positive b values are obtained describing bowing of $a(x)$ above the linear behavior. Also, the bowing parameters seem to be only weakly affected by the presence of vacancies

Table 4.2: Lattice constants and bowing parameters according to Eq. 4.9 of the cubic B1 $(Ta_{1-x}Al_x)_{1-y}N$ and $Ta_{1-x}Al_xN_{1-z}$. All parameters are given in Å.

	a_{TaN}	a_{AlN}	b
perfect	4.419	4.065	0.165
metal vacancies			
y			
0.031	4.406	4.063	0.150
0.063	4.398	4.061	0.141
0.125	4.384	4.063	0.116
0.188	4.365	4.058	0.173
nitrogen vacancies			
z			
0.031	4.415	4.070	0.143
0.063	4.412	4.068	0.132
0.125	4.400	4.072	0.170
0.188	4.387	4.075	0.131

Table 4.3: Lattice constants a of the cubic B1 phase found in literature. All values in Å.

structure	present	experimental	theoretical	
			LDA	GGA
TaN	4.419	4.43[147], 4, 344[62]	4.353[141], 4.386[148], 4.37[149]	4.398[141], 4.424[141], 4.427[142], 4.42[149]
$Ta_{0.969}N$	4.406			4.41[141], 4.417[142]
$Ta_{0.938}N$	4.398			4.40[141], 4.403[142]
$Ta_{0.875}N$	4.384	4.361[150]		4.380[142]
$TaN_{0.969}$	4.419			4.42[141], 4.422[142]
$Ta_{0.875}Al_{0.125}N$	4.391	4.400[86]($Ta_{0.89}Al_{0.11}N$)		
$Ta_{0.750}Al_{0.250}N$	4.362	4.358[86]		
$Ta_{0.625}Al_{0.375}N$	4.322	4.341[86]($Ta_{0.64}Al_{0.36}N$)		
AlN	4.065	4.045[151]	4.06[152], 4.03[153]	4.07[154], 4.069[142]

and no clear trend can be found.

a_{TaN} and a_{AlN} have been reported by a number of both, experimental and theoretical studies, and can thus be used to assess the reliability of the present results. An example of published lattice constants is listed in table 4.3. The lattice parameters presented here clearly agree with the previously reported values.

4.4   Nb-Al-N

As in the Ta-Al-N system, the cubic B1 structure has been reported by various authors in the Nb-Al-N system. An early study of Oya and Onodera described the phase transformation in NbN_{1-z} films from the as deposited cubic state to the hexagonal B_i phase via several intermediate structures [79]. Also, more recent studies found NbN in B1 configuration either as a single phase or as a part as a two-phase mixture depending particular on a deposition technique and conditions [76, 82]. Regarding the quasi-binary $Nb_{1-x}Al_xN$ system, Selinder *et al.* reported a single-phase cubic structure in reactively triode sputtered thin films up to aluminum fractions $x \sim 0.5$ and together with the hexagonal B4 phase up to $x \sim 0.6$ [89]. Experiments by Franz *et al.* yielded similar results for cathodic arc evaporated materials with mainly cubic structure up to $x \sim 0.56$ depending on the bias voltage [91]. *Ab initio* calculations by Holec *et al.* on several relevant structures, however, predicted a hexagonal $B8_1$ phase in the quasi-binary system for $x < 0.14$ [92].

It has to be noted that the study of Holec *et al.* included the $B8_1$ structure, while here the B_i is considered instead. This is justified by their structural similarity which was already pointed out in the mentioned paper. The reason for choosing the B_i structure over $B8_1$ was that it yielded a slightly lower E_f-value for the perfect configuration of NbN. Hence, when comparing the $E_f(x)$ plots for the perfect configurations shown in Fig. 4.10 with the publication of Holec *et al.*, the shape of the hexagonal B_i and $B8_1$ structures are qualitatively the same, but the phase transition from hexagonal to B1 is predicted to happen at a slightly lower aluminum fraction of $x \sim 0.14$ by Holec *et al.* than in the present study. Again, the dashed lines in Fig. 4.10 are third order polynomials $p(x)\big|_{y=z=0}$ (Eq. 4.4) fitted with the E_f values of each structure. This over-fits the function for the B_i phase since only 4 configurations could be successfully relaxed by the calculations. Nevertheless, this fitting function is used in order to describe all structures in a consistent way. Using the calculated data for defective materials, and following the procedure described in section 4.1, functions $E_f(x,y)$ and $E_f(x,z)$ were fitted for all phases in the system, B1, B4, B_i, and B_k. The phase diagram illustrating the influence of metal vacancies is shown in Fig. 4.11. It reveals that the phase boundaries depend on the amount of metallic defects, y. The phase field of the hexagonal B_i diminishes due to an increased metal vacancy content y. The transition from the B_i to the B1 phase is shifted from $x \sim 0.16$, at $y = 0$ to lower x for higher y-values.

The transformation from B1 to B_k moves to higher x for small y values and the B_k phase quickly vanishes at increasing y. The direct transition from B1 to B4, appearing for $y \gtrsim 0.045$, is found at $x \sim 0.72$ for $y \sim 0.1$ and then shifts back to lower x for even higher y-values.

Fig. 4.12 shows the extended phase diagram including the multi-phase regions for the $(Nb_{1-x}Al_x)_{1-y}N$ system. It makes clear that large parts of the original B_i region exist as a B1+B_i two-phase field, especially for high y values. Similarly to the Ta-Al-N system, also
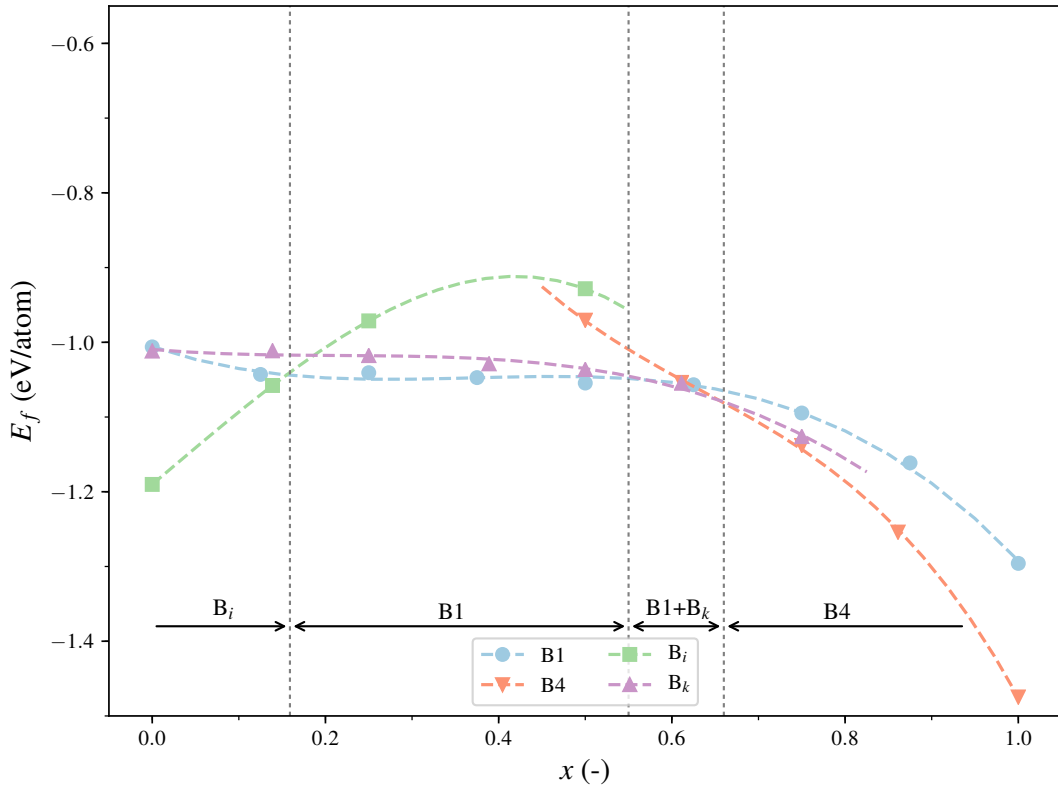
Figure 4.10: E_f for the considered phases as a function of x in the $Nb_{1-x}Al_xN$ system without vacancies

in the Nb-Al-N system the B_k phase never appears as the single most stable phase, but only in combination with either B1 or B4. Moreover, it does not appear for $x \lesssim 0.44$ even in the perfect configuration. The transition from the cubic B1 to the hexagonal $B4/B_k$ structure is initiated by a two-phase field and followed by a region where the E_f values of all three phases lie within the $0.02\,\mathrm{eV/atom}$ range. The edges of the two-phase regions B1+B4 and B1+B_k denote the maximum solubility of Al in the cubic B1 phase, which changes from $x \sim 0.44$ at $y = 0$ to $x \sim 0.66$ at $y \sim 0.107$. A comparison of these values with the experiments of Selinder *et al.* and Franz *et al.* shows quite good agreement for this transition [89, 91]. At lower x, however, the hexagonal B_i phase remains stable in Fig. 4.12 even at higher y values and is found either alone or together with B1. The lowest aluminum fractions probed by the above studies were $x \sim 0.09$ in the case of Selinder *et al.*, and $x = 0.27$ at Franz *et al.*. Therefore, the single-phase cubic materials reported by them for low x do also agree well with the results of the present simulations [89, 91].

The impact of metal vacancies on the energy of formation of the individual phases is illustrated in Fig. 4.13. Consistent with the results of Section 4.3, they produce a strong increase of E_f in all structures for aluminum-rich configurations with the strongest impact on B1. Generally, metal vacancies lead to an increase in the heat of formation compared to
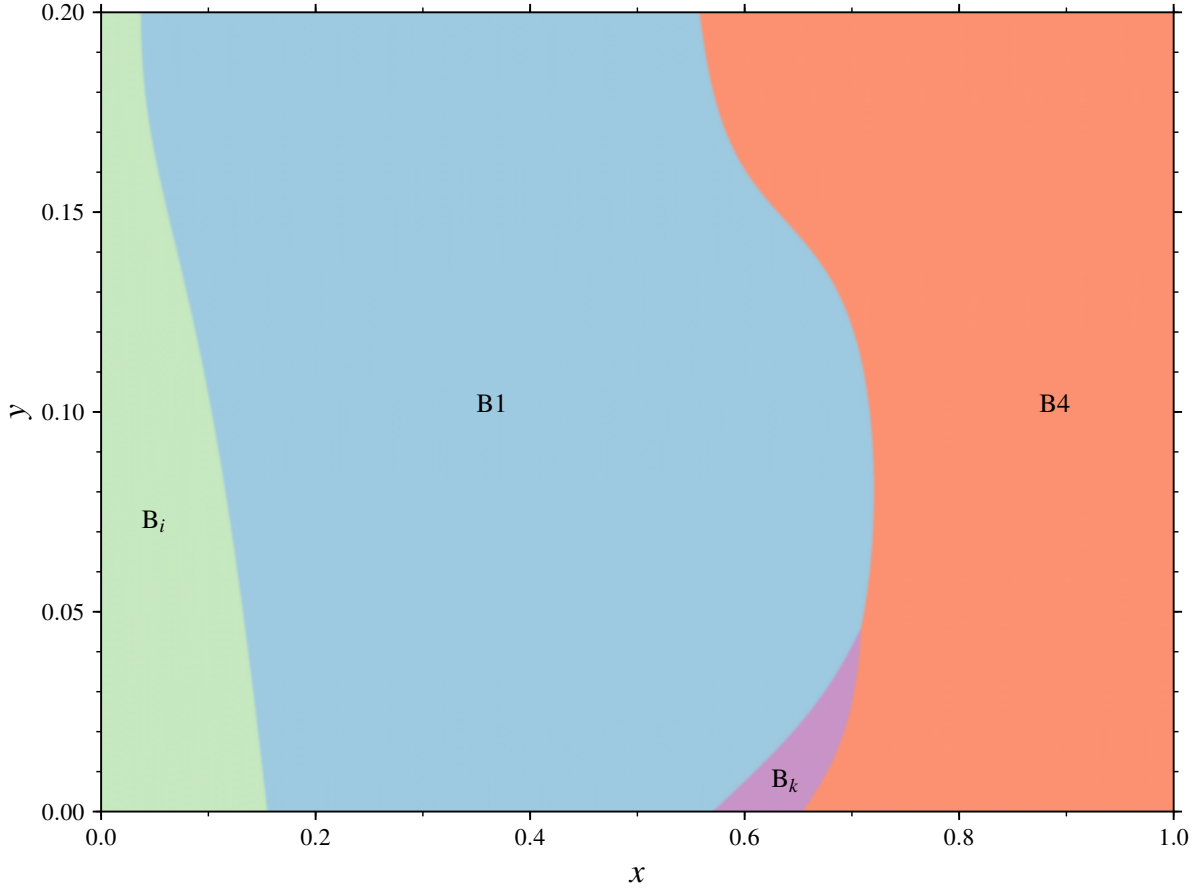
Figure 4.11: Stable phases in $(\text{Nb}_{1-x}\text{Al}_x)_{1-y}\text{N}$ as a function of aluminum concentration on the metal sublattice, x, and the metal vacancy content, y.

