Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Machine Learning for Novel Ternary Materials Discovery

by

Hugo Alexandre Rossignol

A thesis submitted for the degree of

Doctor of Philosophy

School of Physics
Trinity College Dublin

January 2024

Declaration

I, Hugo Alexandre Rossignol, hereby declare that this thesis has not been submitted as an exercise for a degree at this or any other university.

It comprises work performed entirely by myself during the course of my Ph. D. studies at Trinity College Dublin. I was involved in a number of collaborations and, where it is appropriate, my collaborators are acknowledged for their contributions.

I agree to deposit this thesis in the University's open-access institutional repository or allow the Library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

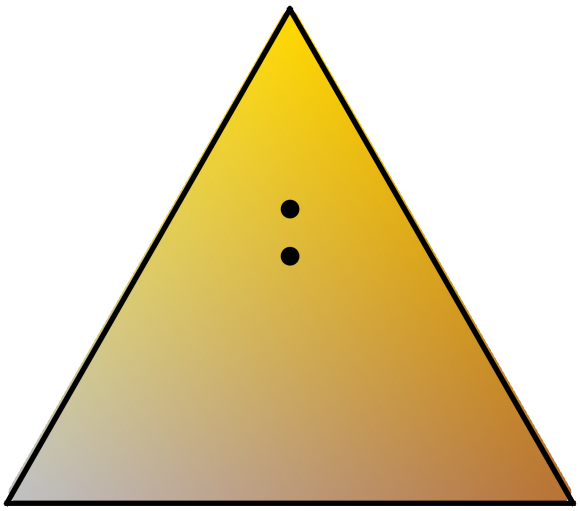
I consent to the examiner retaining a copy of the thesis beyond the examining period, should they so wish.

_____

Hugo Alexandre Rossignol

Quand t'as l'desert à traverser, il y a rien à faire sauf d'avancer.

– Orelsan



Abstract

First principles codes based on DFT are now sufficiently accurate and efficient that they can be used in the design of novel materials with specifically selected properties. Doing so requires checking whether a compound considered is chemically stable or not, which in turn involves constructing an appropriate convex hull. While such an approach is valid for identifying new, synthesizable materials, it demands numerous DFT calculations to ensure reliability. This work aims at accelerating this task through the use of machine learning interatomic potentials (MLIAPs) as screening agents. The process is facilitated by using pre-existing data on large material repositories to form the training sets of the models. Owing to the relative wealth in the number of binary systems compared to the ternaries, the former provide an ideal and extensive database for this training. In contrast, the space of ternaries, being only sparsely examined, forms fertile ground for exploration.

In a first part of this work, an ensemble of spectral neighbour analysis potentials (SNAPs) is trained on binary data of the Ag-Au-Cu system taken from the AFLOWlib repository. The model is tested on different datasets composed entirely of ternary intermetallics. It is shown that an accuracy below 30 meV/atom can be achieved for alloys in their equilibrium structures, sufficient for an effective screening model. The MLIAPs are however unable to perform relaxation due to their poor energy predictions on out-of-equilibrium structures. Since suitable error metrics, capable of pinpointing unrelaxed structures, are verified, the devised model can be used in a high-throughput screening setting, in which candidates are physically sound compounds.

In the follow-up study, this surrogate to DFT is incorporated into a workflow aimed at constructing reliable, DFT-level ternary convex hulls. This is achieved by two means. Firstly, the prototypes used as candidate ternary compounds are built from the structures of the low-enthalpy alloys of the binary subsystems. These form reasonable guesses for equilibrium structures, owing to the close similarity between binary and ternary alloys for transition metals. Secondly, measures are taken in order to increase the robustness of the screening process. These notably involve undertaking partial ionic

relaxation, driven by SNAP, as well as an assessment of the reliability of the predictions made, through the use of an error metric. The final workflow developed is tested on Ag-Au-Cu and Mo-Ta-W, and is capable of identifying novel ternary compounds, absent from AFLOWlib, and thus produces DFT-accurate ternary convex hulls. This is achieved by probing a large number of candidates and focusing all the heavy *ab initio* calculations on the most promising candidates. In a final section, the recently introduced M3GNet universal force-field is inserted into the workflow. This enables higher accuracy and throughput, as exemplified by the better convex hulls obtained and the larger number of compounds tested. It is shown how this M3GNet workflow can be used to identify promising regions of ternary convex hulls, even for magnetic systems.

Keywords: *Machine Learning, DFT, High-Throughput, Machine Learning Interatomic Potentials, Convex Hull, Ternary Alloys*

Acknowledgements

Despite the highly solitary nature of the PhD, the four years were full of collaboration and connection. The journey would not have been worth it without all the people who were there along the way.

I would first like to thank my loving family, who never fail to be fun and upbeat, and have always been my home. For many, six months stuck in one house with their family during the pandemic may have been too much, but for the Rossignols, everything was smooth sailing. I truly cherish the moments we spent together and feel very lucky to know that every travel back home will be the same. A very special thanks goes to my mother, who works tirelessly to create such warmth (and blaze) in our family and who has been unconditionally supportive, especially in the difficult times.

While exciting, venturing into a new country and culture was also a daunting prospect, but Dublin became a new home, thanks to a wonderful group of friends. Laura, Luke, Matteo, Mike, Michelangelo and I all started the PhD at the same time, and I am very grateful to have formed tight bonds with such brilliant, kind and supportive folks so quickly. Thanks to Luke for being a fantastic flatmate, for always being great craic at all of our nights out, for his efforts in integrating me into Irish culture and for being the very best of friends. Thanks to Mike, for putting up with me in all the work we have done together, and for believing in the project. More importantly though, for all the good times spent together, for always being there and for being one of the most caring men I know. Thanks to Laura for putting a smile on everyone's face, organising so many fun occasions and always being so kind. Thanks to Matteo for the quirky restaurant suggestions and for being such a calm and nice guy to be around. And thanks to Michelangelo for the constant enthusiasm, lighting up our pub sessions, and, with Stefania, being such a lovely host.

I would also like to express my gratitude towards the people who supported my

education and research, ultimately leading to the writing of this thesis. First and foremost, to Stefano Sanvito for designing the project, for providing valuable advice to guide the work and for listening to my difficulties. Thank you to Stefania for making sure every single administrative manner was looked after and for keeping an eye out for everyone in the group.

Alessandro Lunghi, Simone Köcher and Urvesh Patil and Rajarshi Tiwari were post-docs in the group who provided helpful support for the project. Additionally, Michail Minotakis, Matteo Cobelli, Luke Gilligan, Bruno Focassio and Michelangelo Domina all contributed to fruitful discussions during regular meetings and informal chats. Thank you to my good friend Gabriel Krenzer for the collaboration on the project on superionic conductivity. The Irish Research Council funded my PhD as part of the Advanced Laureate Award (IRCLA/2019/127) and Trinity College Dublin and the School of Physics provided support over the four years. Calculations were performed on clusters provided by the Trinity Centre for High Performance Computing and the Irish Centre for High End Computing.

I would also like to acknowledge all the members, past and present, of Prof. Sanvito's group. From the members who were here upon joining, Rajarshi, Emmanuele, Mario, Anais, James, Stella, Alessandro, Andrea, Maria, Urvesh, Simone, Peng, Nina and Rui, to those who joined along the way, Paddy, Willy, Suman, Valerio, Annie, Sourav, Akash, Eoin, Reena, Anita, Ümit, Rutchapon, Abhishek, Eoghan, Antik, Declan, and Michael. Every member has contributed to making the group vibrant and diverse, as well as a supportive and welcoming environment. I would also like to thank Bruno, who was a visiting student for a year, who fit in immediately and has been a great friend. A special thanks to the group members who were in Lloyd 2.10, as well as other friends from the Lloyd building, and a special thanks to Meriç and Eoin.

Outside the PhD life, I would like to thank the Trinity Triathlon and Cycling clubs, especially Marc, Matthieu, Francis, Grace, Dani, Sinéad, Amy and Maxence for being a great bunch to be with. Thanks to my friends from Bristol, George, Félix, Gabriel, Reegan and Said for the fun trips and calls. Thank you to my good running buddy Jorge for the coffee breaks. Thank you to Paul for always listening. Finally, thank you to my girlfriend Amélie for believing in me, supporting me and being my ray of sunshine.

Contents

1

Introduction

1.1 Materials in the Modern Era

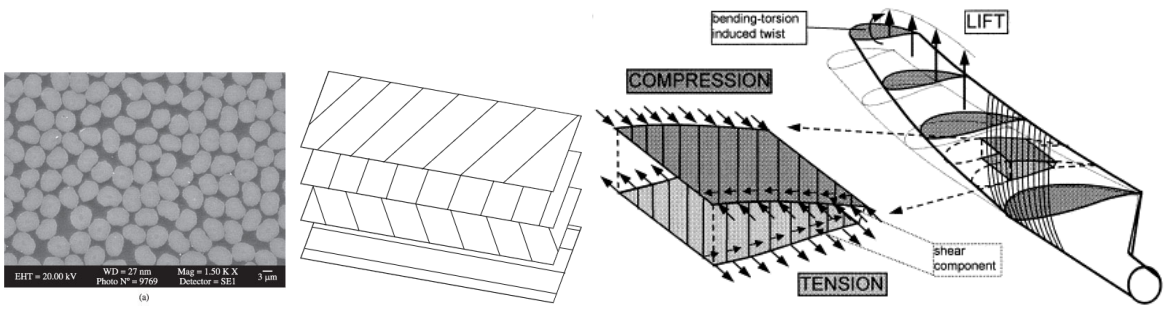
In the digital age, it may seem like our lives have become less rooted in material needs, as it is no longer necessary to buy a newspaper to hear about the latest events or write a letter to a friend far away. Instead, a simple search on a web browser and a few taps of a keyboard suffice, all through a screen that interfaces to an endless virtual world. Yet behind such displays, each laptop and smartphone host a wealth of tailor-made materials, intricately designed and intertwined to form devices with features down to the nanoscale. Silicon lies at the core of modern electronic devices, highlighted by the fact that in the manufacturing of computer chips, it is possible to grow single crystals the size of 2 m of the semiconductor at more than 99.9999% purity. The highest-performing chips require etching with machines worth in excess of $150 million [1].

Aside from silicon, other materials form the bedrock of our world. On the more primitive side, steel and concrete are fundamental to all modern infrastructure, from buildings and bridges to machines and ships. Synthetic plastics have made their way into all industries, thanks to their fantastic tunability, and it is impossible not to encounter them in our everyday lives. All these materials are essential because they combine excellent properties (e.g., tunable electronic conductivity in the case of silicon) with highly scalable and optimised manufacturing processes, such that they can support pivotal industries at a global scale. On top of such fundamental materials, others can be specifically chosen and designed to bring additional attributes. Modern characterisation equipment has notably helped assist in the search and optimisation of

materials design. As an illustration of such a process, the case of the development of carbon fibre in the past 60 years is briefly described.

Carbon fibre is a composite material [2, 3], made of carbon filaments in a matrix of another material, which is most commonly a plastic. The carbon strands form turbostratic carbon cylinders [4], with a diameter of $\sim 1\mu$m and can be several meters in length. An SEM image of the composite is given in Figure 1.1a. They possess a high Young's modulus and tensile strength, as well as being low density, they are generally unreactive and can have excellent electrical and heat transport properties. Such properties are highly desirable for light-weight structures. However, their high anisotropy and small width makes such filaments deformable in the transverse direction and thus cannot withstand compression. The plastic resin that surrounds them in the composites enables the assembly of many carbon strands to form a solid sheet. The former supports and protects the fibres and spreads any load applied across them. In a unidirectional sheet, all fibres are aligned and point in one direction, in which the tensile strength and Young' modulus is very high. In the transverse direction, however, these are far lesser and the material may crack along interfaces between the resin and the strands. To prevent this, the strands may be woven, or several unidirectional sheets may be layered, with each sheet possessing a different direction of alignment, as illustrated in Figure 1.1b. Due to the two-dimensional (2D) characteristic and small thickness of the sheets, the strength in the transverse direction is low, implying low resistance to loads but also higher flexibility in this direction. Carbon fibre can be used to replace materials such as steel and aluminium in light-weight structures such as planes, automobiles and bikes, provided the intended load is applied in the right direction.

One remarkable feature of such a material is how it can be designed, at different areas and in fact scales, to obtain certain intended properties. Indeed, the precursor used to create the carbon filament impacts the microstructure of the carbon, the type of resin used and the chemical interface between resin and fibres can be tuned, and crucially, the lamellar ordering and in-sheet weaving of the material can be changed to obtain desired properties [3]. As an example, sheets can be layered with different filament orientations to cause an expansion-twisting coupling [7, 8], which, as the name suggests, corresponds to a twisting of the sheet upon its expansion due to a load. The



(a) (b) (c)

Figure 1.1: Images and illustrations of carbon fibre and its applications. Panel (a): SEM image of carbon fibre composite showing the carbon filaments in an epoxy matrix (image taken from Reference [5]). Panel (b): Illustration of the lamellar structure of layered carbon fibre sheets with different strand orientations. Panel (c): Illustration of the expansion-bending coupling of carbon fibre on a wind turbine blade (image taken from Reference [6]).

direction and angle of such a distortion can be carefully controlled by the layering of the material. This can be used for useful applications in devices, such as wind turbines [6, 9, 10, 11], where blades are carefully coated with carbon fibre to obtain twisting effects upon bending. This is of significant interest, since in operation, high wind speeds can cause severe shear on the blades and cause damage to the turbine [12]. By tilting with larger blade deflections, the angle of incidence of the flowing wind is changed, such as to decrease shearing imposed on the blade. The expansion-induced bending on a blade is illustrated in Figure 1.1c. What is intended to be highlighted here is that from a tailoring of the structure of the material, the behaviour is changed and adapted to the precise applications it is used in. It is this concept that is at the heart of the inverse materials design framework [13].

1.2 Inverse Materials Design

In this area, the length scale that serves as a playground for designing novel materials is that of Å and nm. Rather than assembling a composite with different phases, atoms are the building blocks, arranged in a microscopic structure that is most often crystalline in nature. The nature of the atoms, the stoichiometric ratios, the atomic neighbourhoods and the lattice parameters all serve as parameters that can be tuned in order to devise novel compounds. The macroscopic properties are thus designed at the most intrinsic level, giving full control over the most rudimentary properties, such

as densities and elastic properties, to more sophisticated ones such as band gaps and thermal conductivities. This is possible because of the fundamental character of the design procedure, coupled with the immensity of the space of exploration, composed of all possible combinations of atomic types in different geometric arrangements. The grail is to be able to effectively map such a space to that space of material properties. This is illustrated in Figure 1.2. This would provide the means to find compounds with pre-determined attributes, thus designing materials. The flip side of having such an expansive search space is the difficulty in probing it efficiently, such as to effectively discover new materials with desired properties.

The design workflow required to produce this mapping would take the form of a variety of computational studies that can be performed at large scales, systematically and in theoretically-controlled settings. This of course relies on the fact that it is possible to accurately predict material properties using such methods. Considering the control is at the atomic level, the codes used are referred to as first principles, since they rely on solving quantum mechanical equations that govern the behaviour of matter at this scale. More specifically, it is the material's electronic structure that must be accurately modelled. For the majority of cases, codes based on density functional theory (DFT) are employed, due to their *ab-initio* nature and hence high accuracy [14, 15], as well as their $O(N^3)$ scaling with system size, which is favourable with respect to other first principle methods. While the calculations act as the guide to finding novel materials, the overall design must rely on a collaboration between computational and experimental scientists. The former identify potential compounds with properties of interest while the latter attempt to synthesize and characterise them, while giving feedback to help refine the computational exploration.

Returning to the example of carbon fibres, within the inverse materials design paradigm, the aim would be to identify the carbon filaments, or even turbostratic carbon, as being a material with excellent tensile strength and a high Young's modulus. In reality, the carbon filaments used in the composites were discovered (for the second time[1]) by accident. Roger Bacon was studying graphite at high temperatures and pressures, when he observed solid filaments deposited on electrodes from the vapour

[1]Joseph Swan and Thomas Edison had in fact discovered them in the 19^{th} century to use them as filaments in light bulbs [16]

phase [17]. This has very much been the standard for the discovery of useful materials throughout human history and is an unreliable methodology when searching for compounds with targeted properties. Instead, the inverse materials design approach strives for a direct exploration of chemical and structural space, with the aim of discovering new materials.

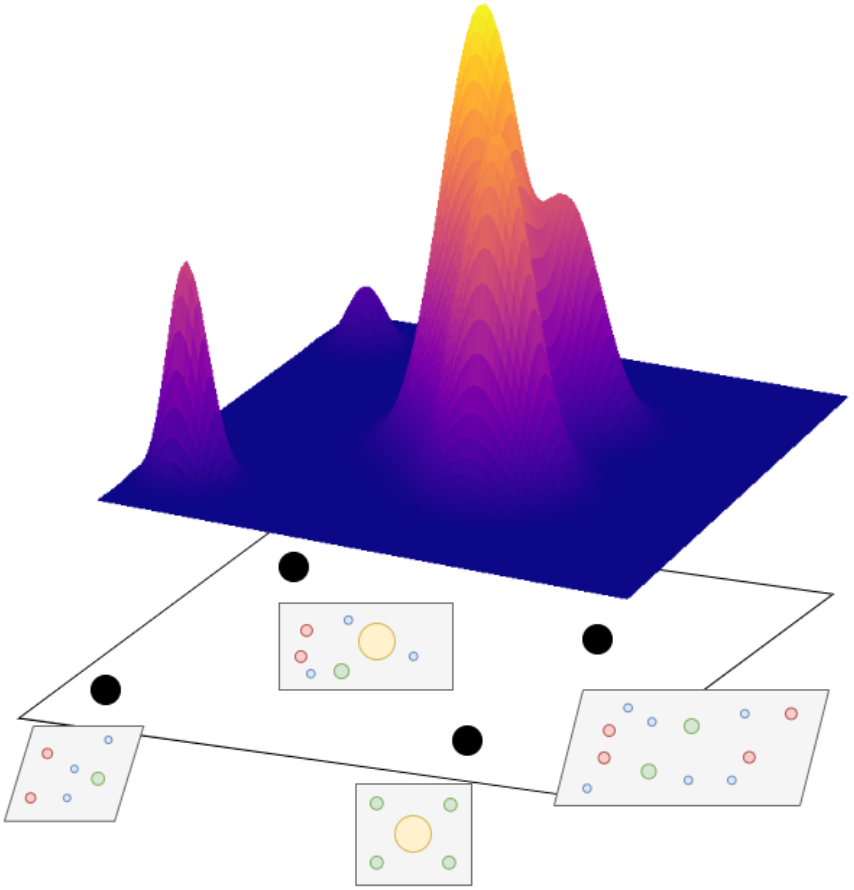


Figure 1.2: Illustration of the mapping between material and property space. The bottom plane represents materials space, in which atomic species and positions as well as unit cell dimensions can vary. It is mapped onto a property space, illustrated above by the variations of the property within that space. In the inverse materials design scheme, the property function is probed to obtain desirable values of that target property and the associate material is revealed.