the perfect configurations of B_k, B_i and B4 for all aluminum fractions x. On the contrary, they lower the E_f values in the cubic B1 structure, but this effect is very sensitive to the chemical composition. Low vacancy concentrations of $y = 0.031$ seem to be energetically slightly more favorable than the perfect structure up to an aluminum fraction of $x \sim 0.875$. This effect is increased to a small amount at $y = 0.063$ for small x, but it then extends only up to aluminum fractions of $x \sim 0.75$. Higher amounts of metal vacancies are only advantageous for very small aluminum content. But even in pure Nb_{1-y}N, the maximum stabilization (most negative E_f) is achieved for vacancy concentrations y between 0.063 and 0.125. The curvatures of the E_f plots presented in Fig. 4.13 for the various structures are mostly concave, with only few local exceptions. This indicates that compositions with $x > 0$ have positive mixing enthalpy and are therefore metastable.

A study by Benkahoul *et al.* on NbN_{1-z} films prepared by reactive magnetron sputtering reported different crystalline structures in the material, depending on the deposition conditions [82]. With substrates heated to $400\,^\circ\text{C}$, the film showed a mixed structure of cubic B1 and hexagonal B8_1 phases for all probed N/Nb-ratios $1-z$. The B8_1 content in the material
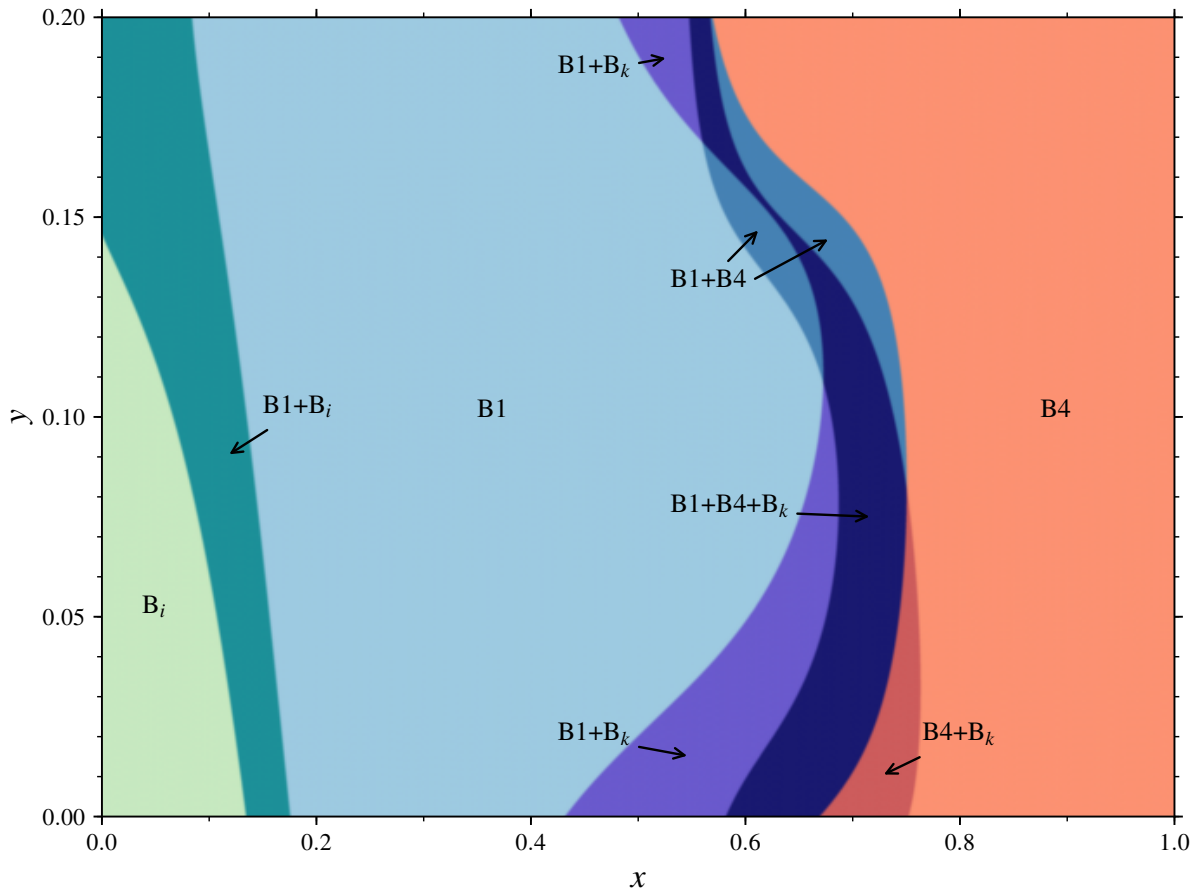
Figure 4.12: Extended phase diagram of $(Nb_{1-x}Al_x)_{1-y}N$. Regions where the $E_f(x,y)$ values of several phases differ less than $0.02\,eV$ are considered to be multi phase fields.

was found to be dependent on z and reached a maximum for $0.97 < (1 - z) < 1$. Films prepared without substrate heating showed a single phase cubic structure for $(1 - z) < 0.98$ and a mixed structure of B1+B8$_1$ for $0.98 \leq (1 - z) \leq 1.06$.

In Fig. 4.12 a two-phase region of cubic B1 and hexagonal B$_i$ phases is found at $x = 0$ for $y > 0.145$, but a single-phase cubic structure for pure Nb$_{1-y}$N cannot be stabilized by metal vacancies. However, the energy difference between the two structures is significantly reduced by the metal vacancies and it is only reasonable that their presence plays an important role in the stabilization of the cubic phase found in experiment. Other factors such as the preparation conditions or intrinsic stresses, which are not considered here but affect the Gibbs free energy in Eq. 4.1, should be taken into account and are expected to improve the agreement of the simulations. Benkahoul *et al.* found the single phase cubic structure in films prepared without heating the substrate. In materials prepared with a substrate temperature of $400\,°C$, the hexagonal structure was present for all N/Nb ratios. It is therefore possible that the formation of the hexagonal phase(s) is kinetically impeded if temperatures are too low, hence leading to crystallization of single-phase cubic films. Hultman pointed out that kinetically

Figure 4.13: E_f against x plots for different metal vacancy contents y of all considered phases in the Nb-Al-N system. The symbols represent the E_f data points and the dashed lines are the third order polynomials of Eq. 4.4 used to fit the coefficients $a_i(y)$ (Eq. 4.5).

limited conditions foster the stabilization of the cubic B1 phase in quasi-binary TMNs [17]. Oya and Onodera, and Skelton *et al.* reported a partial transformation of the cubic B1 phase in NbN films to the hexagonal B_i one upon annealing, which supports this theory, but there is some dispute whether this transformation happens via the $B8_1$ phase or not [78, 79, 155]. Sandu *et al.*, on the other hand, reported the deposition of a single-phase cubic structure (with a $B8_1$ content of less than 5%) in reactively magnetron-sputtered NbN_{1-z} films with substrates heated to $400\,°C$ for a nitrogen partial pressure of 37% [81]. Such deposition conditions, however, would lead to changed chemical potentials in the energy of formation (E_i in Eq. 4.2), and are not adequately represented by the reference states listed in Table 4.1 [3, 37].

The results of the calculations on structures with nitrogen vacancies lead to the phase diagram shown in Fig. 4.14. As in the Ta-Al-N system, the B_k structure was not considered in the calculations of nitrogen deficient cases. It can be seen that nitrogen defects affect
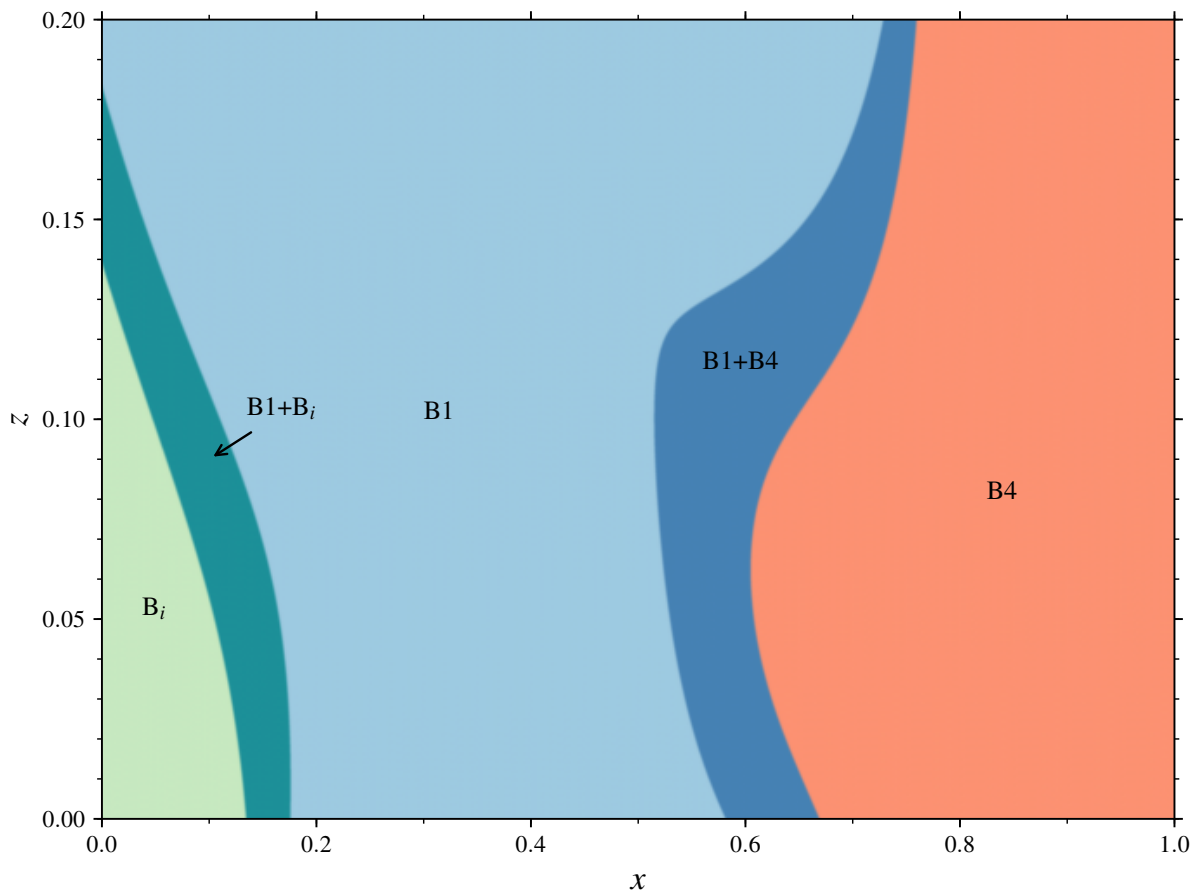
Figure 4.14: Phase diagram of $(Nb_{1-x}Al_x)N_{1-z}$ with aluminum concentration on the metal sublattice, x, and nitrogen vacancy content, z.

the transition from the B1 to the B4 phase only weakly. For $z < 0.07$, a minor shift of the transition to lower x compared to the perfect case can be found. For $0.07 < z < 0.12$, the two-phase field expands because the cubic B1 phase is stable up to higher aluminum fractions. The stability of the hexagonal B4 phase decreases at vacancy contents $z > 0.12$, resulting in a single phase cubic field extending up to $x \sim 0.74$ at $z = 0.2$.