The key idea behind inverse design is that of finding compounds with pre-determined, desired properties. It is therefore not a "blind" set of calculations, as is the case for direct design, for which brute force calculations of different properties are performed over a large range of constituent atoms, compositions and structures. Zunger, the father of the field, defines three modalities of the field [13]. In the first, the constituent elements are fixed and the composition and configurations of these atoms can be changed. A

target property is optimised by means of an algorithm, such as a genetic algorithm, guiding the search. The second modality involves making use of databases of structures and their properties and to develop a means of mapping the property to the compounds by means of proxy calculations or data mining techniques. The tool created can thus be used to discover *existing* novel materials but that unsuspectedly possess desired properties. Note, this method is for a large range of constituent elements and structures. Finally, the third modality focuses on finding novel stable compounds for which neither composition nor structure are known. By extension, their properties are also unknown. It is this last modality that is the focus of this work. The approach typically employed for such a task is to perform many DFT calculations in a high-throughput approach, for many different compounds with varying structures, elements, and compositions. The total energies of these structures, as well as those of competing phases, are calculated and the most stable ones (that lie on the convex hull [13]) are identified as stable. This approach has already led to the discovery of novel materials [18]. Note that the choice of systems studied in this approach could be guided by prior screening, and the materials discovered could undergo further studies.

The design process therefore relies on making use of carefully chosen DFT calculations. These can either be computed in the course of a specific study or previously computed results can be recycled. This former approach is made possible thanks to the existence of online repositories that group many DFT-predicted properties of a wide variety of materials [19, 20, 21]. Such databases also exist for experimental data [22, 23]. The exploration of these databases provides a direct means to perform inverse design workflows for the second modality aforementioned. For example, M. Ashton *et al.* [24] scanned the Materials Project [20] to search for bulk materials that may produce 2D materials via exfoliation. A set of design principles based on structural features (low packing numbers, large interlayer gaps, classifying atomic clusters), along with distance from the convex hull, were used to identify such materials within the repository. This is a common approach to help accelerate the mapping between structure and property. Machine learning provides an alternative approach in performing such a task. It can be used to identify trends in databases [25], automatically infer design features [26] and make direct material predictions, in place or in conjunction with DFT [27] and can thus be used to fast-track the design process. The work presented in this thesis is

situated at the confluence of many of the concepts evoked. It presents a methodology to discover new materials using state-of-the-art high-throughput methods. The novelty in the approach comes from the use of machine learning to accelerate the process, as well as the exploitation of large online repositories. This work inscribes itself in the third modality of the inverse design framework, but uses tools that are typically used in the other modalities.

1.3 Thesis Outline

This thesis shows how DFT-based high-throughput calculations, used to construct ternary convex hull diagrams and hence identify novel compounds, can be accelerated with the use of machine learning. Specifically, machine learning interatomic potentials (MLIAPs) [28] are employed for direct energy predictions and act as inexpensive surrogates to DFT. These potentials are specially trained, by making use of pre-existing data on AFLOWlib [19] that are directly relevant to the construction of the convex hull. This implies that no costly DFT calculations are wasted in the construction of the potential. Due to the higher wealth of data available on binary compounds relative to ternary ones, the former make up the training data for the model used to predict energies of the latter. This extrapolation from binaries to ternaries is not only exploited for the potential, but also for the creation of prototypes used to construct ternary convex hulls, as these are inspired from the structures of binaries.

The thesis is separated into four parts, including two methods chapters, as detailed below. Some of the work completed in the course of the PhD is not presented here. It is in relation to work done in collaboration with G. Krenzer, with the final results published in Reference [29].

Chapter 2 : In the second chapter, the central concepts in DFT are presented. This is the method of choice to model the electronic structure and properties of materials. Details on the core parts of the theory are described, leading to the formulation of Kohn-Sham (KS) DFT. The connection to the central properties that are calculated, and the practical methods employed to solve the KS equations are given. This is presented in line with the implementation in VASP [30], used to perform all *ab-initio* calculations. Finally, emphasis is put on some of the relevant numerical parameters

that need to be converged and optimised.

Chapter 3: In the third chapter, machine learning and MLIAPs are presented. First, the main concepts in machine learning are elucidated, before shifting focus onto practical algorithms that are used to make actual predictions. Building on top of these ideas, MLIAPs are defined. This is broken down into the presentation of their general features. Then, specific atomic fingerprints, used as features to represent the material landscape, are introduced. These are the atom-centered symmetry functions and the bispectrum components that are respectively used with neural networks, to form neural network potentials, and with ridge regression, in the spectral neighbour analysis potential. The latter is the central MLIAP of this work. Before moving to the results, other MLIAPs and their most successful applications are mentioned.

Chapter 4: This chapter focuses on the first set of results. The introduction presents references to related pieces of work, setting the position of the study within the scientific literature. Important points of theory, related to the definition of thermodynamic stability, as well as relevant methods are also given before showcasing the results. These focus on the work published in Reference [31]. An ensemble of SNAP models are trained on binary data from AFLOWlib to predict the energies of associated ternary compounds. Ag-Au-Cu is selected as the test system. Full details are given on the training process and on how the model can be used are given, alongside its limitations. The result is an energy model surrogate to DFT, which can be used to predict the energies of equilibrium ternary compounds and identify out of equilibrium structures through adequate error metrics.

Chapter 5: In the second results chapter, the surrogate model is deployed in a practical workflow to construct ternary convex hulls. Part of these results are in the preprint in Reference [32]. The relevant methods are given first. The workflow makes use of binary compounds to train the SNAP ensemble and to create ternary compounds that form a pool of candidates that may be stable phases. The screening of these candidates is done with the MLIAPs, which are used to partially relax and detect the lowest energy compounds. Note that this process has in-built steps that ensure the robustness of the predictions. During the development of this workflow, M3GNet, a universal force-field, was released. Separate results that integrate the potential in the workflow are also presented and compared to those obtained with SNAP. Note that

several ternary systems are studied, such as the Ag-Au-Cu, Mo-Ta-W, Al-Fe-Ni and Bi-Fe-X (X=Ta, Zr) systems.

Chapter 6: In the final chapter, the main outcomes of the thesis are highlighted. A summary of the main results is also provided. An outline of the future work that could stem from the project is also presented.

2

Electronic Structure and Density Functional Theory

2.1 The Many-Body Wavefunction

When describing the physics of a material, such as a block of iron, one must model the behaviour of the constituent nuclei and electrons. These interact via the Coulomb interaction and the light electrons bind the heavy nuclei in a solid-state structure. The former are essential in determining the physical and chemical properties of a material and the description of their collective behaviour is termed the electronic structure of a material. At the atomic scale, quantum mechanics provides the best framework to model the electronic behaviour. This section follows Reference [33].

Given a system of N atoms and N_e electrons, the system is described by the many-body wavefunction Ψ. It is from this quantity that the material properties are expressed in terms of expectation values. Given that the nuclear positions are labelled $\vec{R}^N$ and the electron positions $\vec{r}^{N_e}$, the wavefunction crucially depends on the positions of all nuclei and electrons. This dependence of Ψ on all coordinates can be shown explicitly as

$$\Psi\left(\vec{R}^N, \vec{r}^{N_e}\right) = \Psi\left(\vec{R}_1, \vec{R}_2, ..., \vec{R}_N, \vec{r}_1, \vec{r}_2, ..., \vec{r}_{N_e}\right), \tag{2.1}$$

where $\vec{R}_1, \vec{R}_2, ..$ ($\vec{r}_1, \vec{r}_2, ..$) are the positions of the individual nuclei (electrons). This dependence on all coordinates simultaneously results from the fact that all nuclei and

electrons interact with each other and thus their positions are correlated. Assuming no relativistic effects are at play and that one is not interested in time evolution, finding this wavefunction for a given system requires solving the time-independent Schrödinger equation

$$\widehat{H}\Psi = E\Psi, \tag{2.2}$$

where E is the total energy of the system and $\widehat{H}$ is the Hamiltonian. Going into more detail, the latter is written explicitly as

$$\begin{aligned}\widehat{H} = &-\frac{\hbar^2}{2m_e}\sum_i \nabla_i^2 - \sum_{i,I}\frac{Z_I e^2}{\left|\vec{r}_i - \vec{R}_I\right|} + \frac{1}{2}\sum_{i\neq j}\frac{e^2}{|\vec{r}_i - \vec{r}_j|} \\ &-\sum_I \frac{\hbar^2}{2M_I}\nabla_I^2 + \frac{1}{2}\sum_{I\neq J}\frac{Z_I Z_J e^2}{\left|\vec{R}_I - \vec{R}_J\right|},\end{aligned} \tag{2.3}$$

where I, J (i, j) label individual nuclei (electrons) and, Z_I (M_I) are the individual nuclear charges (masses). The sums over I (i) run over all nuclei (electrons) in the system. The first term expresses the electron kinetic energy, the second the nucleus-electron interactions, the third the electron-electron interactions, the fourth the nuclear kinetic energy and the fifth the nucleus-nucleus interactions. Since the masses of the nuclei are far larger than those of electrons, the latter move on a much shorter timescale than the former. Within the Born-Oppenheimer (BO) approximation [34], the many-body wavefunction is separated into the electron and the nucleus wavefunctions. For electronic structure theory, the focus is on former. The kinetic energy term of the nuclei may be neglected and the nucleus-nucleus interaction is a fixed energy E_{II}. The problem is therefore recast, such that the nuclei positions are fixed and treated as parameters of the electronic Schrödinger equation, written as

$$\left(\widehat{T} + \widehat{V}_{\text{ext}} + \widehat{V}_{\text{e-e}} + E_{II}\right)\psi = E\psi, \tag{2.4}$$

where ψ is the electronic wavefunction and $\widehat{T}, \widehat{V}_{\text{ext}}$ and $\widehat{V}_{\text{e-e}}$ are respectively the electron kinetic energy, electron-nucleus and electron-electron interaction terms from Equation (2.3). Note that the electron-nucleus term can now be viewed as the energy contribution

from the interaction of the electrons with an external potential, $V_{\text{ext}}(\vec{r})$, set up by the fixed nuclei. This can be expressed as

$$\widehat{V}_{\text{ext}} = \sum_i V_{\text{ext}}(\vec{r_i}), \tag{2.5a}$$

$$V_{\text{ext}}(\vec{r}) = \sum_I V_I\left(\left|\vec{r} - \vec{R}_I\right|\right), \tag{2.5b}$$

where V_I is the Coulomb potential set up by nucleus I.

The external potential term, as well as E_{II}, are parameterised by $\vec{R}^N$. In the BO approximation, E_{II} is simply a classical additive term to the total energy. The electronic structure problem now reduces to finding ψ by obtaining the eigenstates the Hamiltonian of Equation (2.4), but ψ remains a many-body wavefunction dependent upon $\vec{r}^{N_e}$. As a result, the Hilbert space required to describe such a many-body wavefunction scales exponentially with the total number of particles, making even the storage of such an object intractable [33]. This is alleviated by expressing the problem in terms of the electronic density, as will be discussed in the following section.

2.2 Density Functional Theory

The electron density $n(\vec{r})$ gives the number of electrons per unit volume in 3D space, with coordinates $\vec{r}$. It is obtained from the expectation value of the density operator $\widehat{n}(\vec{r})$. Together, these are given by

$$\widehat{n}(\vec{r}) = \sum_{i=1}^{N_e} \delta(\vec{r} - \vec{r_i}), \tag{2.6a}$$

$$n(\vec{r}) = N_e \frac{\int d\vec{r}_2 d\vec{r}_3 ... d\vec{r}_{N_e} \left|\psi(\vec{r}, \vec{r}_2, .., \vec{r}_{N_e})\right|^2}{\int d\vec{r}_1 d\vec{r}_2 ... d\vec{r}_{N_e} \left|\psi(\vec{r}_1, \vec{r}_2, .., \vec{r}_{N_e})\right|^2}. \tag{2.6b}$$

Following this definition, it is also of interest to express the energy contribution to E resulting from the electron interaction with the external potential, the expectation value E_{ext}, in terms of the density,

$$E_{\text{ext}} = \int d\vec{r}\, V_{\text{ext}}(\vec{r})\, n(\vec{r}). \tag{2.7}$$

Density Functional Theory (DFT) relies on a central proof that all properties of a fully interacting many-body system can be expressed in terms of the ground-state density $n_0(\vec{r})$ of the system. More precisely, properties can be written as functionals of the latter. The ground-state density is the one for which E in Equation (2.4) is minimised. In the following subsections, this is proved from the two Hohenberg and Kohn (HK) proofs [35]. The two subsequent subsections are dedicated to showing how the functional expressions can be used practically.

2.2.1 HK Theorems

In their seminal paper in 1964, Hohenberg and Kohn lay the foundations of DFT through two proofs named the Hohenberg-Kohn theorems. The first theorem states that for a system of interacting particles in an external potential $V_{\text{ext}}(\vec{r})$, the external potential is determined uniquely, up to a constant, by $n_0(\vec{r})$.

HK Theorem 1

The theorem is proved *ab absurdum*. Consider two different external potentials $V_{\text{ext}}^{(1)}(\vec{r})$ and $V_{\text{ext}}^{(2)}(\vec{r})$ that differ by more than a constant and define two different electronic Hamiltonians $\widehat{H}^{(1)}$ and $\widehat{H}^{(2)}$ [the Hamiltonians from Equation (2.4)]. It is assumed for the proof that these lead to the same ground-state density, n_0. The two Hamiltonians have two different ground-state wavefunctions $\psi^{(1)}$ and $\psi^{(2)}$, respectively, (as well as energy eigenvalues $E^{(1)}$ and $E^{(2)}$) that both lead to the same ground-state density. $\psi^{(2)}$ is not the ground-state of $\widehat{H}^{(1)}$ (assuming the ground-state is non-degenerate). Thus, by definition,

$$E^{(1)} = \left\langle \psi^{(1)} \middle| \widehat{H}^{(1)} \middle| \psi^{(1)} \right\rangle < \left\langle \psi^{(2)} \middle| \widehat{H}^{(1)} \middle| \psi^{(2)} \right\rangle . \tag{2.8}$$

The right-hand term can be re-written as

$$\begin{aligned} \left\langle \psi^{(2)} \middle| \widehat{H}^{(1)} \middle| \psi^{(2)} \right\rangle &= \left\langle \psi^{(2)} \middle| \widehat{H}^{(2)} \middle| \psi^{(2)} \right\rangle + \left\langle \psi^{(2)} \middle| \widehat{H}^{(1)} - \widehat{H}^{(2)} \middle| \psi^{(2)} \right\rangle \\ &= E^{(2)} + \int d\vec{r} \left[V_{\text{ext}}^{(1)}(\vec{r}) - V_{\text{ext}}^{(2)}(\vec{r}) \right] n_0(\vec{r}) . \end{aligned} \tag{2.9}$$

This leads to,

$$E^{(1)} < E^{(2)} + \int d\vec{r} \left[V_{\text{ext}}^{(1)}(\vec{r}) - V_{\text{ext}}^{(2)}(\vec{r}) \right] n_0(\vec{r}) \,. \tag{2.10}$$

The same reasoning can be followed to obtain the same equation as (2.10) but with the indices (1) and (2) swapped. Adding such an inequality with that above gives,

$$E^{(1)} + E^{(2)} < E^{(1)} + E^{(2)}, \tag{2.11}$$

which is a conflicting inequality. From the *ad absurdum* reasoning, this contradictory statement proves that the initial postulate is false. Therefore, no two external potentials that differ by more than a constant can give the two same (non-degenerate) ground-state charge densities. In other words, the ground-state density uniquely defines the external potential, up to a constant. Since in turn the electronic Hamiltonian is uniquely defined by the external potential, all wavefunctions that are solutions of the Schrödinger equation, ground and excited states, are uniquely determined by the ground-state density. While this is a non-trivial statement, it does not provide any practical way to avoid solving the many-body problem. The second theorem provides a step forward in this direction.

HK Theorem 2

In the second proof, a functional relationship between the energy and the density is established and the procedure for obtaining the ground-state energy and density is given. Note that this proof is valid for "V-representable" densities, which correspond to all densities that are ground-states of Equation (2.4) with some external potential. From HK theorem 1, since $n(\vec{r})$ is a function and the total energy can be expressed as a functional E_{HK} of the density, the individual energy terms in Equation (2.4) can also be expressed as a functional of the density, as,

$$E_{\text{HK}}[n] = T[n] + E_{e-e}[n] + \int d\vec{r}\, V_{\text{ext}}(\vec{r})\, n(\vec{r}) + E_{II}, \tag{2.12}$$

where T and E_{e-e} are the kinetic energy and electron-electron interaction energy functionals, respectively. Together, they correspond to the internal energies and encompass all energy terms related to the electrons and their interactions amongst themselves. As such, these functionals are independent of the external potential and are universal for

all electron systems. It follows from the variational principle, that for a given system the density that minimises E in Equation (2.12) is the ground-state density $n_0(\vec{r})$. This statement corresponds to the second HK theorem. It implies that the minimisation of the energy functional with respect to the density yields the ground-state energy and electron density. The functional is therefore enough to determine the ground-state of the system.

Note that here the functional is not known, and the definition is unclear, as is the space of densities that are generated by some external potential. Levy and Lieb [36, 37, 38] extended the range of applicable densities for which the theorem holds, giving a more precise definition of the energy functional and on how to obtain it. This is enabled by using "N-representable" densities, which correspond to the space of densities that can be obtained from any wavefunction of the space of wavefunctions of N_e electrons. Note that the minimum of the Levy-Lieb functional coincides with that of the HK functional in the case of a system of N_e electrons in an external potential.

While it is now established that minimising $E_{\mathrm{HK}}[n]$ with respect to the density in Equation (2.12) provides a methodology for obtaining the ground-state energy of a system, the exact form of the functional, notably for the kinetic energy and the electron-electron interaction contributions, is unknown. Kohn and Sham published a practical solution to this problem in the following year, by proposing a scheme to find the ground-state density. Furthermore, they suggested some approximations to obtain an analytical expression of the energy functionals.

2.2.2 Kohn-Sham Scheme

The Kohn-Sham approach [39] displaces the problem of solving an interacting many-body system with N_e particles to solving that of a system of N_e independent (non-interacting) particles. In order to do so, a specific *ansatz* is chosen that relies on two assumptions. The first is that the exact ground-state density of the interacting many-body system can be represented by the density of an auxiliary problem of non-interacting particles. This leads to having an auxiliary single-particle Hamiltonian for such a system. The second assumption is that this auxiliary Hamiltonian has the usual kinetic energy operator and only one other term, a local effective potential $V_{\mathrm{eff}}(\vec{r})$. In general, a spin index should be included, but it is excluded here for clarity. The

auxiliary Hamiltonian is,

$$\widehat{H}_{\text{aux}} = -\frac{\hbar^2}{2m_e}\nabla^2 + V_{\text{eff}}\left(\vec{r}\right). \tag{2.13}$$

For an N_e independent electron system, the Hamiltonian $\widehat{H}_{\text{aux}}$ has eigenvalues ϵ_i and eigenstates φ_i. The N_e lowest energy eigenstates are occupied in the ground-state. The density of such a system is then given by,

$$n\left(\vec{r}\right) = \sum_{i=1}^{N_e} \left|\varphi_i\left(\vec{r}\right)\right|^2. \tag{2.14}$$

Two functionals of the density are then defined, the independent-particle kinetic energy T_s

$$T_s = \frac{-\hbar^2}{2m_e}\sum_{i=1}^{N_e} d\vec{r}\left|\nabla\varphi_i\left(\vec{r}\right)\right|^2, \tag{2.15}$$

as well as the classical Coulomb interaction energy of the electron density with itself (named the Hartree energy [33])

$$E_{\text{Hartree}}\left[n\right] = \frac{1}{2}\frac{e^2}{4\pi\epsilon_0}\int d\vec{r}\int d\vec{r}\,'\frac{n\left(\vec{r}\right)n\left(\vec{r}\,'\right)}{\left|\vec{r}-\vec{r}\,'\right|}. \tag{2.16}$$

The latter corresponds to the classical contribution to the interaction of the electron density with itself. It is a part of the electron-electron interaction. In the Kohn-Sham approach, the energy functional of the full interacting many-body system is re-written as,

$$E_{\text{KS}}\left[n\right] = T_s\left[n\right] + \int d\vec{r}\, V_{\text{ext}}\left(\vec{r}\right)n\left(\vec{r}\right) + E_{\text{Hartree}}[n] + E_{II} + E_{\text{xc}}\left[n\right]. \tag{2.17}$$

While T_s is defined explicitly in terms of orbitals in Equation (2.15), it is a functional of the density, through the application of HK theorem 1 to the Hamiltonian of the auxiliary problem. In contrast, $E_{\text{xc}}\left[n\right]$ is the exchange-correlation functional, defined as the difference between the kinetic and electron-electron interaction functionals and the independent-particle functionals, T_s and E_{Hartree}. Thus, $E_{\text{xc}}\left[n\right]$ encompasses non-classical many-body contributions to the energy. The analytical form of the functional

is, however, unknown.

2.3 Kohn-Sham variational equations

In the Kohn-Sham approach, the ground-state is found by minimising the energy functional E_{KS} with respect to the density. Since T_s is expressed in terms of the orbitals corresponding to the eigenstates of the auxiliary problem, and these define the energy in the Kohn-Sham *ansatz*, E_{KS} is minimised with respect to these. This is performed by solving,

$$\frac{\partial E_{\mathrm{KS}}}{\partial \varphi_i} = \frac{\partial T_s}{\partial \varphi_i} + \left[\frac{\partial E_{\mathrm{ext}}}{\partial n} + \frac{\partial E_{\mathrm{Hartree}}}{\partial n} + \frac{\partial E_{\mathrm{xc}}}{\partial n}\right] \frac{\partial n}{\partial \varphi_i} = 0. \tag{2.18}$$

The expression above is obtained through the application of the chain rule. E_{ext} [defined in Equation (2.7) and the second term in Equation (2.17)] is the functional of the density that gives the Coulomb interaction energy between the electron density and the external potential V_{ext}. The first and last terms give, respectively,

$$\frac{\partial T_s}{\partial \varphi_i} = -\frac{\hbar^2}{2m_e}\nabla^2 \varphi_i, \tag{2.19a}$$

$$\frac{\partial n}{\partial \varphi_i} = \varphi_i^*. \tag{2.19b}$$

Solving Equation (2.18) with the Lagrange multiplier, imposing the conservation of the total number of particles, leads to the Kohn-Sham equations

$$\widehat{H}_{\mathrm{KS}}\ \varphi_i = \epsilon_i \varphi_i. \tag{2.20}$$

The effective Kohn-Sham Hamiltonian is,

$$\widehat{H}_{\mathrm{KS}} = -\frac{\hbar^2}{2m_e}\nabla^2 + V_{\mathrm{eff}}\left(\vec{r}\right), \tag{2.21a}$$

$$V_{\mathrm{eff}}\left(\vec{r}\right) = V_{\mathrm{ext}}\left(\vec{r}\right) + \frac{\partial E_{\mathrm{Hartree}}}{\partial n} + \frac{\partial E_{\mathrm{xc}}}{\partial n}. \tag{2.21b}$$

The solutions to the Kohn-Sham equations give the ground-state energy and density of the interacting many-body system with Equations (2.14) and (2.17). The eigenstates φ_i are referred to as KS orbitals and do not represent real, individual electron states.

Clearly, the Kohn-Sham effective potential V_{eff} is dependent upon the density, thus the Kohn-Sham equations are solved self-consistently with Equation (2.14). Note that at the true ground-state density, the Kohn-Sham potential becomes the effective potential in Equation (2.13). While the re-formulation of the many-body problem so far remains exact, since the form of the exchange-correlation functional is unknown, approximations must be introduced to solve the equations. The self-consistent procedure typically followed to find the ground-state is illustrated in Figure 2.1. This is the central scheme followed during DFT calculations and is called a self-consistent field (scf) scheme. Aside from the ground state density $n_0(\vec{r})$, the outputs of the scheme that are of interest for this thesis are the total ground-state energy of the system E_0, the forces $\left\{\vec{F}_I\right\}$ acting on each atom I and the stress tensor $\mathbf{\Sigma}$. Note that all other ground-state quantities (dipole, magnetic moments etc.) can also be derived. For the following sections, the energy E (instead of E_0) is taken to be the ground-state energy of a given system with fixed nuclei positions. All systems considered now are crystalline and thus posses a unit cell with an underlying lattice and atomic basis. Two commonly used, approximate exchange-correlation functionals are now introduced.

2.4 LDA and GGA Functionals

The two most common functionals used in practice for DFT are the Local Density Approximation (LDA) and the Generalised Gradient Approximation (GGA). The former, as indicated by its name, assumes a local picture of the exchange and correlation effects. The exchange-correlation functional within the LDA is thus written as

$$E_{\text{xc}}[n] = \int d\vec{r}\, n(\vec{r})\, \varepsilon_{\text{xc}}^{\text{hom}}([n], \vec{r}), \tag{2.22}$$

where $\varepsilon_{\text{xc}}^{\text{hom}}$ is an exchange-correlation energy term that is a functional of the density and of the position in space. In the LDA, it is assumed to be that of a homogeneous electron gas [40]. The expression of the exchange energy for the homogeneous electron gas is known analytically, while for the correlation term, an analytical expression taken from a fit of numerical results is found to be very accurate [41]. It is a local description as the exchange-correlation energy term at position $\vec{r}$ only depends on the density at that very position. It does not depend on variations of the density nor on the values

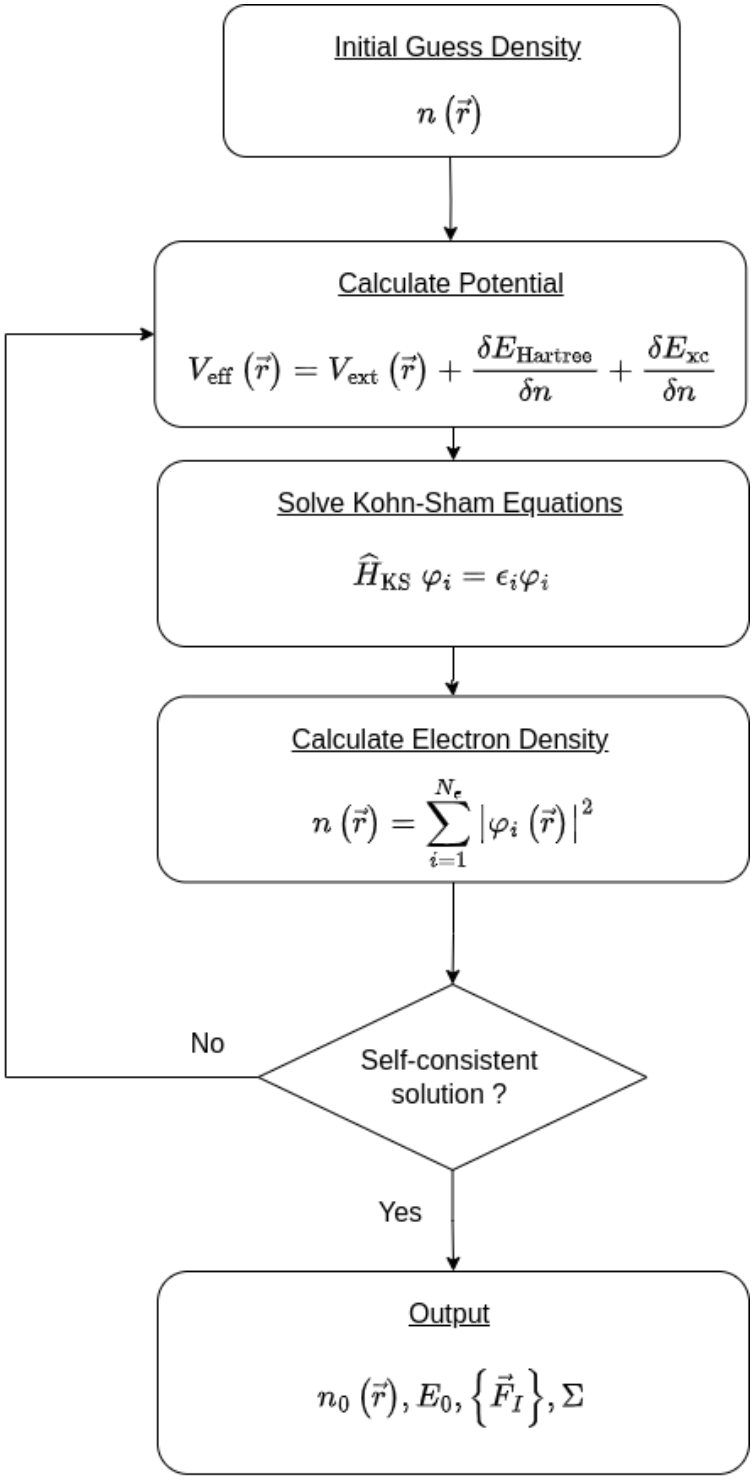


Figure 2.1: Diagram showing the procedure followed by the Kohn-Sham scheme. The output quantities include the ground-state density $n_0(\vec{r})$, the ground-state energy E_0, the forces $\left\{\vec{F}_I\right\}$ and stress tensor $\mathbf{\Sigma}$.

of the density away from $\vec{r}$.

For the GGA functional, the exchange-correlation energy term, $\varepsilon_{\mathrm{xc}}$, is also influenced by the gradient of the density in space. It can be written in a general form as

$$E_{\mathrm{xc}}[n] = \int d\vec{r}\, n(\vec{r})\, \varepsilon_{\mathrm{xc}}\left(\left[n, \vec{\nabla} n, ...\right], \vec{r}\right), \tag{2.23}$$

where the expansion is over different orders of the gradient. As the gradient of the density is also taken into account here, the functional is said to be semi-local, as the exchange-correlation term depends on density changes due to the surroundings of each point. Several methods have been proposed for the expansion of the gradient in Equation (2.23), such as in the Perdew, Burke and Enzerhof (PBE) functional [42].

2.5 Force and Stress Theorems and Bulk Modulus

As indicated in Figure 2.1, the forces acting on each nucleus $\left\{\vec{F}_I\right\}$, as well as the stress tensor $\mathbf{\Sigma}$ of a unit cell, are outputs of the Kohn-Sham scheme. The force theorem [43, 44] provides an expression for the force $\vec{F}_I$ acting on a nucleus I. Considering the nuclei positions $\vec{R}_I$ are parameters of the problem, they are subject to the force,

$$\vec{F}_I = -\frac{\partial E}{\partial \vec{R}_I}. \tag{2.24}$$

This is the energy E in Equation (2.2). Variations of the ground-state wavefunction with respect to atomic coordinates, $\vec{R}_I$, are orthogonal to the ground-state wavefunction. Using this fact, in the derivative in Equation 2.24, the expectation values with dependencies on the variations of the atomic positions vanish. Thus the force is defined only with energy terms that have an explicit dependence on the nuclei positions,

$$\vec{F}_I = -\int \left(d\vec{r}\, n(\vec{r}) \frac{\partial V_{\mathrm{ext}}(\vec{r})}{\partial \vec{R}_I}\right) - \frac{\partial E_{II}}{\partial \vec{R}_I}. \tag{2.25}$$

Here the density is the unperturbed one and only $\vec{R}_I$ is changed. A key finding from this expression is that the force is only influenced by the electron charge density and does not depend on other electron energy contributions (such as the kinetic, exchange, and correlation energies). Note that the expression would have to be re-written, if there

were a dependence of the ground-state wavefunction on the nuclei positions, notably in the case of non-local potentials and if ψ is expressed with certain basis functions.

There is an equivalent expression for the stress tensor $\mathbf{\Sigma}$, based on the stress theorem [45, 46], a generalisation of the virial theorem for pressure. The stress tensor gives the response of a material under strain $\epsilon_{\alpha\beta}$ (α and β are cartesian indices) and it is the derivative of the energy with respect to this strain, per unit volume:

$$\sigma_{\alpha\beta} = -\frac{1}{V}\frac{\partial E}{\partial \epsilon_{\alpha\beta}}. \tag{2.26}$$

In the equation above, $\sigma_{\alpha\beta}$ is a stress tensor element and V is the volume. Since the space is being scaled here, rather than just the ionic positions, the wavefunction and the external potential change with strain as well. The expression for the stress tensor elements, written as an expectation value, is,

$$\sigma_{\alpha\beta} = -\left\langle \psi \left| \sum_k \frac{\hbar^2}{2m_k} \vec{\nabla}_{k\alpha}\vec{\nabla}_{k\beta} - \frac{1}{2}\sum_{k\neq k'} \frac{\left(\vec{x}_{kk'}\right)_\alpha \left(\vec{x}_{kk'}\right)_\beta}{x_{kk'}} \left(\frac{\mathrm{d}}{\mathrm{d}x_{kk'}}\widehat{V}\right) \right| \psi \right\rangle, \tag{2.27}$$

where k and k' are particle indices and $\vec{x}_{kk'}$ is a vector joining particles k and k' and $\widehat{V}$ groups all potential energy terms.

Another quantity that can be derived from DFT is the bulk modulus B_0 of a material at T=0 K. Unlike the previous quantities presented, rather than being the result of a single calculation, this is obtained from several energy calculations performed at different cell volumes. The energy per cell as a function of the volume defines the bulk modulus in the equation of state,

$$B_0 = V_0 \left.\frac{\partial^2 E}{\partial V^2}\right|_{V=V_0}, \tag{2.28}$$

where E, V and V_0, are the cell total energy, volume and minimum volume (at 0 pressure), respectively. The Birch-Murnaghan equation [47, 48] given below provides an analytical expression that can be fit to energy and volume data to extract the bulk modulus, as well the minimum cell energy E_0 and corresponding cell volume V_0. Here B_0' is the pressure derivative of B_0 at equilibrium.

$$E(V) = E_0 + \frac{B_0 V}{B_0'}\left[\frac{\left(\frac{V_0}{V}\right)^{B_0'}}{B_0' - 1} + 1\right] - \frac{V_0 B_0}{B_0' - 1}. \tag{2.29}$$

2.6 VASP and the PAW Method

Since all *ab-initio* calculations presented in this thesis are performed with the Vienna Ab initio Simulation Package (VASP) [30, 49, 50, 51], in this section, the principle elements of the theory underlying the code are presented. Solving the self-consistent Kohn-Sham (KS) equations computationally requires expressing the problem, Hamiltonian and eigenstates, in terms of a basis. The choice of basis functions must be carried out judiciously in order to ensure accuracy and computational efficiency.

2.6.1 Plane Waves

VASP makes use of plane waves as a basis set, which are the subject of this section. Owing to the inherent translational symmetry of the underlying crystal structure, a common choice for solids is to represent the KS orbitals, φ_i, as plane-waves. Note that in VASP it is the pseudo-KS wavefunctions that are expanded as plane waves, as will be explained below. For convenience, in this section it is the KS-orbitals that will be expressed over a plane wave basis.

Consider a crystalline solid-state system with an underlying lattice and a volume V. Let $\vec{T}$ denote a general lattice translation vector and $\vec{G}$ a reciprocal lattice vector of the corresponding reciprocal lattice. The material is periodic and the KS Hamiltonian is translationally invariant with respect to translations $\vec{T}$. In general, the KS orbitals could be expanded in terms of plane waves, as these form a complete basis. Here, however, the KS-orbitals follow the Born-von-Kármán periodic boundary conditions at the crystal boundaries. The wavefunctions obey Bloch's theorem [52] and can be expressed as a specific set of plane waves. Instead of the index i, each orbital is specified by two indices, $\vec{k}$ and n_b, the crystal momentum and the band index of the eigenstate respectively. These can be treated as parameters of the eigenstates. Typically, $\vec{k}$ takes continuous values in the first Brillouin Zone (BZ). The KS-orbital thus becomes $\varphi_{n_b,\vec{k}}$, expressed as the product of a plane wave and a periodic function $u_{n_b,\vec{k}}$ with the periodicity of the lattice

$$\varphi_{n_b,\vec{k}}(\vec{r}) = u_{n_b,\vec{k}}(\vec{r})\, e^{i\vec{k}\cdot\vec{r}}, \tag{2.30a}$$

$$u_{n_b,\vec{k}}(\vec{r}) = \sum_{\vec{G}} U_{\vec{G},n_b,\vec{k}}\, e^{i\vec{G}\cdot\vec{r}}. \tag{2.30b}$$

The second equation explicitly shows the expansion as a Fourier series (i.e., with plane waves) with the periodicity of the lattice, where $U_{\vec{G},n_b,\vec{k}}$ are the expansions coefficients. The expansion is not over all plane waves, but only those with the periodicity of the crystal lattice, with wave vectors $\vec{G}$. Note that all quantities with the same period, such as the KS effective potential and the electron density, can be expressed in the same form as Equation (2.30b). Additionally, the KS equations are solved independently for each $\vec{k}$ and the energy spectrum obtained for each $\vec{k}$ is associated with the band index n_b. The set of plane waves with wave vector $\vec{G}$ thus form the basis, which is orthonormal. The sum in Equation (2.30b) is in practice cut-off at $\frac{\hbar^2}{2m_e}\left|\vec{G}+\vec{k}\right|^2 \leq E_{\text{cut}}$, with E_{cut} being the plane wave cut-off energy. This is performed in order to have a finite basis set. A large E_{cut} increases the accuracy of the representation and hence of the calculations.