The phase fields including the B_i phase are significantly reduced by the presence of nitrogen vacancies. The maximum aluminum fraction is steadily lowered from $x \sim 0.16$ in the perfect case to lower values with increasing z, until it completely vanishes at $z \sim 0.185$. For $z > 0.185$ the single phase field of the cubic B1 phase extends from $x = 0$ to $x = 0.7$.

Fig. 4.15 shows the impact of nitrogen vacancies on the energy of formation of the considered phases in detail. Nitrogen vacancies are apparently undesired in the hexagonal phases B4 and B_i, as their E_f values increase significantly with increasing vacancy content z compared
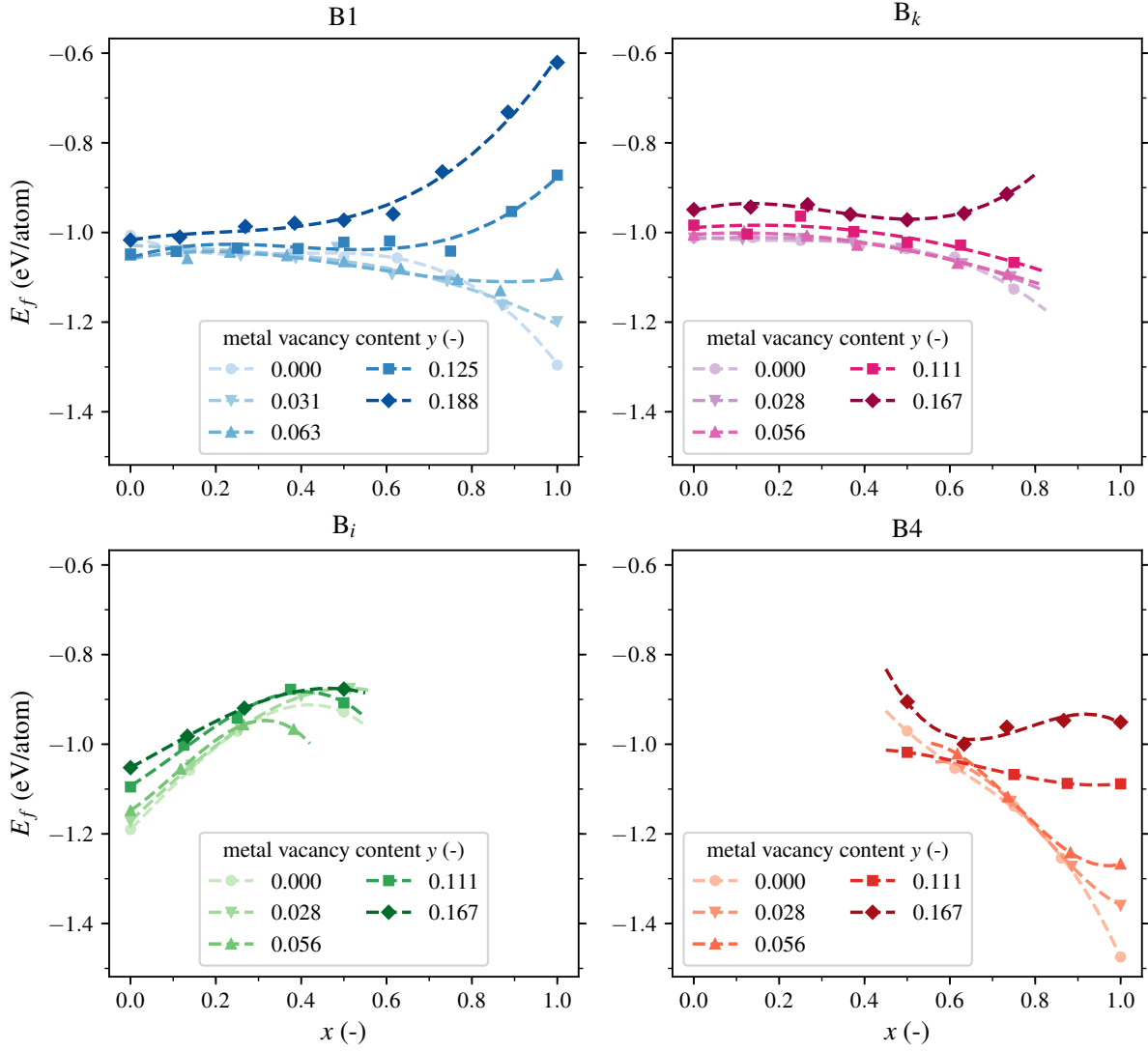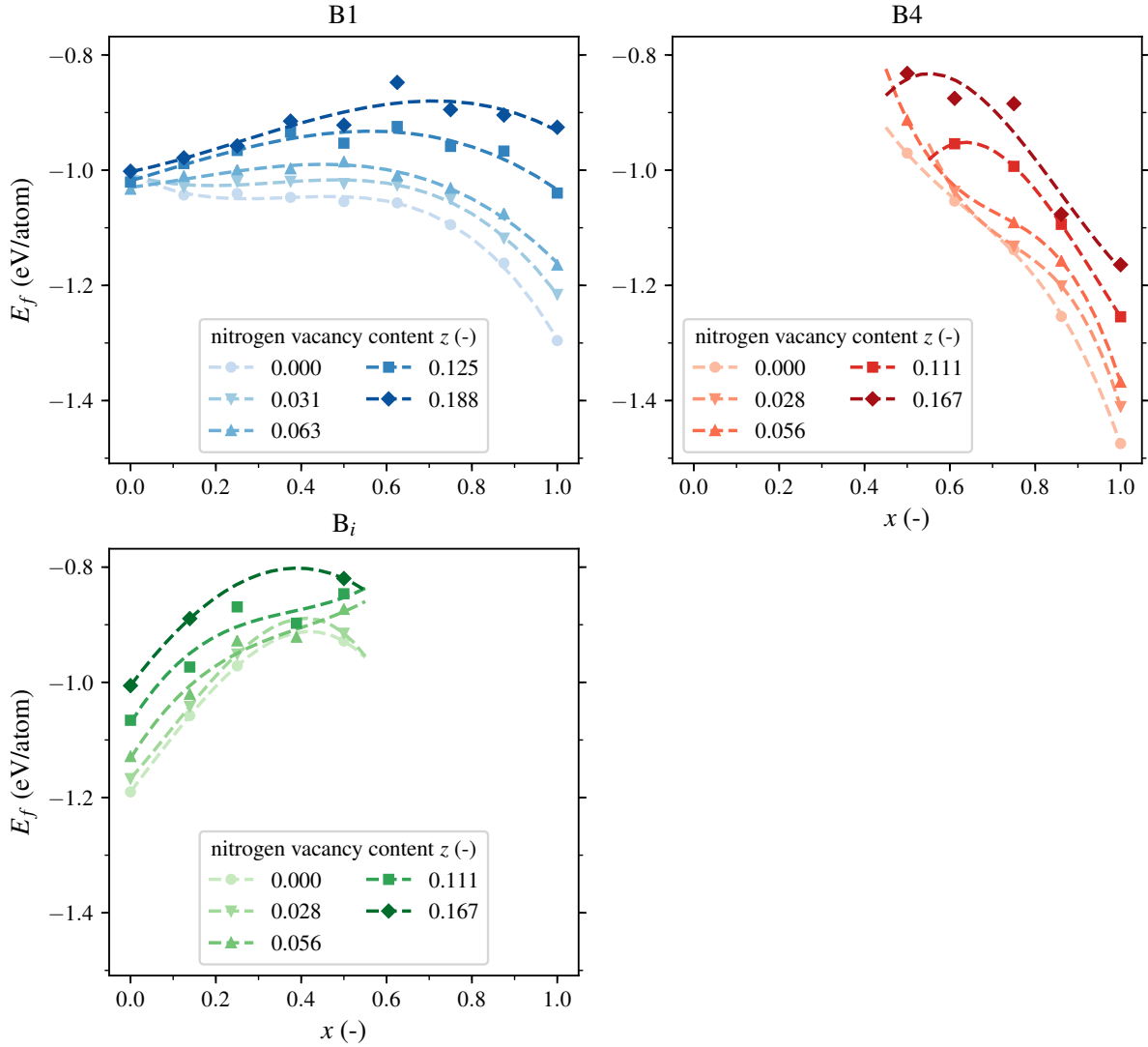
Figure 4.15: E_f against x for different nitrogen vacancy contents z of all considered phases in the Nb-Al-N system. The symbols represent the E_f data points and the dashed lines are the third order polynomials of Eq. 4.4 used to fit the coefficients $a_i(z)$ (Eq. 4.5).

to those of the perfect cases in all chemical compositions considered. But also in the cubic phase, the nitrogen vacancies seem to be unfavorable for most compositions. Only in pure NbN_{1-z}, a small decrease in energy of formation is observed. Nevertheless, for aluminum fractions below 0.3, the energy penalty of the nitrogen vacancies is less pronounced in the B1 than in the B_i structure, effectively making the cubic phase more stable. The same phenomena, only weaker, is seen for B1 and B4 at $x \sim 0.6$.

Fig. 4.16 shows the specific volumes of the perfect structures considered for the calculations on the Nb-Al-N system. Apparently the B1 structure is the most densely packed for $x > 0$. Therefore, compressive stresses are likely to enhance the stability of the cubic phase. In pure NbN, however, the packing densities of the cubic B1 and the hexagonal B_i phases are almost
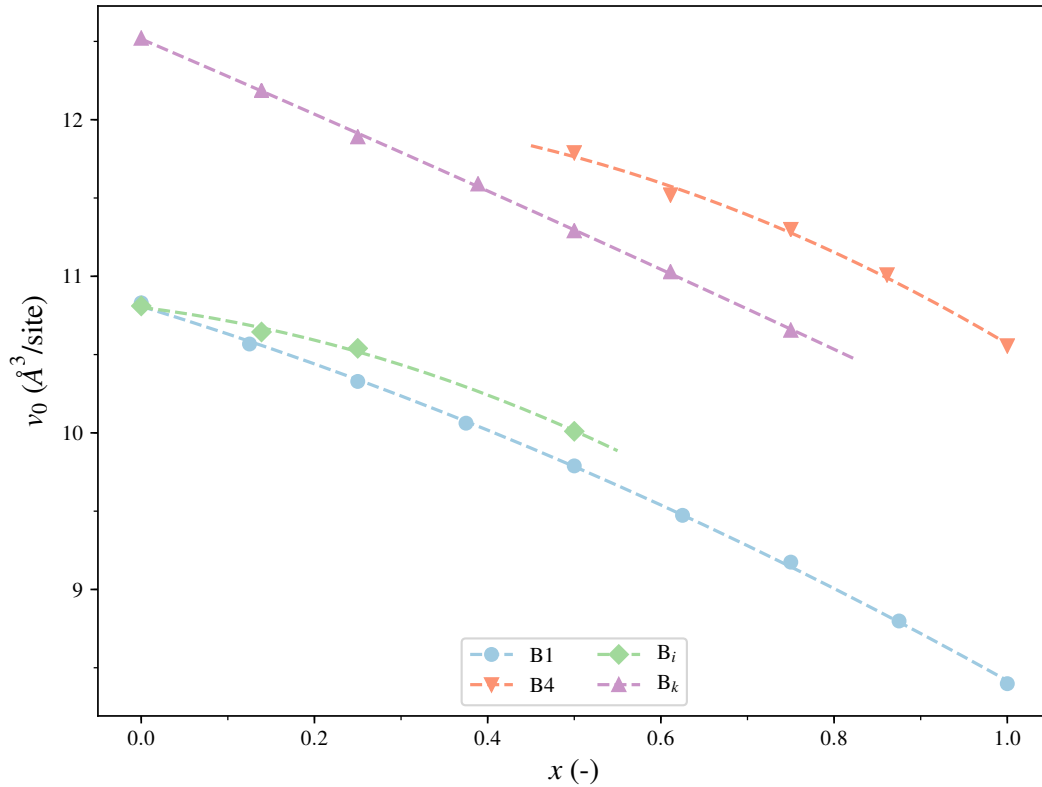
Figure 4.16: Specific volume, v_0, against x for the perfect structures considered in the Nb-Al-N system. The symbols represent the data from the DFT calculations and dashed lines are fitted second order polynomials.

the same and thus no substantial influence on the relative stability between the two phases can be expected.