2.6.2 K-Mesh

The calculation of many quantities, such as the electron density and the total energy of the system, require integrals over all values of $\vec{k}$ (in the first BZ). If the volume of the first BZ is V_{BZ}, the expression of the charge density in Equation (2.14) becomes an integral over all occupied states in the first BZ,

$$n(\vec{r}) = \frac{1}{V_{\text{BZ}}} \sum_{n_b}^{\text{occ.}} \int_{\text{BZ}} \left|\varphi_{n_b,\vec{k}}(\vec{r})\right|^2 d\vec{k}. \tag{2.31}$$

In practice, this integral is replaced by a finite weighted sum at certain k-points that approximates the integral above. These points constitute a uniform mesh across the first BZ. Most often, as is indeed the case for the calculations performed for this thesis, a Monkhorst-Pack mesh [53] is used, where the points are placed at $\vec{k}_{n_1,n_2,n_3}$ given by

$$\vec{k}_{n_1,n_2,n_3} = \sum_{i=1}^{3} \frac{2n_i - N_i - 1}{2N_i} \vec{G}_i, \tag{2.32}$$

Here N_i is the number of points sampled along the direction of the first BZ reciprocal lattice vector $\vec{G}_i$, and n_i takes values between 1 and N_i. The Monkhorst-Pack mesh presents the advantage of removing the highest symmetry points in the sum for even values of N_i, making the summation more efficient. Once the mesh is defined, the irreducible BZ (IBZ) [1] is established. Weights $w_{\vec{k}}$ are associated with each mesh point within the IBZ, which gives the total number of equivalent points to each IBZ k-point (including itself). It is divided by the total number of k-points considered N_k. The integral in Equation (2.31) thus becomes,

$$n\left(\vec{r}\right) = \frac{1}{N_\mathrm{k}} \sum_{n_b}^{\mathrm{occ.}} \sum_{\vec{k}}^{\mathrm{IBZ}} w_{\vec{k}} \left| \varphi_{n_b,\vec{k}}\left(\vec{r}\right) \right|^2 . \tag{2.33}$$

Plane waves offer several advantages over other basis sets, notably their orthogonality, their independence with respect to nuclear positions, given a fixed unit cell, their systematic improvability with higher values of E_cut, as well as their ability to efficiently model the smooth behaviour of valence electrons in the interstitial regions of solid materials, such as alloys. One of their main drawbacks is their inefficiency at representing wavefunctions with rapid oscillations, such as those encountered for both valence and core electrons in the atomic core, close to the nucleus. Many plane waves are required to accurately represent the presence of many nodes in the wavefunctions in this region. This makes plane waves as a sole choice of basis functions intractable for efficient and accurate first principles calculations. Instead, VASP makes use of the projector augmented wave (PAW) method, to still make use of plane waves.

2.6.3 PAW Method and Pseudopotentials

For the PAW method, the starting point is the frozen core approximation, in which all core electrons are treated as being "frozen" for the calculations. This implies that they are not explicitly considered and their state is pre-calculated. Their effect on the valence electrons is introduced through the pseudo-potential, for which the external potential of the nuclei are screened by the core electrons, as illustrated in Figure 2.2. However unlike other pseudo-potential methods [54, 55], for which the description of the core region is static, in the PAW method, the core region is dynamically updated

[1] from which the BZ can be re-created with the lattice's symmetry operations

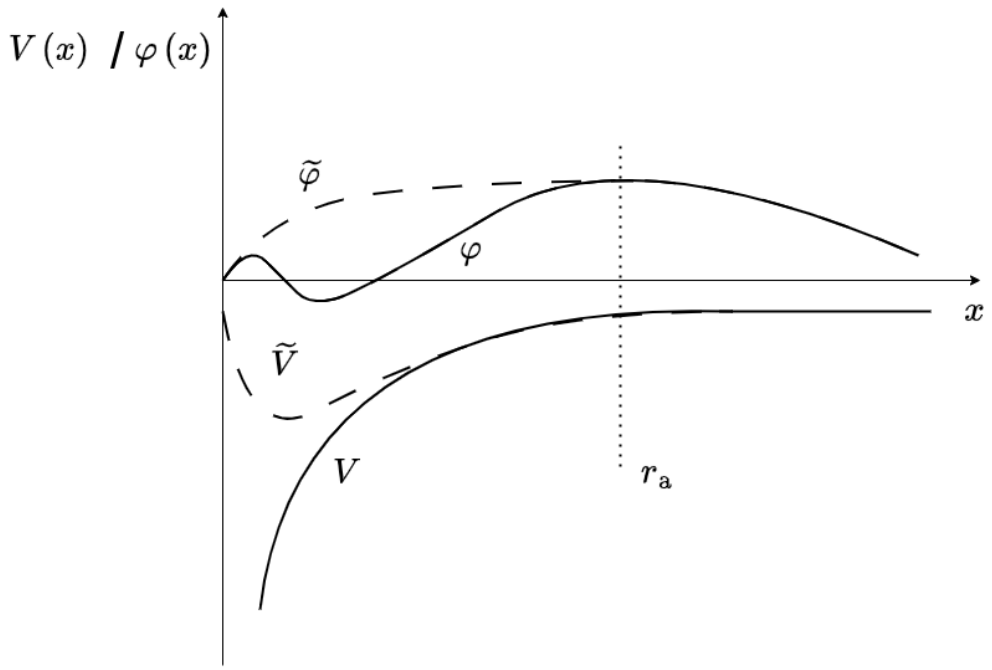


Figure 2.2: Illustration of the concept of the pseudo-potential $\widetilde{V}$, used instead of a typical Coulombic potential well V. The x-axis represents the distance from the nucleus. An illustration of the wavefunction obtained with (without) the pseudo-potential is also given as $\widetilde{\varphi}$ (φ). The augmentation sphere radius r_a is shown.

in a local basis set during the scf cycle. It is formally equivalent to an all-electron computation (within the frozen core approximation) [56], as the core density is reconstructed. Therefore, the full density is used to construct the Hamiltonian. How exactly the pseudo-potential enters the KS equations and the scf cycle is explained below. Note that, while the pseudo-potential is present in other methods, its exact construction here is specific to the PAW method.

The PAW method will be presented in Dirac notation in accordance with the typical notation. The aim is to construct a basis for the KS orbitals $|\varphi_i\rangle$ (i now references the indices $\vec{k}$ and n_b), referred to as the all-electron wavefunction. A central concept is that of the augmented sphere of radius r_a. Outside the sphere, in the interstitial region, the behaviour of the wavefunction is smooth and is represented exactly by a pseudo-wavefunction $|\widetilde{\varphi}_i\rangle$. The latter is a solution to the KS equations with the pseudopotential $\widetilde{V}$ used in place of the effective potential V_eff. This pseudo-wavefunction is expanded in plane waves. It is the expansion coefficients of the pseudo-wavefunction that are optimised during the KS self-consistent cycle. Within the augmentation sphere, the all-electron wavefunction varies more, and it cannot be modelled by the pseudo-

wavefunction. This is because $|\varphi_i\rangle$ varies smoothly, inside the augmentation sphere. Instead, the all-electron and pseudo-wavefunction are related by a linear transformation,

$$|\varphi_i\rangle = \widehat{T}\,|\widetilde{\varphi}_i\rangle\,. \tag{2.34}$$

This transformation operator is made of two parts. Firstly, of the identity operator $\widehat{T}$, such that the all-electron and pseudo- wavefunctions are equal outside the augmentation sphere. Secondly, of an operator that transforms the pseudo-wavefunction inside the sphere. Since $\widetilde{\varphi}_i$ behaves smoothly in this region, it needs to be expressed in terms of a more suitable atom-centered basis. $\widehat{T}$ is written as,

$$\widehat{T} = \widehat{I} + \sum_a \widehat{T}_a, \tag{2.35}$$

where the $\widehat{T}_a$ operator acts at the atomic site a to perform the transformation. The effect of $\widehat{T}_a$ vanishes outside the augmentation sphere surrounding each atom.

A second, atom-centered basis is thus required inside the augmentation sphere. The basis chosen for the on-site part of the wavefunction is made up of the three components: all-electron partial waves $|\phi_n^a\rangle$, the pseudo partial waves $|\widetilde{\phi}_n^a\rangle$ and the projection operators $|\widetilde{p}_n^a\rangle$. The operator $\widehat{T}$ is defined with respect to them as,

$$\widehat{T} = \widehat{I} + \sum_a \sum_n \left(-|\widetilde{\phi}_n^a\rangle + |\phi_n^a\rangle \right) \langle \widetilde{p}_n^a|\,, \tag{2.36}$$

which combined with Equation (2.34) gives,

$$|\varphi_i\rangle = |\widetilde{\varphi}_i\rangle + \sum_a \sum_n \left(-|\widetilde{\phi}_n^a\rangle \langle \widetilde{p}_n^a|\widetilde{\varphi}_i\rangle + |\phi_n^a\rangle \langle \widetilde{p}_n^a|\widetilde{\varphi}_i\rangle \right). \tag{2.37}$$

These new localised functions need to be defined. There are two important points to note regarding them. The first, is that they are all pre-computed and are available with the pseudo-potential for calculations. The second is that they are expanded in terms of radial grids and thus are represented in real space. This is in contrast with the plane-wave expansion of the pseudo wavefunctions. In VASP, the all-electron partial waves are solutions of the radial scalar relativistic non-spin polarised the Schrödinger equation of each isolated atom. Other choices can be made for these, but they must generally

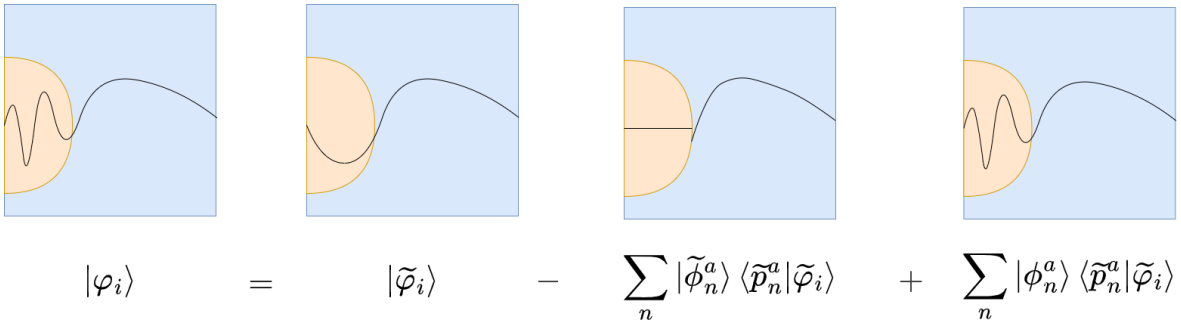


Figure 2.3: Illustration of the contributions to the all-electron wavefunction $|\varphi\rangle_i$ in the PAW method close to an ionic core. Note that the images show the contributions as they are added to each other, rather than showing the individual contributions independently. The orange region represents the augmentation sphere close to the nucleus and the blue region is the interstitial region.

model the bound states of the valence electrons accurately. In a crystal, they highly resemble the atomic ones, since they are localised and remain largely uninfluenced by the presence of other neighbouring atoms in a crystal. The pseudo-partial waves are in turn obtained by solving a pseudisation of the atomic Schrödinger equation for which the atomic pseudo-potential is used in place of the Coulombic potential of the nucleus. This yields a smooth function in the augmentation sphere, which vanishes outside of it. Finally, the projector operators simply need to satisfy the biothorgonality relationship, defined as,

$$\langle \widetilde{p}^a_n | \widetilde{\phi}^{a'}_{n'} \rangle = \delta_{a,a'} \delta_{n,n'}. \tag{2.38}$$

They are crucial in determining the efficiency of the calculations. From Equation (2.37), they must notably be efficient in performing the correspondence between the G-space representation for the pseudo-wavefunctions and the real-space representation of the partial waves. Bessel functions are used for this task in VASP [51].

Figure 2.3 illustrates the contribution of each term in Equation (2.37) to the all-electron wavefunction, the KS orbital $|\varphi_i\rangle$. The first term corresponds to the smooth part of the wavefunction in the interstitial region. The second term, which is a projection onto the pseudo-partial waves, cancels out the contribution of this pseudo-wavefunction inside the augmentation spheres. Finally, the last term expands the pseudo-wavefunction in terms of the all-electron partial wavefunctions to obtain the bound states. This separation of three terms is in fact carried through, when applying

operators to obtain expectation values, as exemplified in the expression for the energy of KS orbitals, that is expressed as,

$$\epsilon_i = \widetilde{\epsilon}_i - \widetilde{\epsilon}_i^1 + \epsilon_i^1, \tag{2.39}$$

where the terms respectively represent the energy contributions (Hamiltonian expectation values from each part of the wavefunction) of the individual terms given above. The first term is thus expressed in plane waves while the other terms are evaluated in a localised basis set.

Since the transformation from the pseudo-wavefunction to the all-electron one only requires a pre-computed linear transformation, as in Equation (2.34), for calculations on a crystalline material, only the pseudo-wavefunction needs to be determined, along with the potential. The KS equations are thus solved with the crystalline pseudopotential as an effective potential.

2.6.4 VASP Scheme

Following the discussion of the PAW method, VASP's KS scf cycle is summarised here. For the most part, the KS scheme presented in Figure 2.1 is followed. There are four important changes/details:

- The KS equations and energy functionals are adapted to the PAW method. Pseudo-KS orbitals are thus used in place of the all-electron KS orbitals.
- The KS pseudo-orbitals and the KS equations are expressed in terms of a matrix equation following the expansion in plane waves.
- The new KS equations are solved through an iterative diagonlisation scheme to refine the wavefunctions $|\widetilde{\varphi}_i\rangle$.
- The density update step uses charge density mixing.

The first two steps follow from the expressions presented in the previous subsection. Regarding the third point, an exact diagonalisation of the Hamiltonian expressed with N_{PW} plane waves is expensive ($O\left(N_{\mathrm{PW}}^3\right)$). Using iterative diagonalisation methods, only the N_{b} lowest eigenstates are determined (with $N_{\mathrm{b}} << N_{\mathrm{PW}}$). Two methodologies

employed for this are the residual minimisation scheme – direct inversion in the iterative subspace (RMM-DIIS) [57, 58] and the Blocked-Davidson methods [59, 60]. Finally, for the fourth point, it is inefficient to use the output charge density as a new input density during the KS scheme. Instead, the Broyden mixing scheme is used to update the charge density at each iteration, in which elements of the newly computed density and the previous input density are mixed together [61].

The main quantities that must be specified when running DFT calculations with VASP are the choice of pseudo-potentials used, the plane wave cut-off energy, which determines the size of the basis used, the energy stopping criterion for the KS scf scheme, the number of eigenstates N_b (related to the number of bands) that need to be computed in the diagonalisation step and the choice of k-mesh used for the calculations.

2.7 Spin

Magnetism is modelled by including spin in the calculations and is briefly described here, following Ref. [62]. For collinear magnetism the electron spin is arbitrarily considered only along the z-axis, and it is therefore assumed to either be up ($\uparrow$) or down ($\downarrow$), indexed by a spin index σ. The density is thus separable into up and down electron spin densities, defined with respect to the total spin density as,

$$n\left(\vec{r}\right) = n_\uparrow\left(\vec{r}\right) + n_\downarrow\left(\vec{r}\right). \tag{2.40}$$

All terms in the KS equations [Equation (2.20)] are attributed a spin index, such that two separate sets of Kohn-Sham equations are obtained for each spin. The spin densities are thus related to the spin-resolved eigenstates as,

$$n_\sigma\left(\vec{r}\right) = \sum_{i=1}^{N_e^\sigma} \left|\varphi_i^\sigma\left(\vec{r}\right)\right|^2. \tag{2.41}$$

Here, N_e^σ and φ_i^σ are the spin-resolved total number of electrons and KS orbitals, respectively. Crucially, the two densities can be different even in the absence of an external magnetic field, since the exchange-correlation functional is dependent on both densities, correlating them. The KS scheme is followed independently for each spin, as each iteration uses the spin densities of the previous round. The spatially-resolved

magnetisation $m(\vec{r})$ can thus be obtained as,

$$m(\vec{r}) = n_{\uparrow}(\vec{r}) - n_{\downarrow}(\vec{r}), \tag{2.42}$$

from which the total magnetisation follows from an integration over the unit cell.

2.8 Ionic and Structure Relaxation

A compound is considered to be in equilibrium when the forces on each atom and the stress tensor elements vanish (in practice, within a certain threshold). In such a case, the underlying structure is at some minimum of the potential energy surface (PES) of the Hamiltonian. The PES is a function of the parameters of Equation (2.4), namely the ionic core positions $\vec{R}^N$ and the lattice parameters. DFT can be used to drive a non-equilibrium (or unrelaxed) structure to an equilibrium (or relaxed) one, a local minimum in the PES, by optimising the structure. This requires the combination of the forces and/or stress tensors with an appropriate optimisation algorithm. In this section, the conjugate-gradient (CG) algorithm [63] for ionic relaxation will be outlined, as it is the algorithm of choice for the work presented.

Consider a system with an initial configuration $\vec{R}^N_{\text{init}}$, which sits in the vicinity of a local minimum $\vec{R}^N_0$ in the PES described by $E\left(\vec{R}^N\right)$. In this region of the PES, E can be approximated by,

$$E\left(\vec{R}^N\right) \simeq \vec{R}^N_0 + \frac{1}{2}\left(\vec{R}^N - \vec{R}^N_0\right)\mathbf{B}\left(\vec{R}^N - \vec{R}^N_0\right), \tag{2.43}$$

where $\mathbf{B}$ is the Hessian matrix of E with respect to each atomic coordinate $\vec{R}_I$. Since this is at a minimum, $\mathbf{B}$ is positive definite and the gradient of E with respect to each atomic coordinate, an atomic force component, vanishes. In practice, the Hessian is expensive to calculate and instead, the gradients are used, since these correspond to the forces and are available at the end of a scf cycle. Starting from $\vec{R}^N_0$, the positions are updated in the first step of the CG algorithm along the direction of the gradient of the PES, $\vec{s}_0$. The atoms are moved in the direction the forces acting on them point in. Their positions are determined with a line search algorithm [64], which uses a cubic interpolation to estimate the position of the minimum along the direction $\vec{s}_0$. Note that

this step requires several energy and force calculations. Once the minimum is found in direction $\vec{s}_0$, this process is repeated following the same steps, aside from the fact that the new directions $\vec{s}_i$ in which the atomic positions are moved is given by

$$\vec{s}_i = \vec{g}_i + \gamma \vec{g}_{i-1}, \tag{2.44a}$$

$$\gamma = \frac{(\vec{g}_i - \vec{g}_{i-1}) \cdot \vec{g}_i}{\vec{g}_{i-1} \cdot \vec{g}_{i-1}}, \tag{2.44b}$$

where $\vec{g}_i$ is the gradient calculated at step i. This is repeated until the forces reach a convergence criterion, set by a threshold value, a parameter of the relaxation in VASP. The direction of the search $\vec{s}$ requires to be updated K+1 times to find the minimum of a PES with K degrees of freedom. For the optimisation of lattice parameters, the same algorithm is followed but with respect to shears $\epsilon_{\alpha\beta}$. The energy derivatives with respect to $\epsilon_{\alpha\beta}$ are the stress tensor elements, used as the gradients for the CG algorithm.