The plots in Fig. 4.17 illustrate the influence of vacancies on the specific volume of the different structures. The volume per atomic site decreases when vacancies are included. This effect is generally stronger for metal vacancies than for nitrogen vacancies, although in the B4 phase, some of the cases containing metal vacancies are outliers from this trend.

The two graphs of Fig. 4.18 show the lattice constants of the B1 phase as a function of the aluminum content x, and the impact of the different types of defects on them. The dashed and dotted lines are the fitted Vegard-like functions of Eq. 4.9. The lattice parameters a of NbN and AlN as well as the fitted bowing parameters b are listed in Table 4.4. Similarly to the Ta-Al-N system, metal vacancies reduce the lattice constant, and the effect is stronger for Nb than for Al vacancies. Nitrogen vacancies reduce a_{NbN} too, but not as strong as metal vacancies. Table 4.5 lists some values of the lattice parameter of the cubic NbN phase calculated in the present study and collected from the literature.
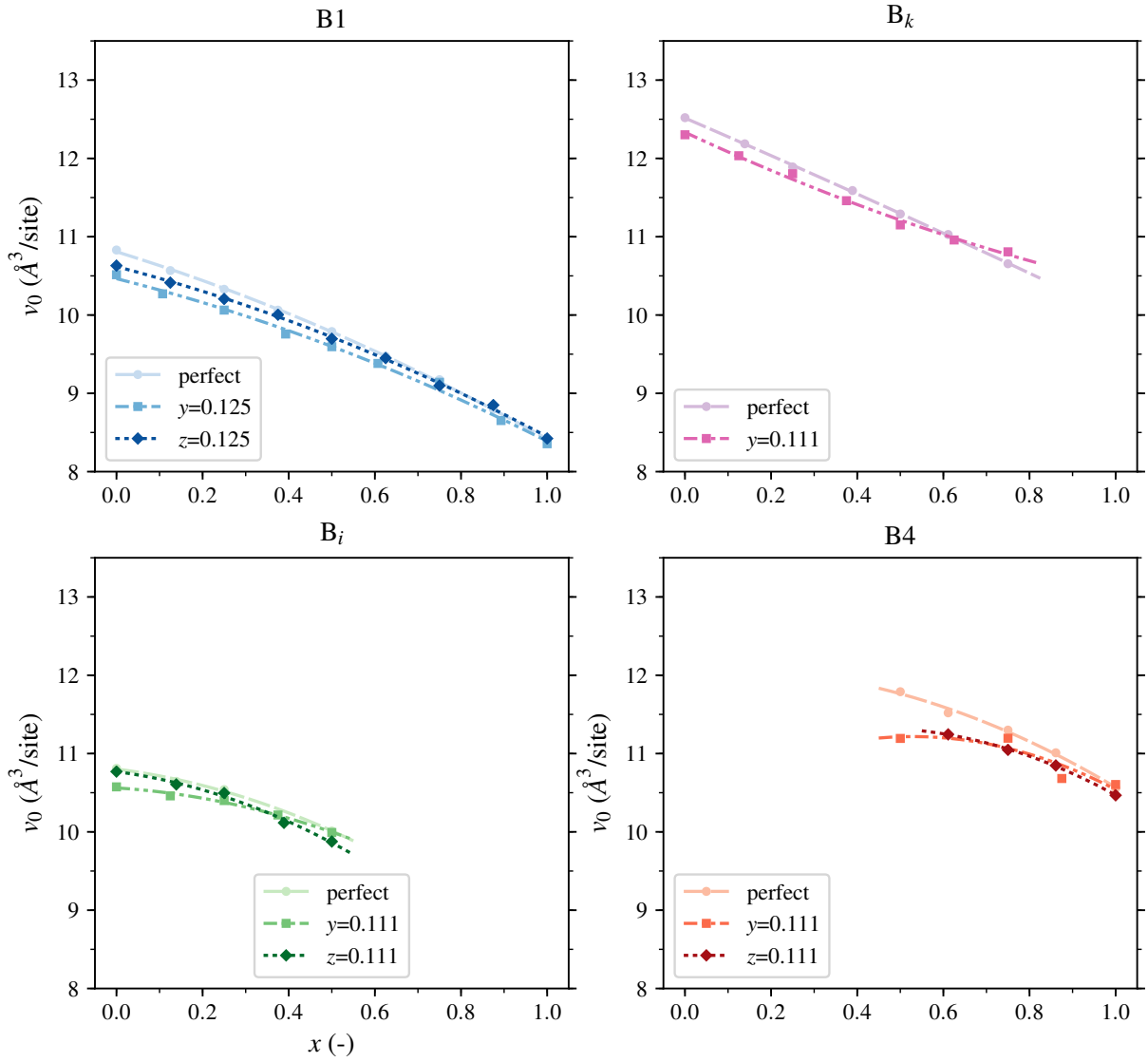
Figure 4.17: v_0 against x plots for the perfect configurations and different vacancy contents y and z of all considered phases in the Nb-Al-N system. The symbols represent the discrete v_0 data points while the dashed and dotted lines are the second order polynomials.
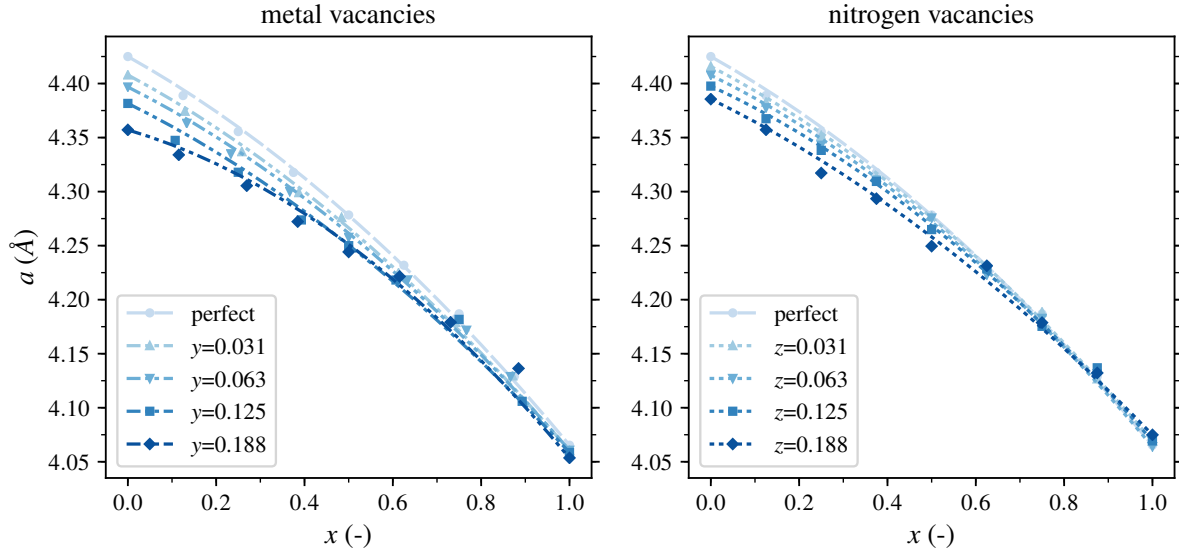
Figure 4.18: Lattice constants $a(x)$ of the cubic B1 phase and the influence of metal and nitrogen vacancies on them in the Nb-Al-N system. The symbols show the actual data and the dashed and dotted lines are fitted functions according to Eq. 4.9.

Table 4.4: Lattice constants and bowing parameters according to Eq. 4.9 of cubic B1 $(Nb_{1-x}Al_x)_{1-y}N$ and $Nb_{1-x}Al_xN_{1-z}$. All parameters are in Å.

	a_{NbN}	a_{AlN}	b
perfect	4.425	4.065	0.131
metal vacancies			
y			
0.031	4.408	4.060	0.129
0.063	4.397	4.061	0.131
0.125	4.382	4.059	0.122
0.188	4.357	4.054	0.183
nitrogen vacancies			
z			
0.031	4.416	4.069	0.134
0.063	4.408	4.064	0.145
0.125	4.398	4.069	0.140
0.188	4.386	4.075	0.111

Table 4.5: Lattice constants a of the cubic B1 phase found in literature. All values in Å.

structure	present	experimental	theoretical LDA	GGA
NbN	4.425	4.394[156], 4.385[157], 4.392[157]	4.36[149],	4.458[92], 4.399[158], 4.429[158],4.42[149]
$Nb_{0.938}N$	4.397	4.38[159]$(Nb_{0.934}N)$		
$Nb_{0.875}Al_{0.125}N$	4.389	4.36[89]$(Nb_{0.9}Al_{0.1}N)$		
$Nb_{0.750}Al_{0.250}N$	4.355	4.33[89], 4.31[92]$(Nb_{0.73}Al_{0.27}N)$		4.38[92]$(Nb_{0.73}Al_{0.27}N)$
$Nb_{0.5}Al_{0.5}N_{0.938}$	4.276	4.293[157]$(Nb_{0.5}Al_{0.5}N_{0.92})$		

Summary and Conclusion

Ab initio simulations on the quasi-binary transition metal nitride systems Ta-Al-N and Nb-Al-N were carried out with the aim to investigate the influence of vacancies on the alloys. The calculations spanned the whole compositional range of $Ta_{1-x}Al_x/Nb_{1-x}Al_x$ on the metal sublattice and included vacancies on either the metal or the nitrogen sublattice up to concentrations of 11 %. The supercells were created using the special quasi-random structure approach (SQS) considering pair interactions up to the 7th coordination shell. From the bare DFT results, the energy of formation, E_f, of the considered configurations was calculated and subsequently used to fit functions $E_f(x, y)$ and $E_f(x, z)$ in aluminum fraction x and metal or nitrogen vacancy content y or z, respectively.

Using these functions, phase diagrams were created by ranking the individual phases according to their E_f-values in the x-y and x-z space.

It was shown, that the presence of metal vacancies makes the energy of formation of the cubic B1 phase more negative in both systems over a wide range of aluminum fractions. E_f of the hexagonal B$_k$ structure is only weakly affected in the Ta-Al-N system and it increases in the Nb-Al-N system. In all other considered structures, metal vacancies lead to an increase in E_f. The combined effects caused by this defect type result in a strongly expanded range of Al/Ta and Al/Nb compositions for which the cubic B1 structure is the most stable one.

A similar trend is predicted also for nitrogen vacancies. The energy of formation of the B1 phase is decreased by their presence, although to a smaller extend than by the metallic ones. The other phases experience an increase in E_f. Again, the stability of the cubic phase is extended, but this extension is less pronounced.

Furthermore, all phases were characterized in terms of their specific volume and the lattice constants were calculated for the cubic phase. The obtained cubic lattice parameters agree well with values found in literature.

Cross-checking the phase stability with experiments also shows good agreement for the quasi-binary systems, and the accuracy of the predicted phase transitions could be increased by taking vacancies into account. Some discrepancies remain for pure TaN_{1-z} and NbN ma-

terials, as the single phase cubic materials found in experiments could not be proven to be caused solely by the influence of vacancies. While, without a doubt, they do play an important role in the stabilization of the B1 phase, additional factors have to be considered as well. The most important ones are diffusion kinetics and internal stresses. Future calculations considering those impacts thus promise to further increase the accuracy of DFT simulations in predicting the phase stability in the Ta-Al-N and the Nb-Al-N systems by bringing them closer towards reality.

Appendix

Python script used to create, assess and rank the supercells. The script can be used for ternary alloys, and the assessment follows the SQS approach respecting pair interactions on multiple coordination shells. For more details see Sections 2.2.2 and 2.2.3.

```python
1   import numpy as np
2   import os
3   import time
4   from numpy.linalg import norm
5   from numpy.random import randint
6   from os.path import realpath
7   from sys import argv
8
9
10  def __main__():
11      '''
12      something
13      '''
14
15      print '\nThis is a python 2 script for the generation and assessment of SQS-supercells'
16      print '————————————————————————————————————————————————————————————————————\n'
17
18      try:
19          if len(argv) == 5:
20              task = int(argv[1])
21              pathSettings, pathStructure, pathPositions = argv[2:]
22
23          elif len(argv) == 7:
24              task = int(argv[1])
25              pathSettings, pathStructure, pathPositions, pathSQSs, pathSROs = argv[2:]
26              assert pathSQSs and pathSROs in os.listdir(realpath(''))
27          else: raise AssertionError
28
29          assert task in [0,1,2]
30          assert pathSettings and pathStructure and pathPositions in os.listdir(realpath(''))
31
32      except AssertionError:
33          print "Who would use the script of death must answer me this questions three (
                  actually four), ere the SQSs he'll see!\n"
34          task = promptOptions( "What is Your quest?:\ngenerate SQS-supercells - 0\tanalyze
                  SQS-supercells - 1\tboth - 2", [0,1,2], strAnswerType='int' )
```

```
35          pathSettings = promptPath( 'What is the name of Your settings-file?' )
36          pathStructure= promptPath( 'What is the name of Your supercell structure-file?' )
37          pathPositions= promptPath( 'What is the name of Your atom positions-file?' )
38
39
40      distTol, shells, Aatoms, Batoms, structures, nrSQS = readSettings(pathSettings)
41      header, basis, latConst, atomPos = readSupercell(pathStructure,pathPositions)
42
43      print 'Distance tolerance: %f' % distTol
44      print 'Nr. of optimized shells: %i' % shells
45      print 'Atom-sites in the supercell: %i' % len(atomPos)
46      print 'Nr. of A-atoms in the supercell: %i' % Aatoms
47      print 'Nr. of B-atoms in the supercell: %i' % Batoms
48      print 'Nr. of C-atoms in the supercell: %i' % (len(atomPos)-Aatoms-Batoms)
49
50      cartAtomPos = [np.dot(pos,basis * latConst) for pos in atomPos]
51
52      if task == 0 :
53          generateSQS( basis*latConst, cartAtomPos, Aatoms, Batoms, shells, distTol,
                  structures)
54
55      elif task == 1 :
56          if 'SQSs' and 'SROs_AB' and 'SROs_AC' and 'SROs_BC' in os.listdir(realpath('')):
57              pathSQSs = 'SQSs'
58              pathSROsAB = 'SROs_AB'
59              pathSROsAC = 'SROs_AC'
60              pathSROsBC = 'SROs_BC'
61          else:
62              pathSQSs = promptPath( 'What is the name of Your SQS-file?' )
63              pathSROsAB = promptPath( 'What is the name of Your SRO-file for AB-SROs?' )
64              pathSROsAC = promptPath( 'What is the name of Your SRO-file for AC-SROs?' )
65              pathSROsBC = promptPath( 'What is the name of Your SRO-file for BC-SROs?' )
66
67          analyzeSQS( pathSQSs, pathSROsAB, pathSROsAC, pathSROsBC, header, latConst, basis,
                  atomPos, nrSQS )
68
69      else:
70          generateSQS( basis*latConst, cartAtomPos, Aatoms, Batoms, shells, distTol,
                  structures)
71
72          if 'SQSs' and 'SROs_AB' and 'SROs_AC' and 'SROs_BC' in os.listdir(realpath('')):
73              pathSQSs = 'SQSs'
74              pathSROsAB = 'SROs_AB'
75              pathSROsAC = 'SROs_AC'
76              pathSROsBC = 'SROs_BC'
77          else:
78              pathSQSs = promptPath( 'What is the name of Your SQS-file?' )
79              pathSROsAB = promptPath( 'What is the name of Your SRO-file for AB-SROs?' )
80              pathSROsAC = promptPath( 'What is the name of Your SRO-file for AC-SROs?' )
81              pathSROsBC = promptPath( 'What is the name of Your SRO-file for BC-SROs?' )
82
83          analyzeSQS( pathSQSs, pathSROsAB, pathSROsAC, pathSROsBC, header, latConst, basis,
                  atomPos, nrSQS )
84
85
86
87
88
89
```

```python
##############################################################################
#                              FUNCTIONS                                     #
##############################################################################

def generateSQS( arSupercellBasis , liSupercellPos , inNRAatoms, inNRBatoms, inNRshells ,
    flDistTolerance , inNRstructures = 100000   ):
    '''
    generates the special-quasirandom structure through several steps:

    1) expandes the supercell to the size 3x3x3 to mimic the periodic boundary conditions in
            the DFT-simulation
    2) assinges the atoms to the different shells according to their distance
    3) genreates random configurations for the binary system
    4) evaluates their short range order parameters
    '''
    # typecasting to list of 1-D arrays in case a 2-D array is handed over
    liSupercellPos = list( liSupercellPos )
    inSupercellSites = len( liSupercellPos )


    # 1) creating a supercell of the supercell to mimic the periodic BC of the DFT-
            clculation
    liPeriodicCellPos = []
    liPositionTags = []
    for a in [-1,0,1]:
        for b in [-1,0,1]:
            for c in [-1,0,1]:
                # list of atom-positions in the "periodic"-cell
                liPeriodicCellPos.extend( [pos + np.dot( np.array([a,b,c]) , arSupercellBasis
                        ) for pos in liSupercellPos] )
                # list of position-identifiers
                liPositionTags = liPositionTags + range(0,inSupercellSites)


    # creating a list of the existing distances
    liDistances = [ norm( pos - liSupercellPos[0] ) for pos in liPeriodicCellPos ]

    # sort
    liDistances.sort()

    # remove zero-entry
    liDistances.pop(0)

    # remove entries that differ less than the needed tolerance
    for i in xrange( len(liDistances)-1, 0, -1 ):
        if (liDistances[i]-liDistances[i-1]) <= flDistTolerance : liDistances.pop(i)

    # reduce the number of distances to number of shells
    inNRshells = min( len( liDistances ), inNRshells )
    liDistances = liDistances[:inNRshells]
    #print liDistances
    print 'The following shellsizes (next neighbour distances) will be used:'
    print liDistances


    # 2) - assigning the atoms to the different shells

    # for each shell
    diShells = {}
```

```python
145         liNRneighbours = []
146     for shellnumber in xrange( 0, inNRshells ):
147         shellatoms = []
148         print 'Creating shell %i with radius %f for every atom of the supercell' % (
                shellnumber+1, liDistances[shellnumber])
149
150         # the distance of every atom in the original supercell
151         summedNRneighbours = 0.
152         for atom in liSupercellPos:
153             shellNeighbours = []
154
155             # to every atom in the periodic cell is checked
156             for neighbour in xrange( 0, len(liPeriodicCellPos) ):
157
158                 # if the distance between them is equal to the shell-radius (within the
                        tolerance)
159                 vec = liPeriodicCellPos[neighbour] - atom
160                 if abs( np.sqrt( (vec*vec).sum(axis=0) ) - liDistances[shellnumber] ) <
                        flDistTolerance:
161                     # the position-tag of the neighbour-atom is stored in a list
162                     shellNeighbours.append( liPositionTags[neighbour] )
163
164             # after all possible neighbours at the current shell-radius for an atom are
                    checked, the neighbour-list is appended to the shell list
165             shellatoms.append( shellNeighbours )
166             summedNRneighbours += len( shellNeighbours )
167
168         # after all atoms in the supercell have a list of neighbours for the current shell-
                size they are stored in a dictionary
169         diShells.update( { shellnumber: shellatoms } )
170         liNRneighbours.append( summedNRneighbours/inSupercellSites )
171     print 'Average nr. of shell-neighbours:'
172     print liNRneighbours
173
174     # 3),4) - Generation of the SQS and calculation of their SRO
175
176     # lists for storing the data
177     SQSs = []
178     SROs_AB = []
179     SROs_AC = []
180     SROs_BC = []
181
182     print 'Generating special quasirandom structures.'
183     start = time.time()
184
185     xA = float(inNRAatoms)/inSupercellSites
186     xB = float(inNRBatoms)/inSupercellSites
187     xC = 1. - xA - xB
188
189     print 'xA = %f' % xA
190     print 'xB = %f' % xB
191     print 'xC = %f' % xC
192
193
194     for structure in xrange(0,inNRstructures):
195         if structure % 100 == 0: print 'processing %i of %i' % (structure+1,inNRstructures)
196
197         arSiteSpin = np.ones(inSupercellSites, int)*-1
198         Aatoms = 0
```

```
199                    Batoms = 0
200
201             # generating random occupation
202             # while the number of A-atoms in the structure is lower than the total one
203             while Aatoms < inNRAatoms :
204                 # a random position in the supercell is chosen and if this position is not
                          already
205                 # occupied by an A-atom another one is added there
206                 index = randint(0,inSupercellSites)
207                 if arSiteSpin[index] == -1:
208                     arSiteSpin[index] = 1
209                     Aatoms += 1
210
211             while Batoms < inNRBatoms :
212                 # a random position in the supercell is chosen and if this position is not
                          already
213                 # occupied by an A-atom another one is added there
214                 index = randint(0,inSupercellSites)
215                 if arSiteSpin[index] == -1:
216                     arSiteSpin[index] = 0
217                     Batoms += 1
218
219             SQSs.append(arSiteSpin)
220
221
222             # evaluating the SROs for every shell
223             structureSROs_AB = []
224             structureSROs_AC = []
225             structureSROs_BC = []
226             for key in diShells:
227                 N_AB = 0.
228                 N_BC = 0.
229                 N_AC = 0.
230                 shell = diShells[key]
231                 # looping through sites of the supercell
232                 for site in xrange(0,inSupercellSites):
233                     # and their neighbours in the current shell
234                     for neighbour in xrange(0,len(shell[site])):
235                         # counter for the bonding type with neighbour
236                         if arSiteSpin[site] == 1 and arSiteSpin[ shell[site][neighbour] ] == 0 :
                                  N_AB += 1.
237                         elif arSiteSpin[site] == 1 and arSiteSpin[ shell[site][neighbour] ] ==
                                  -1 : N_AC += 1.
238                         elif arSiteSpin[site] == 0 and arSiteSpin[ shell[site][neighbour] ] == 1
                                  : N_AB += 1.
239                         elif arSiteSpin[site] == 0 and arSiteSpin[ shell[site][neighbour] ] ==
                                  -1 : N_BC += 1.
240                         elif arSiteSpin[site] == -1 and arSiteSpin[ shell[site][neighbour] ] ==
                                  1 : N_AC += 1.
241                         elif arSiteSpin[site] == -1 and arSiteSpin[ shell[site][neighbour] ] ==
                                  0 : N_BC += 1.
242
243
244             # SRO = 1 - number fo A-B bonds in SC / number of A-B bonds in a totally random
                      alloy |(xi = Ni/N , factor 1/2 for double counting)
245             alpha_AB = 1. - (N_AB / (2 * xA * xB * inSupercellSites * liNRneighbours[key]))
246             alpha_AC = 1. - (N_AC / (2 * xA * xC * inSupercellSites * liNRneighbours[key]))