2.9 Conclusion

In this chapter, density functional theory was introduced as a means to practically and accurately calculate the electronic structure of crystalline materials. Based on the works of Hohenberg-Kohn and Kohn-Sham, it was shown how the problem of describing an interacting N_e-electron system can be reduced to solving a problem with N_e-non-interacting particles with the same ground-state density and energy. A practical methodology employed by the VASP DFT package for solving the Kohn-Sham (KS) equations was also presented. This is based on the use of a plane-wave basis set and projector augmented wave pseudopotentials formalism. The algorithms employed in the self-consistent cycles were also outlined. DFT can thus provide a means to calculate the total energy, atomic forces, stress tensors and bulk moduli of a compound. It can also perform a relaxation towards the equilibrium structure of a system. Despite such capabilities, the computational time of such calculations are high and scale as $O(N^3)$ with system size [33]. It is revealed how this computational burden can be alleviated by using machine learning interatomic potentials in the next chapter.

3

Machine Learning and Interatomic Potentials

The aim of this chapter is to introduce the concept of machine learning interatomic potentials (MLIAPs), also referred to as machine learning force-fields (MLFFs). This first requires an overview of machine learning methods (Section 3.1), along with two common models, ridge regression (Section 3.2) and Neural Networks (Section 3.3). This is followed by the definition of interatomic potentials (Subsection 3.4.1), before introducing common MLIAPs. The notation for Section 3.1 is completely independent to that in Chapter 2.

3.1 Machine Learning

A commonly used definition of the term machine learning (ML), attributed to Arthur Samuel, a pioneer in the area, is the "field of study that gives computers the ability to learn without being explicitly programmed". It groups computer models that have in-built flexibility, in the form of parameters, such that their outcome depends on the data presented to them. It is a sub-field of artificial intelligence (AI). There are several categories of ML models, notably unsupervised [65], and reinforcement learning [66] among many others, but the most common type is supervised learning.

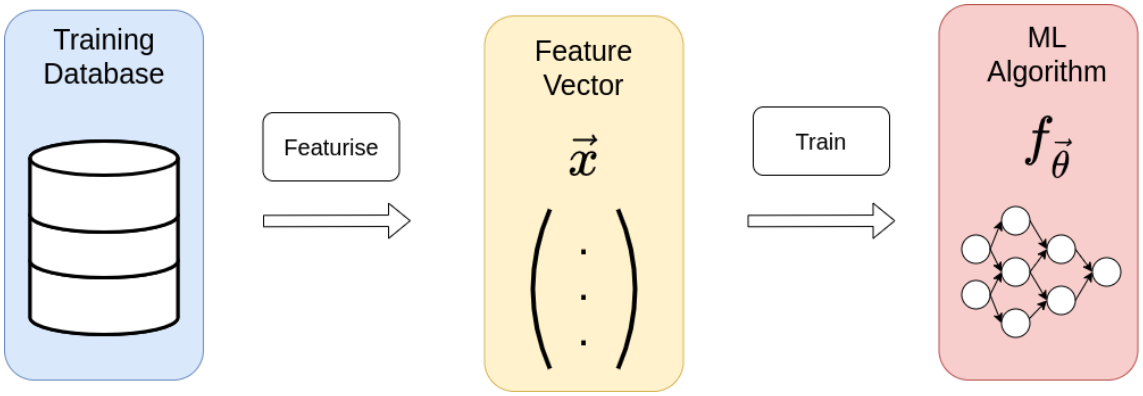


Figure 3.1: Graphical summary of the important steps in training a machine learning model.

3.1.1 Supervised Learning Overview

For models that fall under this branch of ML, the aim is to be able to make predictions of a target quantity given an input. Mathematically, one wishes to construct a general function f, capable of mapping a *feature* vector $\vec{x}$ onto a target quantity $\vec{y}$, and has parameters $\vec{\theta}$ that can be tuned to increase the accuracy of the mapping,

$$f_{\vec{\theta}} : \vec{x} \rightarrow \vec{y}. \tag{3.1}$$

In general, neither the inputs nor the outputs have to be vectors, and they could be matrices or scalars. In fact for the discussion here, it will be assumed that the input is a vector $\vec{x}$ and the output is a scalar y. For such methods to be applicable for a given task, it is crucial that inputs can in some way be "featurised", i.e., represented as a vector (or a more generally as a tensor). The function f can take up many forms with parameters $\vec{\theta}$, which make them adaptable to different tasks. These parameters are tuned by using training data, for which input and output are known. Following the training of the model, it can be used to make predictions on new data, for which only inputs exist.

As a brief example, one could imagine creating a model in which a physics professor wishes to predict how each of the undergraduate students enrolled in their condensed matter course will perform in the upcoming summer exam paper. The target y is their mark on the exam. To do so, the professor has access to a database of student results from the previous years. They also have a database of information on all students (current and past) that they believe is enough to predict what the marks will be.

This full database of past students composes the **dataset** used to train the model. The characteristics used might for example be the number of lectures and tutorials attended, grades in previous exams, numbers of hours studied for the exam, number of questions asked to the lecturer, etc. These will populate the **feature vectors**, $\vec{x}$, for each student. A suitable **machine learning algorithm**, represented by $f_{\vec{\theta}}$, is **trained** by optimising the coefficients, or weights, $\vec{\theta}$, of the model, to accurately map $\vec{x}$ onto y for the training dataset (past students). Considering the simplicity of the feature representation a simple model such as ridge regression or a decision tree would be appropriate [67]. Once trained, the model can be used to predict the grades of the current (and even future) students. A graphical summary of the pillars in machine learning, in bold in this paragraph, is given in Figure 3.1.

The field has seen an incredible advent, most notably in the past decade, thanks to a combination of an advancement in model architectures (convolutional neural networks [68, 69], transformers [70], diffusion models [71]), specific hardware design (GPU, TPU) [72], and perhaps most crucially, the explosion of available data that can be used to train models [73]. The areas that have seen the most success recently are in the fields of computer vision [74], natural language processing [75], recommender systems [76], and so-called generative AI models that can be used to generate text [77, 78], as well as images [79] and speech [80]. These advanced models are able to perform complicated tasks, as they are trained on large databases and can have billions of parameters. The model architecture enables to extract the most meaningful features from the training data inputs during training. Due to this abstraction, these models are referred to as deep learning models. However, such models are not used in this work, as they require extensive training datasets, which are not typically available in many fields of physics.

3.1.2 Machine Learning Definitions

In this subsection, a number of terms employed in the field of machine learning are defined. The first is that of hyperparameters. These constitute all parameters involved in the construction of an ML model that are not explicitly optimised during the training process, i.e. parameters outside the weights $\vec{\theta}$. They can relate to the construction of the feature vectors (number of features), to the model architecture, or simply be parameters inherent to a model. They are kept fixed during any given train-

ing iteration, but are typically optimised by training a range of models with different hyperparameters and select the ones for the best-performing model.

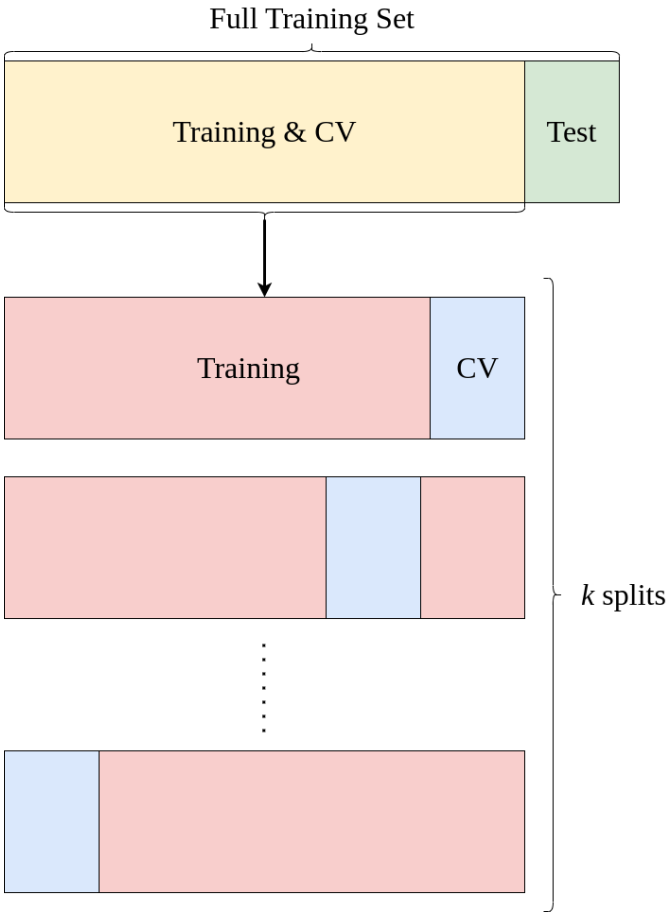


Figure 3.2: Illustration of the partition between training, cross-validation (CV) and test sets. The training and CV sets split k times according to a k-fold CV scheme.

In order to optimise these hyperparameters, the full training dataset (with feature vectors $\vec{x}$ and associated targets y) is split into two subsets. A training set and a cross-validation (CV) set. The former is used as described so far to train the model. The second is set aside, and instead it is used to optimise hyperparameters. The performance of models with different hyperparameter values is tested on the cross-validation set. The hyperparameter values for which the model performs best on the CV set are chosen as optimal ones. In practice, in order to avoid any bias in the model construction, the training and CV sets are not fixed. Instead, several training/CV splits are created following a given scheme. The process of training on the training set and testing on the cross-validation set is thus repeated for each partition. A popular cross-validation scheme, k-fold cross-validation, partitions the full dataset into k

training/cross-validation splits, using different subsets of the dataset for the CV set in every split. Each split has the same ratio of training and CV data points. The cross-validation methodology used in this work is the Monte Carlo CV, in which for each split, the data points of CV set is always determined randomly. Note that the proportion of data points used in the CV set, as well as the number of different training/CV splits used for the cross-validation scheme, are both hyperparameters.

In all cases, the dataset is in fact partitioned into a third "test" set. This set is set aside before model training and optimisation, and is never involved in the training process. Instead, it is used after the model is trained to evaluate the final, unbiased accuracy of the trained model. The partition between training, CV and test sets is illustrated in Figure 3.2.

So far, no metric has been given to assess the accuracy of a model. For supervised machine learning regressors, two common metrics, heavily employed here, are the mean-absolute-error (MAE) and the root-mean-squared-error (RMSE). Consider a dataset of m points, with target property y_i, predicted by the ML model as $\tilde{y}_i$ for data point i. The MAE and RMSE on the dataset are given by:

$$\mathrm{MAE} = \frac{1}{m}\sum_{i=1}^{m} |y_i - \tilde{y}_i| \,, \tag{3.2a}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m} (y_i - \tilde{y}_i)^2}. \tag{3.2b}$$

The MAE gives the direct average absolute error on each example in the dataset, while the RMSE uses the square root of the mean squared error. Since the latter depends on the square of the error of each dataset, it emphasises the error contribution of outliers, for which the model performs poorly.

Another important concept in the field of ML is that of a loss function. It corresponds to a function of the parameters $\vec{\theta}$ of a model, and is to be minimised during model training. It is therefore intended to evaluate how well a model performs on the training set. Similarly to the errors defined above, the loss function typically involves a measure of how different the target quantities y_i of the training dataset are to the predicted ones $\tilde{y}_i$. A typical choice for the loss function is the residual sum of squares, defined as,

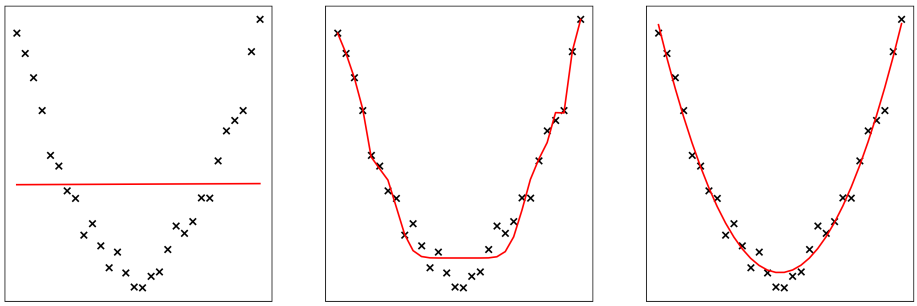


Figure 3.3: Plots illustrating the concepts of underfitting and overfitting. The output data (y-axis) follows a quadratic relationship with respect to the input data (x-axis) with Gaussian noise. In the left plot, a linear fitting is performed, illustrating the concept of underfitting. In the middle plot, a polynomial of order 20 is fit, illustrating the concept of overfitting. On the right, a quadratic fitting is shown, the right order polynomial for fitting.

$$L\left(\vec{\theta}\right) = \sum_{i=1}^{m} \left(y_i - \tilde{y}_i\right)^2, \tag{3.3}$$

where $\tilde{y}_i$ depends implicitly on $\vec{\theta}$.

One final aspect worthy of mention for this work concerns the concepts of underfitting and overfitting. Underfitting refers to an ML model not being able to adequately fit the complexity of the mapping of inputs $\vec{x}$ to outputs y. For such a model, the form of the function $f_{\vec{\theta}}$ is too simple or does not include enough parameters and thus is not able to accurately model the subtler aspects of the relationship between inputs and target. Such a model is said to have high bias and generally has poor performance on training and test sets. An example of this could be fitting a linear model to a dataset for which the target quantity has a quadratic dependence on the input, as illustrated Figure 3.3. Overfitting is the opposite scenario, in which the model is too sophisticated and assumes a functional form that fits the training data too closely, by essentially fitting noise in the data. This is problematic, since the model then generalises poorly on unseen data. It thus performs very well of the training data but poorly on the test data. Returning to the quadratic example, an overfitting model may assume a functional form too complex for the quadratic relationship, such as a higher order polynomial. This is illustrated in Figure 3.3.

Following this introduction to important concepts in ML, two classes of ML models

are now presented. The first is ridge regression, one of the simplest regressors which assumes a linear relationship between features and target. The second class concern neural networks, which can, in general, be far more complex and map highly non-linear relationships between $\vec{x}$ and y.

3.2 Ridge Regression

3.2.1 Model Definition

Let us consider a training dataset with m data points described by p features, and assume that the feature vectors now include the number 1 as their first entry. This is assumed for simplicity's sake, as will be explained below. Individual input vectors are denoted $\vec{x}_i$ and targets y_i. The j^{th} component of vector $\vec{x}_i$ is denoted x_{ij}. In ridge regression, just like for linear regression, the target quantity is assumed to vary linearly with respect to the components of the input vectors $\vec{x}_i$. For a single example, this relationship is given as

$$y = f_{\vec{\theta}}^{\text{Ridge}}(\vec{x}_i) = \vec{\theta} \cdot \vec{x}_i, \tag{3.4}$$

where $\vec{\theta}$ are the model coefficients. The first such coefficient is the intercept of the model, that multiplies the unity entry in each input vector. It acts as an additional degree of freedom in the fitting, independent of any "real" feature.

What distinguishes ridge regression to linear regression is how the linear coefficients are determined. Here, they are determined by minimising the loss function

$$L_{\lambda}^{\text{Ridge}}\left(\vec{\theta}\right) = \sum_{i=1}^{m}\left(y_i - \sum_{j=1}^{p} x_{ij}\theta_j\right)^2 + \lambda \sum_{j=1}^{p} \theta_j^2, \tag{3.5}$$

where θ_j are individual coefficients. The first term in this equation is identical to that of the loss function used for typical linear regression. The second term is additional and specific to ridge regression. It is a regularisation term, and corresponds to L_2 regularisation [67]. Note that the sum does not include the first fitting parameter, the intercept. The aim of such a term is to counteract overfitting by imposing a constraint on how large the fitting coefficients can be. Indeed, overfitting is enabled by having

full flexibility in the values the weights can take. As is the aim of a loss function, it is minimised with respect to fitting parameters and intercept, to provide optimal mapping of the function $f_{\vec{\theta}}^{\text{Ridge}}$ to the training set. The regularisation coefficient λ weighs the importance of this second term with respect to the first, fitting term. It is a hyperparameter of the model.

3.2.2 Matrix Notation

In order to show how the coefficients are determined, it is easier to adopt a matrix notation. Here, the intercept and the unity term in the input vectors are not included for the following equations. Following a transformation of the inputs, the intercept can be retrieved following optimisation of the other parameters, as detailed in Reference [67]. $\mathbf{X}$ is the $m \times p$ matrix of input examples, where each row corresponds to a training example with p features. The targets are grouped in a column vector $\vec{y}$ of length m. The coefficients remain $\vec{\theta}$, a column vector with length p. Equations (3.5) and (3.4) can respectively be written as

$$\vec{y} = \mathbf{X}\vec{\theta}, \tag{3.6a}$$

$$L_{\lambda}^{\text{Ridge}}\left(\vec{\theta}\right) = \left(\vec{y} - \mathbf{X}\vec{\theta}\right)^{T}\left(\vec{y} - \mathbf{X}\vec{\theta}\right) + \lambda\vec{\theta}^{T}\vec{\theta}. \tag{3.6b}$$

The optimal coefficients, $\vec{\theta}^{\text{Ridge}}$, that minimise the loss function are given by

$$\vec{\theta}^{\,\text{Ridge}} = \left(\mathbf{X}^{T}\mathbf{X} + \lambda\mathbf{I}\right)^{-1}\mathbf{X}^{T}\vec{y}. \tag{3.7}$$

In practice, inverting the matrix in Equation (3.7) is time-consuming for large datasets. Another approach to finding $\vec{\theta}^{\,\text{Ridge}}$ is to use optimisation algorithms, just like those presented in Chapter 2 for structural relaxation. In this case, however, the aim is not to find a minimum on the PES but to find a minimum of the loss function in the space of all parameters $\vec{\theta}$. Many methodologies exist for this [81], but most make use of at least the first derivative, and for some higher order derivatives, of the loss function. The simplest and most common of these methods is the gradient descent algorithm, introduced here.