247             alpha_BC = 1. - (N_BC / (2 * xB * xC * inSupercellSites * liNRneighbours[key]))
248
```

```
249              structureSROs_AB.append(alpha_AB)
250              structureSROs_AC.append(alpha_AC)
251              structureSROs_BC.append(alpha_BC)
252
253          SROs_AB.append( np.array(structureSROs_AB) )
254          SROs_AC.append( np.array(structureSROs_AC) )
255          SROs_BC.append( np.array(structureSROs_BC) )
256
257      # storing
258      print 'Storing data.'
259
260      arSQSs = np.array( SQSs )
261
262      arSROs_AB = np.array( SROs_AB )
263      arSROs_AC = np.array( SROs_AC )
264      arSROs_BC = np.array( SROs_BC )
265
266      if 'SQSs' and 'SROs_AB' and 'SROs_AC' and 'SROs_BC' in os.listdir( realpath( '' ) ):
267          print "There are already SQS- and SRO-files in this directory."
268          mode = promptOptions( "Do you want to append or overwrite the existing files?\nPress
                 :\ta - to append\tw - to overwrite", ['a','w'], strAnswerType='str' )
269          sqs = open( 'SQSs', mode )
270          sro_ab = open( 'SROs_AB', mode )
271          sro_ac = open( 'SROs_AC', mode )
272          sro_bc = open( 'SROs_BC', mode )
273          np.savetxt( sqs, arSQSs, fmt = '%1i' )
274          np.savetxt( sro_ab, arSROs_AB, fmt = '%5f' )
275          np.savetxt( sro_ac, arSROs_AC, fmt = '%5f' )
276          np.savetxt( sro_bc, arSROs_BC, fmt = '%5f' )
277          sqs.close()
278          sro_ab.close()
279          sro_ac.close()
280          sro_bc.close()
281      else:
282          np.savetxt( 'SQSs', arSQSs, fmt = '%1i' )
283          np.savetxt( 'SROs_AB', arSROs_AB, fmt = '%5f' )
284          np.savetxt( 'SROs_AC', arSROs_AC, fmt = '%5f' )
285          np.savetxt( 'SROs_BC', arSROs_BC, fmt = '%5f' )
286      end = time.time()
287      print 'elapsed time: %f' % (end-start)
288
289 #- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -#
290
291 def analyzeSQS( stPathSQSs, stPathSROsAB, stPathSROsAC, stPathSROsBC, stOUTheader,
        flLatConst, arOUTbasis, liOUTatompos, inNRoutputSQSs=5 ):
292      '''
293      '''
294      try:
295          print 'Loading data.\n'
296          arSQSs = np.loadtxt( stPathSQSs, dtype = int )
297          arSROs_AB = np.loadtxt( stPathSROsAB, dtype = float )
298          arSROs_AC = np.loadtxt( stPathSROsAC, dtype = float )
299          arSROs_BC = np.loadtxt( stPathSROsBC, dtype = float )
300
301      except:
302          print 'Error while loading data.'
303          exit(0)
304
305      print 'Analyzing SROs.\n'
```

```
306        inNRshells = len( arSROs_AB[0] )
307
308        # array with weights of the different SROs
309        arWeights = np.array( [ (1.+2.*(inNRshells-i)/(inNRshells-1)) for i in range(1,
               inNRshells+1) ] )
310        print 'SRO-weights are:'
311        print '%f ' * len(arWeights) % tuple(arWeights) + '\n'
312        # array of weighted SROs
313        weightedSROs = np.sum( np.absolute(arSROs_AB) * arWeights, axis = 1 ) + np.sum( np.
               absolute(arSROs_AC) * arWeights, axis = 1 ) + np.sum( np.absolute(arSROs_BC) *
               arWeights, axis = 1 )
314        # array containing the indices of the lowest weighted SROs
315        arIndices = np.argpartition( weightedSROs, range(0,inNRoutputSQSs+1) )[:inNRoutputSQSs]
316
317        # the best values
318        liMinSROs = [weightedSROs[index] for index in arIndices]
319        liBestSROs_AB= [arSROs_AB[index] for index in arIndices]
320        liBestSROs_AC= [arSROs_AC[index] for index in arIndices]
321        liBestSROs_BC= [arSROs_BC[index] for index in arIndices]
322        liBestSQSs= [arSQSs[index] for index in arIndices]
323
324        print 'Creating output-files.\n'
325        for outfile in xrange(0, len(liBestSQSs) ):
326
327            fileObj = open( realpath( 'SQSsupercell%i.vasp' % (outfile+1) ), 'w' )
328            fileObj.write( stOUTheader + ' SQS%i (%f)\n' % (outfile+1, liMinSROs[outfile] ) )
329            fileObj.write( str(flLatConst)+ '\n' )
330
331            for ai in arOUTbasis:
332                fileObj.write( '%f %f %f\n' % tuple(ai) )
333
334            indexA = []
335            indexB = []
336            indexC = []
337            for index in xrange(0,len(liOUTatompos)):
338                if liBestSQSs[outfile][index] == 1: indexA.append(index)
339                elif liBestSQSs[outfile][index] == 0: indexB.append(index)
340                else: indexC.append(index)
341
342            fileObj.write( 'A B C\n' )
343            fileObj.write( '%i %i %i\n' % (len(indexA), len(indexB), len(indexC)) )
344            fileObj.write( 'Direct\n' )
345
346            for A in indexA: fileObj.write( '%.10f %.10f %.10f ! atom A\n' % tuple(liOUTatompos[
                   A]) )
347            for B in indexB: fileObj.write( '%.10f %.10f %.10f ! atom B\n' % tuple(liOUTatompos[
                   B]) )
348            for C in indexC: fileObj.write( '%.10f %.10f %.10f ! atom C\n' % tuple(liOUTatompos[
                   C]) )
349            fileObj.close()
350
351        print 'Done!\n'
352
353 #- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -#
354
355 def promptOptions( strInputQuestion, liAnswerOptions, strAnswerType='str' ):
356        '''
357        function to prompt input from the command line
358
```

```
359     Usage: input = promting( 'Question', [answer1,answer2,...], stAnswerType= type)
360            type is either string or integer.
361            To use integer-type answers give a list of integer-answer options and specify
                   strAnswerType = 'int'
362            ('i','Int','INT' and 'I' also recognized).
363            In any other case the answer-type is string and a list of character/string
                   options is needed.
364     '''
365     validAnswer = False
366     while not validAnswer:
367
368         try:
369             print strInputQuestion
370
371             if strAnswerType[0].lower() == 'i':
372                 answer = raw_input('> ')
373                 if 'holy grail' in answer.lower():
374                     print '\nHail to You Arthur, king of the britains!\n'
375                 else: answer = int(answer)
376
377             else: answer = raw_input('> ')
378
379             if answer in liAnswerOptions:
380                 validAnswer = True
381                 return answer
382
383             else: raise ValueError
384
385         except ValueError: print '\aNNNI! We want a valid input!\n'
386
387 #- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -#
388
389 def promptPath( strInputQuestion ):
390
391
392     validAnswer = False
393     while not validAnswer:
394         try:
395             print strInputQuestion
396             answer = raw_input('> ')
397
398             if answer in os.listdir( realpath( '' ) ):
399                 validAnswer = True
400                 return answer
401
402             else: raise ValueError
403
404         except ValueError: print '\aNNNI! We want valid input!\n'
405
406 #- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -#
407
408 def readSettings( strPathSettings ):
409     try:
410         settings = open( realpath(strPathSettings) )
411
412         distTol = float( settings.readline().split()[0] )
413         shells = int( settings.readline().split()[0] )
414         Aatoms = int( settings.readline().split()[0] )
415         Batoms = int( settings.readline().split()[0] )
```

```
416          structures = int( settings.readline().split()[0] )
417          nrSQSoutput = int( settings.readline().split()[0] )
418
419          settings.close()
420
421          return distTol, shells, Aatoms, Batoms, structures, nrSQSoutput
422
423      except (Exception, ValueError, IndexError) as Error:
424          print Error
425          print Error.args
426          print 'COULD NOT READ SETTINGS FILE!'
427          print 'You must return with a proper one or else You will never pass through this
                  script ... alive! An exampe-file is printed.\n'
428          exampleInputFile(0)
429          exit(0)
430
431
432 # - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -#
433
434 def readSupercell( strPathStructure, strPathAtomSites ):
435      try:
436          # read structure file
437          foStructure = open( realpath(strPathStructure) )
438          strucHeader  = foStructure.readline().strip()
439          flLatConst = float( foStructure.readline() )
440
441          liBasis = []
442          for i in [0,1,2]:
443              liBasis.append( foStructure.readline().split()[:3] )
444          foStructure.close()
445          arBasis = np.array( liBasis, float )
446
447          # read atom-site file
448          foPositions = open( realpath(strPathAtomSites) )
449          posHeader  = foPositions.readline().strip()
450          nrIons = int( foPositions.readline() )
451
452          liAtomPos = []
453          for line in foPositions:
454              direct = np.array( line.split()[:3], float )
455              liAtomPos.append( direct )
456          foPositions.close()
457          #arSites = np.array( liSites, float )
458
459
460          # small consistency checks of the input
461          if strucHeader != posHeader: print 'BE WARNED: the two files have different headers!
                  They might belong to different structures.'
462
463          if len(liAtomPos) > nrIons:
464              liAtomPos = liAtomPos[:(nrIons+1)]
465              print 'BE WARNED: number of sites and number of coordinate-sets do not fit! Only
                      first %i coordinate sets used!' % NrSites
466          elif len(liAtomPos) < nrIons: raise ValueError('Number of sites and number of
                  coordinate-sets do not fit!')
467
468          # ATTENTION: list of position-arrays is returned not array
469          return strucHeader, arBasis, flLatConst, liAtomPos
470
```

```
471        except (Exception, ValueError, IndexError) as Error:
472            print Error
473            print Error.args
474            print 'COULD NOT READ SUPERCELL FILES!'
475            print 'We want some proper ones! Some that look nice. And not too expensive. Example
                       -files are printed.\n'
476            exampleInputFile(1)
477            exit(0)
478
479 #  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  - -#
480
481 def exampleInputFile( inType ):
482     try:
483         if inType == 0:
484             ex = open( realpath( 'exampleSettingsFile' ), 'w' )
485             ex.write('tolarance distance for shell classification (float)\nnumber of shells
                       to be considered (int)\n')
486             ex.write('number of atoms of type A (int)\nnumber of atoms of type B (int)\
                       nnumber of structures to be generated (int)\n')
487             ex.write('number of SQS-supercells to be written out (int)')
488             ex.close()
489
490         elif inType == 1:
491             ex1 = open( realpath( 'exampleSupercellStructureFile'), 'w' )
492             ex1.write( 'name/description (string)\nlattice constant (float)\n' )
493             ex1.write( 'a_1 a_2 a_3 (komponents of the basis-vectors separated by space,
                       format float)\n')
494             ex1.write( 'b_1 b_2 b_3 (must have the right length according to the used
                       supercell)\n')
495             ex1.write( 'c_1 c_2 c_3 (eg. 2x basis vectors of the unit cell for a 2x2x2
                       supercell)\n')
496             ex1.close()
497
498             ex2 = open( realpath( 'exampleSupercellAtompositionFile'), 'w' )
499             ex2.write( 'name/description (string)\nnumber of atoms in the supercell (int)\n'
                       )
500             ex2.write( 'x_1 y_1 z_1 (direct coordinates of the first atom separated by space
                       , format float)\n')
501             ex2.write( 'x_2 y_2 z_2 (direct coordinates of the second atom)\n...')
502             ex2.close()
503         else: raise ValueError
504     except: print 'Could not print example-input files'
505 #  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  -  - -#
506
507 if __name__ == '__main__':
508     __main__()
```