3.2.3 Gradient Descent

In gradient descent, the optimal weights are found through an iterative process. At each iteration, they are updated to decrease the value of the loss function. This is performed by following the negative gradient of L_λ^{Ridge} with respect to parameters $\vec{\theta}$. This corresponds to the derivative of the loss function with respect to each weight θ_j:

$$\frac{\partial L_\lambda^{\text{Ridge}}\left(\vec{\theta}\right)}{\partial \theta_j} = -2\sum_{i=1}^{m} x_{ij}\left(y_i - \sum_{j=1}^{p} x_{ij}\theta_j\right) + 2\,\lambda\,\theta_j, \tag{3.8}$$

or in matrix notation, the gradient vector, $\vec{g}^{\,\text{Ridge}}$, is given as,

$$\vec{g}_\lambda^{\,\text{Ridge}} = -2\mathbf{X}^T\left(\vec{y} - \mathbf{X}\vec{\theta}\right) + 2\lambda\vec{\theta}. \tag{3.9}$$

The new coefficients $\vec{\theta}'$ can then be updated from the previous ones $\vec{\theta}$ as,

$$\vec{\theta}\,' = \vec{\theta} + \eta\,\vec{g}_\lambda^{\,\text{Ridge}}. \tag{3.10}$$

η is the learning rate, a hyperparameter that controls how fast the coefficients can change upon each iteration. This iterative update is typically performed until the loss function stops decreasing more than a certain threshold value after an iteration. The parameters are assumed to be the optimal ones after reaching convergence. In the case of ridge regression, solving Equation (3.7) or using gradient descent to find the optimal weights is equivalent [82]. However, gradient descent scales better for larger datasets.

Since gradient descent drives the search of the best parameters to the closest minimum in the loss function, it is prone to find local minima rather than global minima in the case of a non-convex search space. This can in part be address using mini-batch and stochastic gradient descent. In general however, while such an optimisation algorithm may be appropriate for simple linear models such as ridge regression, it is not well suited for optimising non-linear models such as neural networks.

3.3 Neural Networks

The other class of machine learning algorithms described in this chapter is that of artificial neural networks (NN). These can take several forms, such as convolutional

neural networks, recurrent neural networks or graph neural networks, but here the discussion will focus on one, most common type of NNs, feed-forward NNs. As such, NNs will refer to feed-forward NNs from now on. In this section, the architecture, mathematical formalism and optimisation methods of NNs will be discussed.

3.3.1 Model Definition

A Neural Network can be represented as a network of nodes, arranged in different layers, and edges, connecting the different nodes. There are three different types of layers, the input layer, the hidden layers and the output layer. Layers are labelled k, from 1 to K (the total number of layers). The input layer corresponds to a layer with a number of nodes equal to the number of input features, or p in our case. The output layer is a single node which corresponds to the output (predicted target value) of the model. Finally, the hidden layers are all other layers, pictured to be between the input and output layers. Nodes of layer k are only connected to other nodes, present in different layers, either preceding (layer k-1) or succeeding (layer k+1) that layer. Each node has an associated number, the activation, $a_l^{(k)}$, of the node, obtained from the activations of the nodes in the previous layer. For the input layers, the activations are just the entries of the feature vectors, input into the model. The data is fed-forward from the input layer to the other ones. The edges also have associated numbers, corresponding to the weights of the model. A general representation of a feed-forward neural network is given in Figure 3.4. The feed-forward element of the NN is highlighted by the fact that the arrows only point from a preceding to a succeeding layer, and not the opposite. Note that the NNs are fully connected, as all nodes of layer k have edges with all nodes in layer k+1. The number of hidden layers, as well as the number of nodes in each layer, are both hyperparameters of a NN.

Each value $a_l^{(k)}$, depends on the linear combination of the activations $a_{l'}^{(k-1)}$, stored on all nodes, labelled l', of layer $k-1$. The linear combination is called the logit $z_l^{(k)}$. This relationship is in general, written as,

$$a_l^{(k)} = \sum_{l'}^{\substack{\text{Nodes} \\ \text{Layer } k-1}} h^{(k)}\left(\theta_{ll'}^{(k)} a_{l'}^{(k-1)} + \theta_{l0}^{(k)}\right) = \sum_{l'}^{\substack{\text{Nodes} \\ \text{Layer } k-1}} h^{(k)}\left(z_l^{(k)}\right), \tag{3.11}$$

where $h^{(k)}$ is an activation function, added to build in non-linearity. It is the same

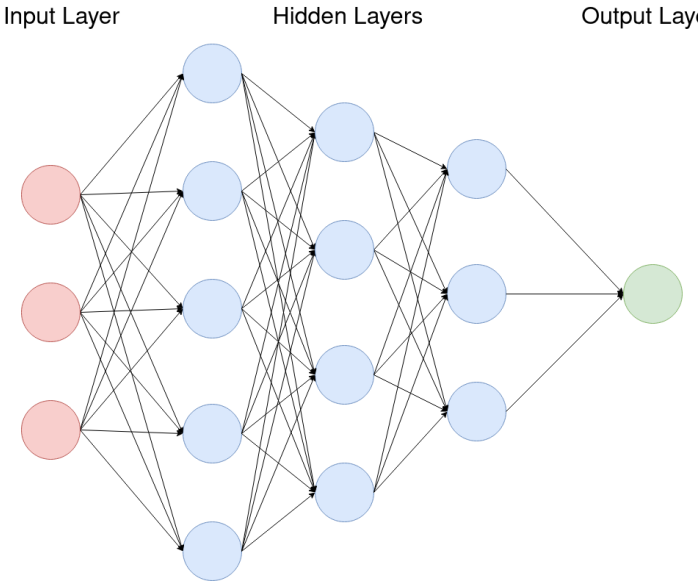


Figure 3.4: Graphical representation of a fully-connected feed-forward neural network. There are three hidden layers, with 5, 4 and 3 nodes. The input layer has 3 nodes.

activation function for all nodes in a given layer. $\theta_{ll'}^{(k)}$ are the single weights on the edge between the node l' of the layer k-1 and the node l of the layer k. $\theta_{l0}^{(k)}$ are the intercept values for node l on layer k. In matrix notation, the activations of all nodes in a layer are grouped into a vector denoted $\vec{a}^{\,(k)}$. The feed-forward from the values $\vec{a}^{\,(k-1)}$ in layer $k-1$ to those in layer k, $\vec{a}^{\,(k)}$, can be written as,

$$\vec{a}^{\,(k)} = h^{(k)}\left(\boldsymbol{\theta}^{(k)}\vec{a}^{\,(k)} + \vec{\theta}_0^{(k)}\right). \tag{3.12}$$

The weights are now arranged in a matrix $\boldsymbol{\theta}^{(k)}$, with elements $\theta_{ll'}^{(k)}$. The intercepts $\theta_{l0}^{(k)}$ form the so-called bias vector, $\vec{\theta}_0^{(k)}$. The activation function is applied element-wise to all values in the vector argument.

These are the equations that govern the relationship between subsequent layers. Hence, using them from layer to layer, the inputs can be fed-forward to the output, such as to create the general mapping from $\vec{x}$ to y. Different non-linear functions can be used as activation functions. They are typically zero or rapidly converge to a constant value for negative arguments, and converge to the $f(x) = x$ function for positive arguments. Popular choices include the rectified linear units (ReLU) function [83], exponential rectified units (ELU) function [84] and the Softplus function [85]. Their graphical representations are plotted in Figure 3.5. Note that neural networks are universal approximators [86], so if large enough, can represent any function. For

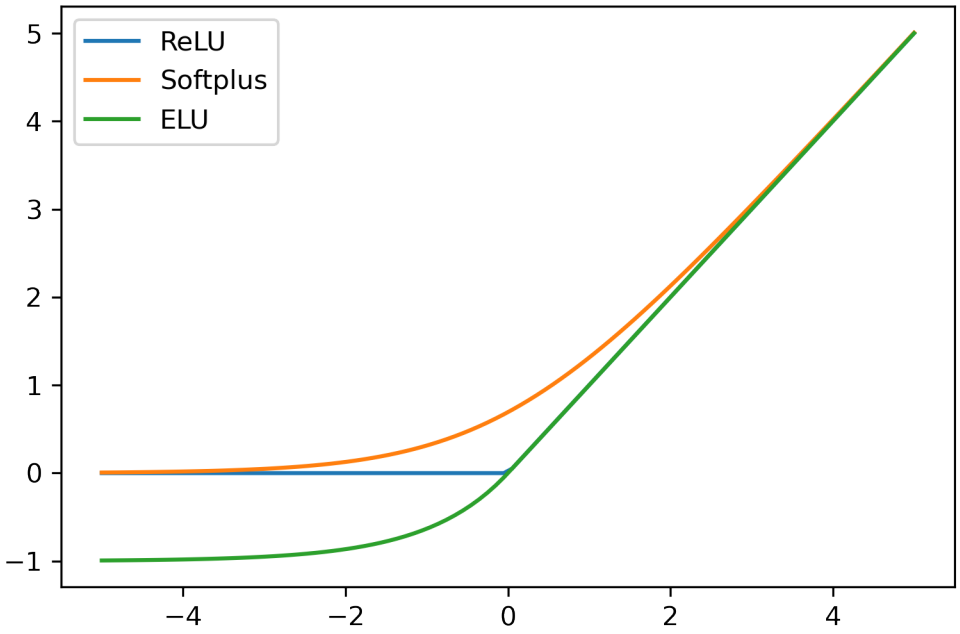


Figure 3.5: Graphical representation of the non-linear activation functions, ReLU, Softplus and ELU.

this mapping to be accurate, the weights on all edges need to be optimised. This is done with backpropagation.

3.3.2 Backpropagation

Backpropagation [87] is the methodology employed to update the weights of the NN during the training process. In the case of ridge regression, the weights could be directly optimised in terms of the error on the output, expressed in the loss function. This may be possible for NNs in the case of the last hidden layer, as it is directly connected to the output layer. But for all other hidden layers, knowing how to relate the error "back" to the weights is non-trivial. Hence, the use of backpropagation. The derivation that follows is for a single training example.

The aim is to find how each weight $\theta_{ll'}^{(k)}$ should be updated. To use a method such as gradient descent to drive the update steps, the gradient of a loss function L with respect to each weight [such as the one given in Equation (3.3)], is required. One must first define the error term for each node, which corresponds to the change of the loss

function with respect to the logit $z_l^{(k)}$ of each node, which is simply $\frac{\partial L}{\partial z_l^{(k)}}$. Since the loss function explicitly depends on the activation of the output layer (the final output of the model), it is of interest to express this derivative with respect to $a_l^{(k)}$, as,

$$\frac{\partial L}{\partial z_l^{(k)}} = \frac{\partial L}{\partial a_l^{(k)}} \frac{\partial a_l^{(k)}}{\partial z_l^{(k)}}. \tag{3.13}$$

Let us first consider the error of the output layer. The first term above is the derivative of the loss function with respect to the output, which is easily obtained (see Equation (3.8) for an example). Similarly, the second term is simply the derivative with respect to the activation function $h^{(k)}$ of the output layer, which can readily be computed, since all such functions are analytically defined. Thus, the error on the hidden layer is obtained. The task is then to propagate this back to the preceding layer, to obtain the error $\frac{\partial L}{\partial z_{l'}^{(k-1)}}$, on node l' of layer $k-1$. This can be written as,

$$\frac{\partial L}{\partial z_{l'}^{(k-1)}} = \frac{\partial L}{\partial z_l^{(k)}} \frac{\partial z_l^{(k)}}{\partial z_{l'}^{(k-1)}}. \tag{3.14}$$

The first term is known since it is the error on the output layer. Using the chain rule, the second term can be written as,

$$\frac{\partial z_l^{(k)}}{\partial z_{l'}^{(k-1)}} = \frac{\partial z_l^{(k)}}{\partial a_{l'}^{(k-1)}} \frac{\partial a_{l'}^{(k-1)}}{\partial z_{l'}^{(k-1)}}. \tag{3.15}$$

As for Equation (3.13), the second term is the derivative of the activation function $h^{(k)}$. The first term is the derivative of the logit $z_l^{(k)}$ with respect to the activation $a_{l'}^{(k-1)}$ of a node on the $k-1$ layer. Since the logit is just a linear combination of activations, the derivative is just the weight of the edge joining node l' of layer $k-1$ and node l of layer k, namely $\theta_{ll'}^{(k)}$. The final equation for the error in Equation (3.14) is, thus,

$$\frac{\partial L}{\partial z_{l'}^{(k-1)}} = \frac{\partial L}{\partial z_l^{(k)}} \theta_{ll'}^{(k)} h'\left(z_l^{(k')}\right). \tag{3.16}$$

This corresponds to the error on node l' of the final hidden layer. In fact, since in Equations (3.14) - (3.16), layers were referenced as $k-1$ and k, these equations are valid for any hidden layer. Starting from the output layer, the error can be backpropagated to the first hidden layer, and the errors on these nodes can equally be propagated to the penultimate hidden layer and so forth. Since the error $\frac{\partial L}{\partial z_l^{(k)}}$ on each node is known,

the derivative of the weights $\theta_{ll'}^{(k)}$ with respect to the loss function can now be expressed as

$$\frac{\partial L}{\partial \theta_{ll'}^{(k)}} = \frac{\partial L}{\partial z_l^{(k)}} \frac{\partial z_l^{(k)}}{\partial \theta_{ll'}^{(k)}} = \frac{\partial L}{\partial z_l^{(k)}} a_{l'}^{(k-1)}. \tag{3.17}$$

This is the equation required to update each weight. Note that the derivation presented considered the initial loss function for a single training point. It can be generalised for the partial or full training dataset.

From the expression of the gradient, any first-order optimisation scheme may be used to find the optimal weights for a neural network [81], of which the Adam optimiser [88] is a common choice [89, 90, 91]. Since neural networks, notably ones with many hidden layers, are non-linear, the weight optimisation procedure is crucial for adequate model performance and to prevent overfitting. Several methods could be used, such as mini-batch and stochastic optimisation, regularisation of the loss function (as for ridge regression), the use of drop out, max norm constraints, etc. [92].

3.4 Machine Learning Interatomic Potentials

Following the introduction of machine learning, common terms employed in when constructing ML models and presenting ridge regression and neural networks as examples of ML algorithms, the focus of the rest of the chapter will shift back towards physics. In the next sections, machine learning interatomic potentials will be presented, with a notable focus on atomic fingerprints, neural network potentials (NNPs) and the spectral neighbour analysis potential (SNAP), the latter which is heavily used in this work. Before concluding, references to other common MLIAPs are given. To begin, a definition of interatomic potentials is given.

3.4.1 Interatomic Potentials

Interatomic potentials are functions that map the energy of a system to the coordinates of its constituent atoms [93]. They approximate the PES of a system and their derivatives with respect to these positions correspond to the forces acting on the atoms, and thus constitute a force-field. Such potentials can then be used to predict any property

that depends on an accurate PES mapping. This could be to run structural relaxation, molecular dynamics (MD) simulations or to predict many quantities directly or through simulations, such as formation enthalpies, surface and interfacial energies, lattice parameters, thermal and elastic properties, compute phonon spectra and model inter- and intra-molecular interactions, even including chemical reactions in certain cases. The interatomic potential may depend on whether the chemical bonding relevant to a system is metallic, ionic, covalent, or dispersive in nature. Separate interatomic potentials may be used for each type, or a single one may be used to describe several simultaneously. They can also be physically motivated or derived from empirical data. In general, the potential energy U of a classical system, that depends on the positions $\vec{R}^N$ of the constituent atoms, can formally be expanded in terms of N-body terms,

$$U\left(\vec{R}^N\right) = \sum_I U_1\left(\vec{R}_I\right) + \sum_{I<J} U_2\left(\vec{R}_I, \vec{R}_J\right) + \sum_{I<J<K} U_3\left(\vec{R}_I, \vec{R}_J, \vec{R}_K\right) + \dots , \quad (3.18)$$

where the sums are over all two-body (U_2), three-body (U_3), etc. potentials. Two-body terms imply a dependence only on interatomic distances, three-body on distances and angles between three atoms, etc. An expansion of this form is used explicitly in the cluster expansion potentials [94, 95], truncated at a certain term. Many potentials take the form of two-body potentials, such as the Morse [96], Buckingham [97], Lennard-Jones [98] among many others. While these offer impressive insights onto real system behaviours, despite their simple dependence on interatomic distances, higher-order body terms are required to obtain more accurate descriptions, notably for covalent bonding. Commonly employed many-body potentials (more than two-body) include the Embedded Atom Method (EAM) [99, 100], second-moment tight-binding expansion (TB2M) and Tersoff potentials [101, 102].

By having a relatively simple functional form and thus being very fast to compute, these potentials are able to model large systems over long timescales, many of them quite robustly and with good transferability. However, this comes at the price of a decreased accuracy compared to first principles calculations such as DFT. Machine learning interatomic potentials (MLIAPs) offer an approach to combine the benefits of the speed of classical interatomic potentials and the accuracy of DFT. These are the

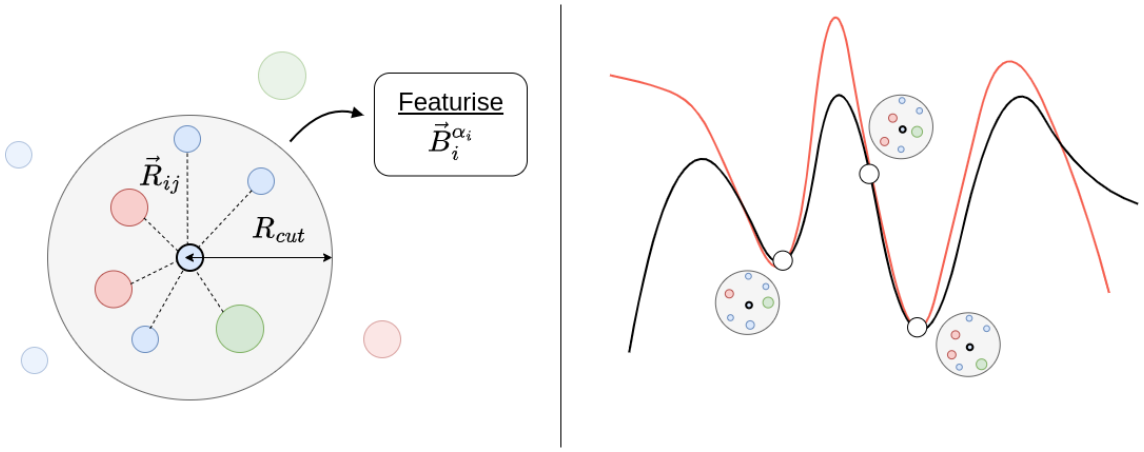


Figure 3.6: Illustration of a chemical environment (left) and the PES (right) mapped by DFT (black) and the MLIAP (red). The points on the PES represent training points, and are therefore in regions well described by the MLIAP.

focus of the rest of the chapter.