ternarySQS.py

Bibliography

[1] B. Alling, A. V. Ruban, A. Karimi, O. E. Peil, S. I. Simak, L. Hultman, and I. A. Abrikosov, Phys. Rev. B **75**, 045123 (2007).

[2] H. Euchner and P. Mayrhofer, Surface and Coatings Technology **275**, 214 (2015).

[3] N. Koutná, D. Holec, O. Svoboda, F. F. Klimashin, and P. H. Mayrhofer, Journal of Physics D: Applied Physics **49**, 375303 (2016).

[4] H. O. Pierson, *Handbook of Refractory Carbides and Nitrides*, edited by H. O. Pierson (William Andrew Publishing, Westwood, NJ, 1996).

[5] F. A. Cotton, G. Wilkinson, C. A. Murillo, and M. Bochmann, *Advanced Inorganic Chemistry*, 6th ed. (Wiley, New York, 1999).

[6] C. Janiak, T. Klapötke, H.-J. Meyer, and R. Alsfasser, *Moderne Anorganische Chemie*, 3rd ed., edited by E. Riedel (De Gruyter, Berlin, Boston, 2008).

[7] S. Oyama, *The Chemistry of Transition Metal Carbides and Nitrides*, edited by S. Oyama (Springer Netherlands, 1996).

[8] L. Toth, *Transition Metal Carbides and Nitrides*, Refractory materials (Academic Press, 1971).

[9] W. Lengauer, "Transition metal carbides, nitrides, and carbonitrides," in *Handbook of Ceramic Hard Materials* (Wiley-VCH Verlag GmbH, 2008) pp. 202–252.

[10] R. Rachbauer, D. Holec, M. Lattemann, L. Hultman, and P. H. Mayrhofer, International Journal of Materials Research **102**, 735 (2011).

[11] D. Sangiovanni, L. Hultman, and V. Chirita, Acta Materialia **59**, 2121 (2011).

[12] P. Ettmayer and W. Lengauer, "Nitrides," in *Ullmann's Encyclopedia of Industrial Chemistry* (Wiley-VCH Verlag GmbH & Co. KGaA, 2000).

[13] V. A. Gubanov, A. L. Ivanovsky, and V. P. Zhukov, *Electronic Structure of Refractory Carbides and Nitrides:* (Cambridge University Press, 1994).

[14] W. Weißbach, *Werkstoffkunde - Strukturen, Eigenschaften, Prüfung*, 18th ed. (Vieweg+Teubner Verlag, 2012).

[15] M. Berg, C. Budtz-Jørgensen, H. Reitz, K. Schweitz, J. Chevallier, P. Kringhøj, and J. Bøttiger, Surface and Coatings Technology **124**, 25 (2000).

[16] W. S. Williams, JOM **49**, 38 (1997).

[17] L. Hultman, Vacuum **57**, 1 (2000).

[18] P. H. Mayrhofer, *Lecture Notes of 308.875-Surface Technology* (Institute of Materials Science and Technology, TU Wien, Getreidemarkt 9, 1060 Wien, Austria, 2015).

[19] J. Vetter, Surface and Coatings Technology **76**, 719 (1995).

[20] K. Bobzin, E. Lugscheider, R. Nickel, and P. Immich, Materialwissenschaft und Werkstofftechnik **37**, 833 (2006).

[21] A. Reiter, V. Derflinger, B. Hanselmann, T. Bachmann, and B. Sartory, Surface and Coatings Technology **200**, 2114 (2005).

[22] W. Münz, Journal of Vacuum Science & Technology A: Vacuum, Surfaces, and Films **4**, 2717 (1986).

[23] H. Holleck, Surface and Coatings Technology **36**, 151 (1988).

[24] K. Kutschej, P. Mayrhofer, M. Kathrein, P. Polcik, R. Tessadri, and C. Mitterer, Surface and Coatings Technology **200**, 2358 (2005).

[25] F. Vaz, L. Rebouta, M. Andritschky, M. da Silva, and J. Soares, Journal of the European Ceramic Society **17**, 1971 (1997).

[26] P. H. Mayrhofer, A. Hörling, L. Karlsson, J. Sjölén, T. Larsson, C. Mitterer, and L. Hultman, Applied Physics Letters **83**, 2049 (2003).

[27] A. Hörling, L. Hultman, M. Oden, J. Sjölén, and L. Karlsson, Surface and Coatings Technology **191**, 384 (2005).

[28] D. Holec, R. Rachbauer, L. Chen, L. Wang, D. Luef, and P. H. Mayrhofer, Surface and Coatings Technology **206**, 1698 (2011).

[29] K. Kutschej, N. Fateh, P. Mayrhofer, M. Kathrein, P. Polcik, and C. Mitterer, Surface and Coatings Technology **200**, 113 (2005).

[30] R. Rachbauer, D. Holec, and P. H. Mayrhofer, Applied Physics Letters **97**, 151901 (2010).

[31] F. Rovere, D. Music, J. Schneider, and P. Mayrhofer, Acta Materialia **58**, 2708 (2010).

[32] C. Koller, R. Hollerweger, C. Sabitzer, R. Rachbauer, S. Kolozsvári, J. Paulitsch, and P. Mayrhofer, Surface and Coatings Technology **259, Part C**, 599 (2014).

[33] C. Koller, R. Hollerweger, R. Rachbauer, S. Kolozsvári, J. Paulitsch, and P. Mayrhofer, Surface and Coatings Technology **283**, 89 (2015).

[34] P. Blaha, J. Redinger, and K. Schwarz, Phys. Rev. B **31**, 2316 (1985).

[35] D. Ellis, *Density Functional Theory of Molecules, Clusters, and Solids*, Understanding Chemical Reactivity (Springer Netherlands, 1995).

[36] L. Yu, C. Stampfl, D. Marshall, T. Eshrich, V. Narayanan, J. M. Rowell, N. Newman, and A. J. Freeman, Phys. Rev. B **65**, 245110 (2002).

[37] C. Stampfl and A. J. Freeman, Phys. Rev. B **71**, 024111 (2005).

[38] C. Stampfl and A. J. Freeman, Phys. Rev. B **67**, 064108 (2003).

[39] M. Magnuson, M. Mattesini, C. Höglund, J. Birch, and L. Hultman, Phys. Rev. B **80**, 155105 (2009).

[40] *Brevier technische Keramik*, Verband der Keramischen Industrie e.V. and Informationszentrum Technische Keramik, 4th ed. (2003).

[41] A. Kranzmann, in *Concise Encyclopedia of Advanced Ceramic Materials*, edited by R. Brook (Pergamon, Oxford, 1991) pp. 8 – 9.

[42] C. Stampfl and C. G. Van de Walle, Phys. Rev. B **59**, 5521 (1999).

[43] O. Madelung, U. Rössler, and M. Schulz, eds., "Aluminum nitride, crystal structure, lattice parameters, thermal expansion: Datasheet," in *Landolt-Börnstein - Group III Condensed Matter Volume 41: Group IV Elements, IV-IV and III-V Compounds in SpringerMaterials* (Springer-Verlag Berlin Heidelberg) copyright 2002 Springer-Verlag Berlin Heidelberg.

[44] M. Choi, C. Dubourdieu, A. J. Kellock, K. L. Lee, R. A. Haight, A. Pyzyna, M. M. Frank, A. A. Demkov, and V. Narayanan, Journal of Vacuum Science & Technology B, Nanotechnology and Microelectronics: Materials, Processing, Measurement, and Phenomena **32**, 051202 (2014).

[45] T. Oku, E. Kawakami, M. Uekubo, K. Takahiro, S. Yamaguchi, and M. Murakami, Applied Surface Science **99**, 265 (1996).

[46] J. Nazon, B. Fraisse, J. Sarradin, S. Fries, J. Tedenac, and N. Fréty, Applied Surface Science **254**, 5670 (2008).

[47] S. M. Rossnagel and H. Kim, Journal of Vacuum Science & Technology B: Microelectronics and Nanometer Structures Processing, Measurement, and Phenomena **21**, 2550 (2003).

[48] W.-H. Lee, J.-C. Lin, and C. Lee, Materials Chemistry and Physics **68**, 266 (2001).

[49] T. Riekkinen, J. Molarius, T. Laurila, A. Nurmela, I. Suni, and J. Kivilahti, Microelectronic Engineering **64**, 289 (2002), {MAM2002}.

[50] A. Mellberg, S. P. Nicols, N. Rorsman, and H. Zirath, Electrochemical and Solid-State Letters **7**, G261 (2004).

[51] T. Lee, K. Watson, F. Chen, J. Gill, D. Harmon, T. Sullivan, and B. Li, in *2004 IEEE International Reliability Physics Symposium. Proceedings* (2004) pp. 502–508.

[52] L. Gladczuk, A. Patel, J. D. Demaree, and M. Sosnowski, Thin Solid Films **476**, 295 (2005).

[53] S. Kim and B. Cha, Thin Solid Films **475**, 202 (2005), asian-European International Conference on Plasma Surface Engineering 2003 Proceedings of the 4th Asian-European International Conference on Plasma Surface Engineering.

[54] H. Yusa, F. Kawamura, T. Taniguchi, N. Hirao, Y. Ohishi, and T. Kikegawa, Journal of Applied Physics **115**, 103520 (2014).

[55] G. Lee, H. Kim, H. Choi, and J. Lee, Surface and Coatings Technology **201**, 5207 (2007), proceedings of the Fifth Asian-European International Conference on Plasma Surface Engineering (AEPSE) 2005.

[56] N. Terao, Japanese Journal of Applied Physics **10**, 248 (1971).

[57] C.-S. Shin, D. Gall, Y.-W. Kim, P. Desjardins, I. Petrov, J. E. Greene, M. Odén, and L. Hultman, Journal of Applied Physics **90**, 2879 (2001).

[58] B. Predel, "Nitrogen - tantalum: Datasheet," in *Landolt-Börnstein - Group IV Physical Chemistry Volume 5H in SpringerMaterials*, edited by O. Madelung (Springer-Verlag Berlin Heidelberg) copyright 1997 Springer-Verlag Berlin Heidelberg.

[59] K. Frisk, Journal of Alloys and Compounds **278**, 216 (1998).

[60] H. Wiesenberger, W. Lengauer, and P. Ettmayer, Acta Materialia **46**, 651 (1998).

[61] H. Marihart, *Structure, phase evolution and mechanical properties of HiPIMS, pulsed DC and DC sputtered Ta-N films*, Master's thesis, Montanuniversität Leoben (2015).

[62] J. Gatterer, G. Dufek, P. Ettmayer, and R. Kieffer, Monatshefte für Chemie - Chemical Monthly **106**, 1137 (1975).

[63] G. Jouve, C. Séverac, and S. Cantacuzène, Thin Solid Films **287**, 146 (1996).

[64] N. Hayashi, I. Murzin, I. Sakamoto, and M. Ohkubo, Thin Solid Films **259**, 146 (1995).

[65] M. Nakanishi, Y. Tsuchiya, N. Nakayama, K. Yoshimura, K. Kosuge, S. Nagata, and S. Yamaguchi, Physica C: Superconductivity **253**, 33 (1995).

[66] A. B. Kaul, S. R. Whiteley, T. V. Duzer, L. Yu, N. Newman, and J. M. Rowell, Applied Physics Letters **78**, 99 (2001).

[67] Z. Wang, A. Kawakami, and Y. Uzawa, Physica C: Superconductivity and its Applications **282**, 2465 (1997).

[68] M. Merker, C. Bohn, M. Völlinger, K. Ilin, and M. Siegel, IEEE Transactions on Applied Superconductivity **27**, 1 (2017).

[69] Y. Nakamura, H. Terai, K. Inomata, T. Yamamoto, W. Qiu, and Z. Wang, Applied Physics Letters **99**, 212502 (2011).

[70] J. M. Murduck, J. Vicent, I. K. Schuller, and J. B. Ketterson, Journal of Applied Physics **62**, 4216 (1987).

[71] Y. Gotoh, M. Nagao, T. Ura, H. Tsuji, and J. Ishikawa, Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms **148**, 925 (1999).

[72] P. Alén, M. Ritala, K. Arstila, J. Keinonen, and M. Leskelä, Thin Solid Films **491**, 235 (2005).

[73] J. Manaud, A. Poulon, S. Gomez, and Y. L. Petitcorps, Surface and Coatings Technology **202**, 222 (2007).

[74] M. Larsson, P. Hollman, P. Hedenqvist, S. Hogmark, U. Wahlström, and L. Hultman, Surface and Coatings Technology **86**, 351 (1996).

[75] K. Rutherford, P. Hatto, C. Davies, and I. Hutchings, Surface and Coatings Technology **86**, 472 (1996).