3.4.2 Concepts in Machine Learning Interatomic Potentials

The central concept in MLIAP is that of nearsightedness [103], which states that the most significant contributions to electronic properties arise from local interactions. The effects of so-called long range interactions are not negligible, but are of limited effect compared to local ones, as interatomic interactions tend to decay with distance. In MLIAPs, smoothness of the PES is also assumed [104]. In essence, they resemble empirical interatomic potentials, as they have free parameters that are found through fitting of data [93]. A key difference with MLIAPs is the form they assume, which is not restricted to any functional form and is made as general as possible. Instead, the built-in physical behaviour present for empirical interatomic potentials must be learnt by the MLIAP, through the relevant sampling of the PES.

As is the case for any ML model, there are three main ingredients for constructing an MLIAP, those presented in Figure 3.1. Here, the data take the form of high-fidelity energy and force values (typically computed with DFT) for different structures. Since the aim is to accurately map the PES in regions that the MLIAP is intended to probe, the choice of the structures sampled is vital. The latter must lie within and around the PES region of interest. The choice of these structures was physically-motivated for the first MLIAPs, taking not only ground-state structures of relevance, but also ones

with displaced atoms to obtain out-of-equilibrium structures. These provide forces (and stress tensor elements) that are important for fitting and mapping the PES around local minima. In fact, other structures such as interstitials, surface slabs, structures from MD runs, etc. are also commonly used in the training database and provide essential changes in the local environments of the atoms [105]. More recently, automatic schemes have also been employed to create training databases, that typically rely on metrics that quantify the diversity of a database [106, 107, 108, 109, 110, 111].

The second component concerns the feature vector or descriptor. For the majority of MLIAPs, such as those described in this chapter, these are associated with individual atoms, rather than being global descriptors of a structure. They must represent the so-called *chemical environment* of an atom, and therefore possess knowledge of its atomic neighbourhood. Crucially, they must be able to effectively represent these chemical environments, such that atoms in different structures have different descriptors. The atomic neighbourhood is usually considered up to a cut-off radius R_{cut} from the center of a given atom. Within the feature representation, N-body terms should be included, and thus they must depend on interatomic distances, angles, torsional angles etc. Figure 3.6 shows an illustration of an atomic chemical environment. In many cases, up to 3-body terms are represented, and they often encompass up to 4-body terms. There are a number of constraints on the descriptors. Since they are vectors, they must possess a fixed length, no matter the atom and environment, so that the single regressor can be used. To highly reduce the number of training points required, the descriptors have in-built symmetries. These include invariance with respect to translations and rotations of the system and with permutations of atoms of the same type. The descriptors are also differentiable with respect to atomic coordinates in order to compute forces, as for analytical interatomic potentials [104].

The final component of an MLIAP is the regressor. Its purpose is exactly the same as the ML algorithms presented previously, that is, to map the input descriptors to the output PES. This implies that any model can be used for this task. Linear models, neural networks and kernel methods are amongst the most commonly used regressors. The sketch of a PES mapping by a MLIAP with its training points is illustrated in Figure 3.6.

Mathematically, the general form that MLIAPs take follows the Behler-Parinello

approach [112]. As indicated above, this assumes that the total energy E of an N-atom system can be broken down into individual atomic energy contributions $E_i^{\alpha_i}$, written as,

$$E = \sum_i^N E_i^{\alpha_i}, \tag{3.19}$$

where α_i labels the atomic type. A regressor is then used to map the atomic descriptors $\vec{B}_i^{\alpha_i}$ to the atomic energies. The same ML model and architecture, represented by f, is kept for each atomic species, but the parameters of the model change depending on the atomic species. The relationship between model and descriptors can thus be written as,

$$E_i^{\alpha_i} = f_{\vec{\theta}^{\alpha_i}}\left(\vec{B}_i^{\alpha_i}\right). \tag{3.20}$$

This formalism is general for descriptor-based MLIAPs. In the following sections, two specific potentials will be introduced in detail, namely the neural network potential (NNP) and the spectral neighour analysis potential (SNAP). Most importantly, their descriptors will be presented, before giving an overview of their applications. NNPs are introduced since they were the first of the modern (2^{nd} generation) MLIAP developed and provide a clear example of how descriptors are constructed [112]. SNAP is the central model used in this work and is therefore also detailed. Note that many other, arguably better performing MLIAPs exist, but are not directly relevant here, and they are simply touched upon briefly in the final section.

3.5 Neural Network Potential

For NNPs, the features are based on so-called atom-centered symmetry functions (ASCFs) [112, 113, 114]. The first component, which is in fact common to many atomic features, is the cut-off function f_{cut}, which smoothly goes to zero at the cut-off radius R_{cut}. It is centered around the central atom for which the descriptors are calculated, and it is defined as

$$f_{\text{cut}}\left(R_{ij}\right) = \begin{cases} 0.5 \cdot \left[\cos\left(\frac{\pi R_{ij}}{R_{\text{cut}}}\right) + 1\right] & \text{for } R_{ij} > R_{\text{cut}}, \\ 0.0 & \text{for } R_{ij} \geq R_{\text{cut}}, \end{cases} \tag{3.21}$$

R_{ij} is the interatomic distance between atoms i and j. This function monotonically decreases with interatomic distance, and thus reduces the importance of an interaction with increasing distance. All atoms within a given cut-off region are considered to be in the neighbourhood of the central atom. The distances R_{ij} and the angles ϕ_{ijk} the neighbours form with the central atoms are collected. The simplest ASCF, G_i^1 is defined as

$$G_i^1 = \sum_j f_{\text{cut}}\left(R_{ij}\right), \tag{3.22}$$

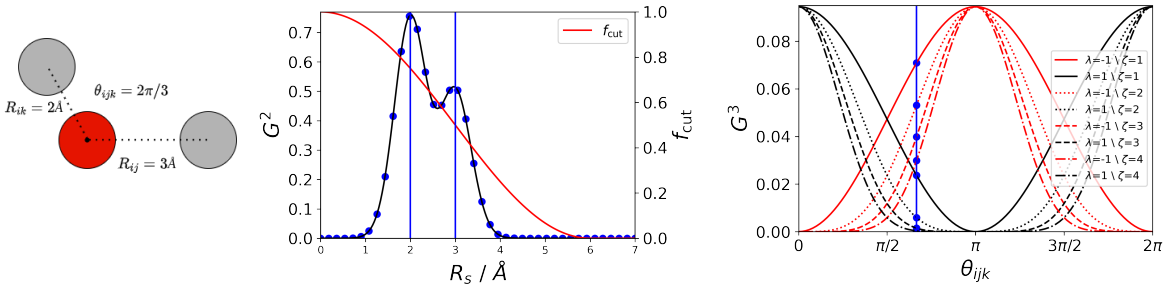
which corresponds to a coordination number for which atoms further from the central atom are less weighted by f_{cut}. The sum is over all neighbours within the cut-off sphere. Additionally, two-body ASCFs G_i^2 and the three-body ASCFs G_i^3 are used to respectively provide radial and angular distribution information. They are defined as sums within the atomic neighbourhood, as

$$G_i^2 = \sum_{j=1} e^{-\eta\left(R_{ij} - R_s\right)^2} \cdot f_{\text{cut}}\left(R_{ij}\right), \tag{3.23a}$$

$$\begin{aligned} G_i^3 = 2^{1-\zeta} \sum_{j=1} \sum_{k \neq j} \Big[& \left(1 + \lambda \cdot \cos\left(\phi_{ijk}\right)\right)^{\zeta} \cdot e^{-\eta\left(R_{ij}^2 + R_{ik}^2 + R_{jk}^2\right)} \\ & \times f_{\text{cut}}\left(R_{ij}\right) \cdot f_{\text{cut}}\left(R_{ik}\right) \cdot f_{\text{cut}}\left(R_{jk}\right) \Big]. \end{aligned} \tag{3.23b}$$

Each ASCF possesses what are considered to be hyperparameters, namely η and R_s for G_i^2 and ζ, λ (different to the regularisation parameter in ridge regression) and η. The two-body term can be viewed as a sum of Gaussians, each centred at R_s, with a width η. The height of the Gaussian depends on how close R_s is to R_{ij}. The sum in essence counts how many atoms are within a spherical shell, a distance R_s from the central atom, with width η. For the angular components G_i^3, the cut-off function remains but

must include the three interatomic distances. The Gaussian weighs the importance of interatomic distance for each angle. The cosine term serves to probe the angular distribution. λ takes values 1 and -1 to centre the cosines at $\theta_{ijk} = 0$ and π. ζ controls the widths of the distribution. This symmetry function can therefore also be viewed as shells of different widths over the angular distribution. Examples of the symmetry functions are given in Figure 3.7.



(a) (b) (c)

Figure 3.7: Example symmetry functions for the red atom in panel (a). Panel (b) shows how the value of G^2 varies as a function of hyperparameter R_s, that sweeps the radial neighbour distribution. The cut-off function f_{cut} is shown in red for $R_{\text{cut}} = 6.0$Å. Note η=4.0 Å^{-1}. The blue points show how different values of R_s may be chosen to construct a feature vector. The vertical blue lines mark the positions of the two grey atoms. Panel (c) gives basis symmetry functions for the angular values G^3 as a function of θ_{ijk} for different values of λ and ζ. Here, η=0.01 Å^{-1}. The blue vertical line marks $2\pi/3$, the angle in the example molecule. The blue points give the values that would be used for G^3 to construct the feature vector from each of the basis functions.

In order to construct a feature vector $\vec{B}_i^{\alpha_i}$, symmetry functions centred on atom i must be calculated: for G_i^1, each different atomic type $(\alpha_1, \alpha_2, ...)$; for G_i^2, each different species pair including type the central atom $(\alpha_i - \alpha_1, ..., \alpha_i - \alpha_i, ...)$; and for G_i^3, each species triplet including the central atom $(\alpha_i - \alpha_i - \alpha_1, ..., \alpha_i - \alpha_1 - \alpha_1, ...)$. This leads to a combinatorial explosion in the number of features and weights to fit with an increasing number of species [115]. For each G_i^1, G_i^2 and G_i^3, different hyperparameters are calculated. Details of how to effectively choose the hyperparameters is given in Reference [113]. The cut-off radius R_{cut} is kept fixed. Note that since the regressors are trained differently for each atomic type, the choice of hyperparameters (apart from R_{cut}) is in fact species-dependent. They must even depend on the type of interaction considered, so depend on the pair and triplet as well. Once the hyperparameters are chosen, the resulting symmetry functions essentially serve as basis functions. They are

then fixed to construct all descriptors (training and test).

In the NNP approach, the descriptors constructed with ASCFs are used in conjunction with neural networks, which have been covered in Chapter 3. The initial conception of NNPs only included the formalism presented here. However, since then, they have been further developed, notably resulting in the addition of electrostatic interactions [116], as well as other long-range interactions [117] and highly non-local dependencies of electronic properties on structural features [118]. NNPs have successfully been applied as interatomic potentials for large-scale systems in a diversity of contexts, such as for bulk systems [119], interfaces [120], nanomaterials [121] as well as for thermal conductivity studies [122], to cite a few. Following the discussion of NNPs, SNAP, a different class of MLIAPs, is introduced.

3.6 Spectral Neighbour Analysis Potential

In SNAP [28], rather than using simple descriptors combined with a highly non-linear model, as is the case for NNPs, ridge regression is employed and uses a more complicated feature representation. The starting point for the latter is the neighbour density function for each atom i, ρ_i. As for the ASCFs, the same cut-off function is used to define the nighbourhood of each atom. Within the radius R_{cut}, the neighbour density is given by,

$$\rho_i(\vec{r}) = \delta\left(\vec{r} - \vec{R}_i\right) + \sum_j w^{\alpha_j} \delta\left(\vec{r} - \vec{R}_j\right) f_{\text{cut}}(R_{ij}), \tag{3.24}$$

where $\vec{R}_i$ is the real space vector pointing to atom i and w^{α_j} are real-valued atomic weights that serve to distinguish different atomic types. They are treated as hyperparameters for the purpose of this work. In order to construct a feature vector, this function is expanded in terms of appropriate basis functions. Considering the spherical nature of the function, a natural choice for expanding the density are the spherical harmonics for the angular components, along with a radial basis. This comes with drawbacks, however. Having two different basis sets implies having to tune each one, as well as having coupling channels within and across both basis sets, leading to a large number of descriptors [123]. To alleviate this issue, Bartók *et al.* [124] showed

that it is in fact advantageous to project the density function in Equation 3.24 onto the higher dimensional 3-sphere. By projecting the 3-dimensional function on the unit hypersphere in 4D, both 3-dimensional radial and angular components (r, θ, ϕ) are encoded in 4-dimensional angular components (θ, ϕ, θ_0). Since the function is represented on the 4D unit sphere, only an angular basis is required, removing the need for a radial basis. The 3D polar radial coordinate r is projected onto an angle θ_0 as

$$\theta_0 = \pi \frac{r}{r_0}, \tag{3.25}$$

where $r_0 > R_{\text{cut}}$ determines how points are projected on the 3-sphere. Large values project many points to the north pole of the 3-sphere whereas at $r_0 = R_{\text{cut}}$, points would be projected onto the south pole, removing angular information. This value is kept fixed at 4/3, consistent with how it is originally introduced. The other two angles ϕ, θ are the same as the 3D polar angles (θ is not an ML weight). An illustration of such a projection in the case of a 2D circle is given in Figure 3.8. Note this is an illustration of a Riemann projection [124], defined slightly differently to the one used here but similar in nature.

A function on the surface of a four-dimensional sphere can be represented in terms of hyperspherical harmonics, which can be viewed as 4D versions of the regular spherical harmonics. These are denoted $U^J_{m,m'}(\phi, \theta, \theta_0)$, where J takes half-integer values and $m, m' = -J, -J+1, ..., J-1, J$. The expansion of the density in terms of this basis is written as

$$\rho_i(\phi, \theta, \theta_0) = \sum_{J=0}^{\infty} \sum_{m=-J}^{J} \sum_{m'=-J}^{J} u^J_{i,m,m'} U^J_{m,m'}(\phi, \theta, \theta_0), \tag{3.26}$$

where $u^J_{i,m,m'}$ are the expansion coefficients for the neighbour density of atom i. These are found by taking the inner product of the density with the basis function. Since the latter is a sum of weighted delta functions, this leads to the expansion coefficients being written as

$$u^j_{i,m,m'} = U^J_{m,m'}(0,0,0) + \sum_j f_{\text{cut}}(R_{ij})\, w^{\alpha_j} U^J_{m,m'}(\phi, \theta, \theta_0). \tag{3.27}$$

Creating a descriptor typically takes the form of using the expansion coefficients as

features. At this stage, the expansion coefficients are invariant upon translations of the system and permutations of identical atoms. Additionally, if the sum in Equation (3.26) is truncated, there is a fixed, constant number of coefficients for each atom. However, the coefficients are complex-valued and are not invariant upon rotations. This leads to the construction of the bispectrum components, B_{J,J_1,J_2}, which are real-valued and are invariant upon rotations [124]. They are defined in terms of the following triple inner product,

$$B_{J,J_1,J_2} = \sum_{m_1,m_1'=-J_1}^{J_1} \sum_{m_2,m_2'=-J_2}^{J_2} \sum_{m,m'=-J}^{J} \left(u^J_{i,m,m'}\right)^* H^{J,m,m'}_{J_2,m_2,m_2'^{J_1,m_1m_1'}} u^{J_1}_{i,m_1,m_1'} u^{J_2}_{i,m_2,m_2'}. \tag{3.28}$$

The indices J_1 and J_2 are restricted to $\|J_1 - J_2\| \leq J \leq \|J_1 + J_2\|$. The coupling coefficients, which correspond to the Clebsch-Gordan coefficients of SO(4), are related to the commonly used Clebsch-Gordan coefficients of SO(3) by,

$$H^{J,m,m'}_{J_2,m_2,m_2'^{J_1,m_1m_1'}} = C^{J,J_1,J_2}_{m,m_1,m_2} C^{J,J_1,J_2}_{m',m_1',m_2'}. \tag{3.29}$$

The bispectrum components can thus be used as features to describe atomic environments and respect the constraints and symmetries previously outlined. All components up to a value $J = J_{\max}$ are used to populate the feature vector, which corresponds to a truncation in the expansion. They quantify the strength of the correlation between three points on the 3-sphere, with lower order components describing coarser features and higher order ones are associated with more intricate correlations. Note that these descriptors account for up to 4-body interactions [125]. In practice, many of these components are zero or equal. It has been shown [126] that the number of unique bispectrum components in the feature vector $\vec{B}^{\alpha_i}$ is given by

$$\frac{(J_{\max}+1)\,(J_{\max}+2)\left(J_{\max}+\frac{3}{2}\right)}{3}. \tag{3.30}$$

Note this is the number per species. The number of different bispectrum components for different values of $2J_{\max}$ is given in Table 3.1.

In the linear SNAP formalism, the local atomic energies are written directly as a dot product between the descriptor vector and the model coefficients. Assuming a bias

term is included, the energy (Equation (3.19) for SNAP) and forces (written as the force on atom j) are written explicitly as

$$E = \sum_{i=1}^{N} \vec{\theta}^{\alpha_i} \cdot \vec{B}_i^{\alpha_i}, \tag{3.31a}$$

$$\vec{F}_j = -\sum_{i=1}^{N} \vec{\theta}^{\alpha_i} \cdot \frac{\partial \vec{B}_i^{\alpha_i}}{\partial \vec{R}_j}. \tag{3.31b}$$

The forces are written in terms of the derivative of the bispectrum components with respect to the atomic position of atom j. It is of interest to show what form the fitting of the model takes. In general, the weights of the model are optimised using the energy, forces, and stress tensor elements for the compounds in the training database. Here, stress tensors are excluded as they are not used for this work. For each DFT calculation, a single energy value and $3N$ forces are obtained. The full set of simultaneous equations can be written as associated to the fitting of the training set, omitting the contribution of the regularisation term, is written in matrix form as

$$\begin{pmatrix} \vdots & \vdots & \vdots \\ \sum_i^{N^{\alpha_1}} \vec{B}^{\alpha_1} & \dots & \sum_i^{N^{\alpha_K}} \vec{B}^{\alpha_K} \\ w_F \sum_i^{N^{\alpha_1}} \frac{\partial \vec{B}^{\alpha_1}}{\partial \vec{R}_1} & \dots & w_F \sum_i^{N^{\alpha_K}} \frac{\partial \vec{B}^{\alpha_K}}{\partial \vec{R}_1} \\ \vdots & \vdots & \vdots \\ w_F \sum_i^{N^{\alpha_1}} \frac{\partial \vec{B}^{\alpha_1}}{\partial \vec{R}_N} & \dots & w_F \sum_i^{N^{\alpha_K}} \frac{\partial \vec{B}^{\alpha_K}}{\partial \vec{R}_N} \\ \vdots & \vdots & \vdots \end{pmatrix} \cdot \begin{pmatrix} \vec{\theta}^{\alpha_1} \\ \vdots \\ \vec{\theta}^{\alpha_K} \end{pmatrix} = \begin{pmatrix} \vdots \\ E \\ \vec{F}_1 \\ \vdots \\ \vec{F}_N \\ \vdots \end{pmatrix}. \tag{3.32}$$

Here, w_F is a general "force weight" that weighs the contribution of the forces with respect to that of the energies. N^{α_1} is the number of atoms of type species α_1. There is a total of K different species. The weights are also divided according to species type. The equations for a single system are shown, and there are $3N$ terms for forces (dotted lines between the first and last atomic force), as opposed to the single energy data point per system. Note as well that in practice during the fitting, the bispectrum components of all atoms of the same species are summed together. There should be

the same equations above and below the terms shown for all compounds in the dataset.