[76] V. Zhitomirsky, I. Grimberg, L. Rapoport, N. Travitzky, R. Boxman, S. Goldsmith, A. Raihel, I. Lapsker, and B. Weiss, Thin Solid Films **326**, 134 (1998).

[77] G. Fontalvo, V. Terziyska, and C. Mitterer, Surface and Coatings Technology **202**, 1017 (2007), 34th International Conference and Metallurgical Coatings and Thin Films (ICMCTF 2007).

[78] G. Brauer, Journal of the Less Common Metals **2**, 131 (1960).

[79] G. Oya and Y. Onodera, Journal of Applied Physics **47**, 2833 (1976).

[80] G. Brauer and R. Esselborn, Zeitschrift für anorganische und allgemeine Chemie **309**, 151 (1961).

[81] C. Sandu, M. Benkahoul, M. Parlinska-Wojtan, R. Sanjinés, and F. Lévy, Surface and Coatings Technology **200**, 6544 (2006), eMRS 2005 Symposium on Protective Coatings and Thin Films.

[82] M. Benkahoul, E. Martinez, A. Karimi, R. Sanjinés, and F. Lévy, Surface and Coatings Technology **180–181**, 178 (2004), proceedings of Symposium G on Protective Coatings and Thin Films-03, of the E-MRS 2003 Spring Conference.

[83] N. Cansever, M. Danisman, and K. Kazmanli, Surface and Coatings Technology **202**, 5919 (2008).

[84] C. Koller, A. Kirnbauer, R. Rachbauer, S. Kolozsvári, and P. Mayrhofer, Scripta Materialia **113**, 75 (2016).

[85] Y.-I. Chen, J.-H. Lin, and C.-C. Chou, Surface and Coatings Technology **303, Part A**, 41 (2016), 2015 International Thin Films Conference [TACT 2015].

[86] Z. Zhang, *Impact of Al on structure and mechanical properties of NbN and TaN*, Master's thesis, Montanuniversität Leoben (2011).

[87] D. Holec, L. Zhou, R. Rachbauer, and P. H. Mayrhofer, "Alloy based design from first principles: An application to functional hard coatings," in *Density Functional Theory: Principles, Applications and Analysis* (2013) pp. 259–284.

[88] R. Zhang and S. Veprek, Acta Materialia **57**, 2259 (2009).

[89] T. Selinder, D. Miller, K. Gray, M. Sardela, and L. Hultman, Vacuum **46**, 1401 (1995).

[90] H. C. Barshilia, B. Deepthi, K. Rajam, K. P. Bhatti, and S. Chaudhary, Journal of Materials Research **23**, 1258–1268 (2008).

[91] R. Franz, M. Lechthaler, C. Polzer, and C. Mitterer, Surface and Coatings Technology **204**, 2447 (2010).

[92] D. Holec, R. Franz, P. H. Mayrhofer, and C. Mitterer, Journal of Physics D: Applied Physics **43**, 145403 (2010).

[93] K. Momma and F. Izumi, Journal of Applied Crystallography **44**, 1272 (2011).

[94] J. G. Lee, *Computational Materials Science: An Introduction*, 2nd ed. (CRC Press, Taylor & Francis Group, 6000 Broken Sound Parkway NW, Suite 300 Boca Raton, FL 33487-2742, 2012).

[95] S. Cottenier, *Density Functional Theory and the family of (L)APW-methods: a step-by-step introduction*, 2nd ed. (Center for Molecular Modeling(CMM) & Department of Materials Science and Engineering (DMSE), Ghent University, Technologiepark 903, BE-9052 Zwijnaarde, Belgium, 2002).

[96] D. Holec, *Lecture 308.882: Atomistic Materials Modelling* (Institute of Materials Science and Technology, TU Wien, Getreidemarkt 9, 1060 Wien, Austria, 2015).

[97] E. Schrödinger, Annalen der Physik **384**, 361 (1926).

[98] M. Born and R. Oppenheimer, Annalen der Physik **389**, 457 (1927).

[99] D. Holec, *Multi-scale modelling of III-nitrides: from dislocations to the electronic structure*, Ph.D. thesis, University of Cambridge, Department of Materials Science and Metallurgy (2008).

[100] L. H. Thomas, Mathematical Proceedings of the Cambridge Philosophical Society **23**, 542–548 (1927).

[101] E. Fermi, Zeitschrift für Physik **48**, 73 (1928).

[102] P. Hohenberg and W. Kohn, Phys. Rev. **136**, B864 (1964).

[103] W. Kohn, *Nobel Lecture, January 28, 1999: Electronic Structure of Matter - Wave Functions and Density Functionals* (The Nobel Foundation).

[104] W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965).

[105] K. Burke and friends, *The ABC of DFT* (Department of Chemistry, University of California, Irvine, CA 92697, 2007).

[106] J. P. Perdew and Y. Wang, Phys. Rev. B **45**, 13244 (1992).

[107] J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. **77**, 3865 (1996).

[108] G. Kresse and D. Joubert, Phys. Rev. B **59**, 1758 (1999).

[109] Y. Wang and J. P. Perdew, Phys. Rev. B **44**, 13298 (1991).

[110] J. R. Chelikowsky, L. Kronik, I. Vasiliev, M. Jain, and Y. Saad, in *Special Volume, Computational Chemistry*, Handbook of Numerical Analysis, Vol. 10 (Elsevier, 2003) pp. 613 – 637.

[111] D. R. Hamann, M. Schlüter, and C. Chiang, Phys. Rev. Lett. **43**, 1494 (1979).

[112] D. Vanderbilt, Phys. Rev. B **41**, 7892 (1990).

[113] P. E. Blöchl, Phys. Rev. B **50**, 17953 (1994).

[114] P. E. Blöchl, C. J. Först, and J. Schimpl, arXiv:cond-mat/0201015v2 [cond-mat.mtrl-sci] (2002).

[115] C. Rostgaard, arXiv:0910.1921v2 [cond-mat.mtrl-sci] (2009).

[116] R. Gross and A. Marx, *Festkörperphysik*, 2nd ed. (De Gruyter Oldenbourg, Berlin, Boston, 2016).

[117] F. Bloch, Zeitschrift für Physik **52**, 555 (1929).

[118] A. Zunger, S.-H. Wei, L. G. Ferreira, and J. E. Bernard, Phys. Rev. Lett. **65**, 353 (1990).

[119] S.-H. Wei, L. G. Ferreira, J. E. Bernard, and A. Zunger, Phys. Rev. B **42**, 9622 (1990).

[120] J. M. Cowley, Phys. Rev. **77**, 669 (1950).

[121] G. Kresse, M. Marsman, and J. Furthmüller, *VASP the GUIDE*, Sensengasse 8/12, A-1090 Wien, Austria (2016).

[122] G. Kresse and J. Furthmüller, Computational Materials Science **6**, 15 (1996).

[123] G. Kresse and J. Furthmüller, Phys. Rev. B **54**, 11169 (1996).

[124] G. Kresse, M. Marsman, and J. Furthmüller, *VASP Workshop Lectures* (Computational Materials Physics, Faculty of Physics, Universität Wien, Sensengasse 8/12, A-1090 Wien, Austria).

[125] H. J. Monkhorst and J. D. Pack, Phys. Rev. B **13**, 5188 (1976).

[126] S. J. Clark, S. M. D., C. J. Pickard, P. J. Hasnip, M. J. Probert, K. Refson, and M. C. Payne, *CASTEP Documentation: Theory in CASTEP*, BIOVIA, Dassault Systèmes, 2014 (About CASTEP: First principles methods using CASTEP, Zeitschrift fuer Kristallographie, 220 (5-6), 567-570 (2005)).

[127] E. R. Davidson, Journal of Computational Physics **17**, 87 (1975).

[128] E. Davidson, in *Methods in Computational Molecular Physics, NATO Advanced Study Institute*, Series C: Mathematical and Physical Sciences, Vol. 113, edited by G. Diercksen and S. Wilson (New York, 1984).

[129] J. A. Hirschfeld, *Ab initio investigation of ground-states and ionic motion in particular in zirconia-based solid-oxide electrolytes*, Ph.D. thesis, Universität Duisburg-Essen, Fakultät für Physik (2012).

[130] P. Pulay, Chemical Physics Letters **73**, 393 (1980).

[131] D. M. Wood and A. Zunger, Journal of Physics A: Mathematical and General **18**, 1343 (1985).

[132] H. Hellmann, *Einführung in die Quantenchemie* (F. Deuticke, Leipzig, 1937).

[133] R. P. Feynman, Phys. Rev. **56**, 340 (1939).

[134] N. D. Mermin, Phys. Rev. **137**, A1441 (1965).

[135] M. Methfessel and A. T. Paxton, Phys. Rev. B **40**, 3616 (1989).

[136] C.-S. Shin, D. Gall, P. Desjardins, A. Vailionis, H. Kim, I. Petrov, J. E. Greene, and M. Odén, Applied Physics Letters **75**, 3808 (1999).

[137] H. Kim, C. Lavoie, M. Copel, V. Narayanan, D.-G. Park, and S. M. Rossnagel, Journal of Applied Physics **95**, 5848 (2004).

[138] M. H. Tsai, S. C. Sun, C. E. Tsai, S. H. Chuang, and H. T. Chiu, Journal of Applied Physics **79**, 6932 (1996).

[139] Y. Y. Wu, A. Kohn, and M. Eizenberg, Journal of Applied Physics **95**, 6167 (2004).

[140] P. H. Mayrhofer, C. Mitterer, L. Hultman, and H. Clemens, Progress in Materials Science **51**, 1032 (2006).

[141] M. Grumski, P. P. Dholabhai, and J. B. Adams, Acta Materialia **61**, 3799 (2013).

[142] N. Koutná, *Impact of point defects on stability and mechanical properties of nitrides and oxides*, Master's thesis, Masaryk University, Faculty of Science, Department of Condensed Matter Physics (2016).

[143] L. Vegard, Zeitschrift für Physik **5**, 17 (1921).

[144] A. R. Denton and N. W. Ashcroft, Phys. Rev. A **43**, 3161 (1991).

[145] D. Holec, F. Rovere, P. H. Mayrhofer, and P. B. Barna, Scripta Materialia **62**, 349 (2010).

[146] P. H. Mayrhofer, *Lecture Notes of 308.862-Materials Science of Metallic Materials* (Institute of Materials Science and Technology, TU Wien, Getreidemarkt 9, 1060 Wien, Austria, 2016).

[147] *Powder Diffraction File 01-079-5712*, International Center for Diffraction Data (2014).

[148] J. Chang, G.-P. Zhao, X.-L. Zhou, K. Liu, and L.-Y. Lu, Journal of Applied Physics **112**, 083519 (2012).

[149] C. Stampfl, W. Mannstadt, R. Asahi, and A. J. Freeman, Phys. Rev. B **63**, 155106 (2001).

[150] *Powder Diffraction File 00-049-1284*, International Center for Diffraction Data (2007).

[151] *Powder Diffraction File 00-046-1200*, International Center for Diffraction Data (2014).

[152] N. E. Christensen and I. Gorczyca, Phys. Rev. B **50**, 4397 (1994).

[153] S. Saib and N. Bouarissa, The European Physical Journal B - Condensed Matter and Complex Systems **47**, 379 (2005).

[154] D. Holec and P. H. Mayrhofer, Scripta Materialia **67**, 760 (2012).

[155] E. F. Skelton, M. R. Skokan, and E. Dukauskas, Journal of Applied Crystallography **14**, 51 (1981).

[156] *Powder Diffraction File 01-071-0162*, International Center for Diffraction Data (2007).

[157] M. Miki, T. Yamasaki, and Y. Ogino, Materials Transactions, JIM **33**, 839 (1992).

[158] M. Chauhan and D. C. Gupta, Computational Materials Science **90**, 182 (2014).

[159] M. Benkahoul, *Niobium nitride based thin films deposited by DC reactive magnetron sputtering: NbN, NbSiN and NbAlN*, Master's thesis, Ecole Polytechnique Fédérale de Lausanne (2005).