SNAP provides a simple linear model that requires few features per compound to fit the PES, particularly if stress tensors and forces are not used for the fitting. SNAP has successfully been applied to model bulk single-element [127, 28] and multi-element alloys [128], liquids [28], ionic materials [129] and molecules [129]. Extensions have been introduced to include quadratic terms [126] and to have a better description of multi-element systems [130]. These have notably been used to simulate systems with 10^9 atoms [131], illustrating the computational efficiency of the model.

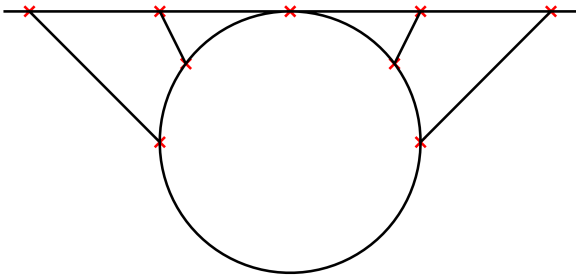


Figure 3.8: Example of a 2D Riemann projection.

Table 3.1: Number of bispectrum component features per species for different values of $2J_{\max}$.

$2J_{\max}$	2	4	6	8	10
# Bispectrum	5	14	30	55	91

3.7 Applications & Other MLIAPs

Considering their promise of high accuracy at a low computational cost, the past decade has seen an explosion in the development of MLIAPs. Research has focused both on finding novel, efficient representations and on new models. In this section, an overview of these developments is offered. NNPs are considered to be the first MLIAPs which combine local, invariant descriptors with a ML model. Using NNs has been a common approach for several MLIAPs [132, 133, 134, 135, 136, 137, 138, 139]. Other models following the same recipe soon after emerged, notably Gaussian approximation potentials (GAP) [140], which initially combined a Gaussian kernel between bispectrum component vectors as a representation with Gaussian Process Regression (GPR) [141]. Since, the Smooth Overlap of Atomic Positions (SOAP) descriptor was adopted to form the current implementation of GAP [124, 142]. Kernel methods notably offer a way to have non-linear mapping of the PES with respect to the representation for relatively few training points. GPR offers the additional advantage of offering an uncertainty

estimate on predictions. Other MLIAPs adopting GPR [108, 143] and kernel methods [144, 145, 146, 147] were since developed. SNAP [28] and qSNAP [126] are members of the same class of MLIAPs, but use sophisticated features with a linear model. Other linear models have the same approach, the most notable of which are the moment tensor potential (MTP) [148] and the atomic cluster expansion (ACE) [125]. The former uses basis functions built on moment tensor descriptors that rely on radial functions and outer products of atomic neighbour positions. The final descriptors are constructed as contractions of these tensors that provide rotational invariance and higher-body terms. For ACE, permutationally invariant descriptors are built from one-particle basis functions to form two-body terms and these are multiplied together for higher order basis functions. Crucially, such a description with a polynomial expansion of the atomic basis allows to mathematically represent *any* N-body function [such as those in Equation (3.18)] efficiently. These descriptors are shown to encompass the other ones described above and due to their completeness, they can be used with a linear model. Other efficient and flexible representations that can be used with a linear regressor have also been developed [149].

Tailor made descriptors offer enough expressivity and flexibility to be used with a large range of ML models. Another approach in constructing MLIAPs has revolved around using a deep learning model (NNs), for which the complexity of the model enables the determination of relevant features. This is in fact what is meant by deep learning [150]. This removes the necessity to construct sophisticated features and to ensure that enough many-body terms are included. The downfall is that large datasets are required for the training process. For these deep learning models, compounds and molecules are represented as graphs in which atoms are the nodes and bonds are represented by the graph edges. Such a representation is more natural than other ones used with NNs (such as images in CNNs). The deep learning models are graph neural networks, in the form of message-passing NNs, as was first introduced in 2017 [151]. Since then, similarly to the representation based methods, a plethora of models have emerged. Some of the most established ones include SchNet [152] and the directional message passing neural network (DimeNet) [153]. Other deep learning models that integrate equivariance have also become prominent [154, 155]. The newly developed MACE model combines the deep learning architecture with the ACE many-body ex-

pansion [156]. All of these approaches have centred around models for specific systems, but MLIAPs and especially graph neural networks have also been effective at performing screening in high-throughput searches [157, 158, 159, 160, 161, 162] and to be used as universal force-fields [163, 164]. M3GNet, released in late 2022, is even capable of performing energy screening and structure relaxation for materials containing atoms of a library of 78 different atomic species. In fact it is used in in this work, as will be presented in Chapter 5.

This work is, overall, largely focused on using the descriptor-based model SNAP. In a comparison between the state-of-the-art models [165], including NNP, GAP, SNAP, qSNAP and MTP (and not including ACE at that time), it was shown that all models can achieve an accuracy below 10^{-2} eV/atom on a standardised dataset. SNAP notably performed well with a low number of features. At their heart, these models are ideal for performing simulations over large length scales and long timescales, making MD simulations ideal. For such simulations, they have been used to model nucleation processes [166], amorphous phases [167], ionic conductivity [29] and fracture formation [168] to cite a few. As mentioned above, another area in which they have proved invaluable is in the search for new materials. They have been used as screening tools for high-throughput simulations [169], as a driving tool for finding new structures, either in combination with established models [170, 171, 172] or with novel methods [173, 174], and they have been used to directly construct convex hulls [175]. All applications which require lengthy DFT calculations for the predictions of energies, forces and stress tensors can be accelerated by MLIAPs, as is the case for calculating phonon spectra [176] and finite temperature and pressure phase diagram construction [177], as well as free energy calculations [178].

One of the limitations of these models is the fact that they can, by construction, only model local effects. Thus, long-range interactions and even more so non-local effects cannot be well captured. New formalisms have been developed in order to incorporate some of these effects, notably long-range interactions [116, 117]. The spin degree of freedom is also ignored in the representation, and adaptations of either the descriptors or the model to include some contribution of the spin-exchange interaction have been developed [179, 180, 181, 182]. There remains some difficulty here in generating a desired training dataset.

3.8 Conclusion

Two principal topics were covered in this chapter. Firstly, definitions and important concepts in machine learning were presented. Particular attention was brought to two specific ML algorithms, ridge regression and neural networks. The focus was then shifted to the new class of machine learning interatomic potentials. MLIAPs make use of many-body descriptors that described the local chemical environments of the constituent atoms of a system to map the PES. As for any ML model, they have parameters that must be tuned during a training process in order to accurately performing such mapping. The training data is provided by energies (as well as forces and stress tensors) of high-fidelity DFT calculations. Their accuracy thus typically matches that of DFT, coming at a fraction of the cost. However, they are only applicable for systems that resemble those of the training set. Different classes of MLIAPs were given, and there was a particular focus on descriptor-based models, most notably NNP and SNAP. The latter is central in the next two chapters, as it is used as a screening tool for the high-throughput search of novel ternary materials.

4

SNAP as a Screening Tool for Ternary Systems

4.1 Introduction

Two initial remarks are made before commencing the chapter. The first is that all the work presented here was done in full collaboration with Michail Minotakis. Alessandro Lunghi and Matteo Cobelli provided the code to train the machine learning interatomic potentials and were involved in the project development. The second is that this chapter is related to work described in a published article in Reference [31].

The search for novel materials is crucial to match the demand for ever-increasing performance in fields such as aerospace, advanced electronic devices and energy technologies. Following advancements in hardware performance and the development of user-friendly *ab-initio* algorithms for materials modelling [30, 183, 184], it has become feasible to utilise these computational methods to expedite the discovery process. Notable successes in theory-driven materials discovery include the advancements in Li-ion cathodes [185], high entropy alloys [186], and magnetic materials [18]. The current state-of-the-art workflow entails a high-throughput search (HT) [187], leveraging efficient density functional theory (DFT) calculations to predict material properties for a large virtual pool of compounds, with the hope of uncovering previously unknown "hidden compounds" [188]. Material databases house collections of such prototypes [19, 20]. The initial step in this workflow involves stability screening, ensuring that selected compounds are chemically viable and thermodynamically stable [189]. This

screening necessitates the calculation of a candidate system's total energy. The work presented in this chapter primarily focuses on demonstrating how machine learning (ML) can assist in this initial screening process.

Two essential elements are required to identify new stable phases: a method for generating candidate structures and a method for evaluating their energies. The former, not the primary focus of this study, is typically achieved by constructing a library of prototypes [18] or using dedicated methodologies for proposing novel structures [190, 191, 192]. Recently, ML has been used to accelerate this step [173, 193]. In contrast, the evaluation of total energies generally involves local/semi-local DFT calculations, which offer reasonably accurate results at a manageable computational cost. However, even with light approximations, *ab-initio* calculations remain the rate-limiting step in materials search workflows, constraining the number of candidate structures that can be tested and the size of their unit cells. To expedite this process, ML has been employed in various ways. For example, it has been used to predict improved starting charge densities for the self-consistent loop required by Kohn-Sham DFT [194, 195, 196]. ML has also been applied in active-learning frameworks to accelerate geometry optimization [197] and in *ab-initio* molecular dynamics (AIMD) [198] by directly predicting energies and forces. In these cases, ML models map local atomic configurations to energies and forces by training on DFT-provided training data. Once trained, the models can provide predictions significantly faster than DFT calculations. This work follows a similar approach by developing a machine-learning model that acts as a surrogate for energy predictions made by DFT, serving as a screening tool in high-throughput searches.

Several approaches have been employed to construct energy surrogates for DFT in the computational materials discovery process [189, 199]. One commonly used strategy is cluster expansion, where the energy of a system is expressed as a sum of energy contributions from various clusters formed by constituent atoms. The effective cluster interactions' strengths are determined by fitting the model to a set of DFT energies. Thus, this surrogate model leverages DFT data to interpolate for new structures [200]. This is similar to the approach taken by many ML tools, which can equally be used to predict energies based on interpolation of DFT data.

A multitude of studies have utilised ML in materials discovery, including compo-

nent predictions [201], forecasting the most likely compositions for a given set of atomic species, and structure predictions [202], determining the most probable crystal structure. These methods also extend to predicting a material's energy distance from the convex hull [203, 204]. Such techniques rely on databases containing diverse materials composed of different species and employ various machine-learning algorithms, such as Bayesian optimisers, random forests, and support vector machines. Feature vectors typically encompass general information about the material structure and species characteristics of the constituent atoms. However, they are not usually trained to differentiate between compounds with similar structures. For such tasks, machine learning interatomic potentials (MLIAPs) are relied upon. These have been extensively described in Chapter 3. Similar to the cluster expansion method, MLIAPs employ a DFT dataset to fit the model parameters and can predict energies and forces with *ab-initio* accuracy, provided that the structures fall within the model's interpolation range [165]. MLIAPs combine locally defined atomic configuration fingerprints, which are invariant under translation and atomic permutation, with ML algorithms to predict energies, forces, and stress tensors. This makes them ideal for high-accuracy molecular dynamics simulations involving large systems and long timescales [128, 167, 205, 206].

In the context of predicting material stability, MLIAPs are used to map the potential energy surface of multiple phases and construct the T=0 K phase diagram to determine the lowest energy structures, namely, constructing the convex hull. SNAP and NNP have previously been employed for this task for metallic alloys [207, 208, 209, 210]. However, SNAP's application was limited to a range of different structures and stoichiometries, and the potential was not used to discover new stable alloys. Conversely, NNPs relied on an extensive training set (approximately 10^4 structures), matching the number of DFT-computed structures needed to construct a fully *ab-initio* convex hull. The same holds for GAP models trained as general potentials across the phase diagrams of C [211] and Si [168]. The computational cost of kernel methods increases as $O(N^3)$ with training set size (using direct inversion of the kernel matrix), therefore limiting the application of typical kernel methods to a certain dataset size (typically $\sim 10^5$). Moreover, these methods plateau in performance beyond a given training set size (dependent upon the model size), leading to only marginal accuracy gains for increasing training set sizes [212]. To lower the cost of predictions for the

case of kernel methods and most importantly to reduce the computational and human effort of creating a training set, it is of interest to minimise the size of this data set.

Gubaev et al. [175] presented one of the few notable examples of utilising a MLIAP, in the form of a moment tensor potential, trained on a limited number of structures to predict materials stability at an accelerated pace. In their study, they conducted *ab-initio* calculations on a range of 383 to 2,393 structures to train an MTP capable of reproducing binary and ternary convex hulls. The selection of these structures was accomplished through an efficient active-learning process [106], involving the exploration of approximately 10^4 to 10^5 configurations for each phase diagram. This research demonstrated the ability of MLIAPs to accelerate the computational high-throughput search for new alloys.

In this work, it is shown how to build an effective MLIAP, SNAP, in the view of speeding up the search for new ternary intermetallic compounds. The aim is to build a rapid screening tool, which is similar in philosophy to the *specialised* MLIAP proposed by Artrith *et al.* to compute the binary convex hull of Li_xSi [213]. However, the approach here differs in terms of the training database selection. Instead of dedicating resources to curate a training set and conducting DFT calculations solely for MLIAP training purposes, we leverage existing materials convex hull databases, particularly the AFLOWlib repository [19]. This approach offers several advantages. Firstly, the data is readily available, and the computational efforts invested in *ab-initio* calculations are also pertinent to the construction of the convex hull. Within AFLOWlib, extensive exploration of binary phase diagrams has typically been conducted, resulting in a substantial amount of available data and a lower likelihood of discovering new stable binary alloys. In contrast, ternary hulls tend to be less explored, despite the combinatorial explosion of possible derivative structures that can be generated from prototype structures, providing greater room for exploration. Furthermore, for ternary systems, the enthalpy term in the Gibbs free energy remains significant compared to the entropic term at room temperature, as opposed to the case for quaternary or higher multi-species compounds [214], emphasizing the continued relevance of free-energy calculations. Consequently, the proposed method utilizes SNAP to guide the screening of the ternary space.

The chapter is organised as follows. At first, an overview of the theory and method-

ology regarding the construction of phase diagrams is presented, leading to the explanation of the concept of convex hulls. Following this, the AFLOWlib repository is introduced, along with the relevant software tools that are used for this study. In a third subsection, the Ag-Au-Cu ternary phase diagram, along with the associated Ag-Au, Ag-Cu and Au-Cu binary phase diagrams are described. These will serve as the reference system for the bulk of the work. After outlining these methodological aspects and reviewing the relevant literature, the results are presented. At first, the ensemble SNAP model employed, and the training databases are described. This is followed by the first benchmark tests performed for the training of SNAP on single binary systems. The main results are then shown, as a SNAP model trained on the three binary systems is used to make predictions on the ternary compounds. This is extended to different sets of ternary structures, such as to assess the full range of applicability of the devised screening tool. Further models for two other ternary systems, Mo-Pt-Ti and Cd-Hf-Rh are then assessed. After presenting the methodology followed for hyperparameter optimisation, a subsection is dedicated to the study of different error metrics, aimed at detecting when the MLIAP used performs extrapolation. This is crucial to evaluate the reliability of predictions and establish proactive measures for situations where they prove unreliable. The results are finally summarised in the conclusions section.

4.2 Identifying Stable Compounds

To put this work into context with the broader field of materials science and engineering, a brief recap of the physics underpinning phase diagrams is presented. This requires notions from thermodynamics [215, 216, 217]. The systems studied are characterised by their composition $\{x_k\}$, where each x_k represents the relative concentration of species k, as well as their other state variables, most importantly pressure P, temperature T and volume V. The system is closed, so there is no mass exchange with its surroundings. The different substances present (different atoms or molecules) are referred to as the components of the system. In the study of phase diagrams, the aim is to identify what type of phase the system forms under certain conditions, i.e. for given state-variable values. This is established when the system is in a stable equilibrium. A phase is defined as a perfectly homogeneous region of space made up by a material.

This includes phases such as solid, liquid and gas phases, as well as different microscopic structures of solid phases. Note that under certain conditions, the system could be a mixture of several phases (e.g. solid liquid mix). Phase diagrams include typical temperature/pressure phase diagrams for single-component systems (unaries), as well as temperature/composition phase diagrams used for alloy phase diagrams (binaries for two-components, ternaries for three components, etc.). In all practical applications, T, P and $\{x_k\}$ are held constant, since these can be controlled.

The shape of a phase diagram is underpinned by the thermodynamic competition between different phases. It is therefore necessary to establish which thermodynamic quantity determines the phase that a system will adopt. At constant T and P, it is the Gibbs free energy G. Given a system with internal energy U, the sum of its total potential and kinetic energy, and entropy S, in the absence of external fields, the Gibbs free energy is defined as

$$G = U + PV - TS. \tag{4.1}$$

The aim is to determine the direction of spontaneous change, which drives the adoption of the most stable phase. From the first law of thermodynamics, under constant T and P conditions, the change in internal energy is

$$dU = dQ - PdV, \tag{4.2}$$

with dQ being the heat exchanged. The change in Gibbs free energy can be expressed as

$$dG = dQ - TdS. \tag{4.3}$$

From the second law of thermodynamics, in a spontaneous process

$$dQ \leq TdS, \tag{4.4}$$

which, expressed in terms of the Gibbs free energy, yields,

$$dG \leq 0. \tag{4.5}$$

Thus, at constant T and P, a system will spontaneously change to minimise its Gibbs free energy. This implies that, in phase diagrams in which temperature and pressure are controlled, the stable phase(s) minimise the Gibbs free energy of the system. Not only can this explain the shapes of experimental phase diagrams, but it can also be used in the field of inverse materials design to search for novel stable materials.

The theoretical approach to establishing a phase diagram of a multi-component system would require a set of methodologies to calculate all relevant quantities in Equation (4.1) for all possible phases, including the liquid phase and different solid state phases. In theory, an infinite number of solid phases should be sampled [218]. However, this is limited in practice and the choice of phases modelled is guided by the knowledge of well-known phases of related phase diagrams. In the case of a three component system, this must be performed for ternary phases as well as for the associated binary and unary phases. From computed values of the Gibbs free energy for different phases, a phase diagram can be constructed using the phenomenological CALPHAD technique [219]. This, in fact, only relies on a database of thermodynamic properties, and thus can be used with experimental, as well as computational results.

In computational studies, the difficulty lies in computing Equation (4.1) accurately, in a way that reflects the physical reality of real materials, studied experimentally. Typically, all compounds studied are assumed to be perfect crystals, with a well-defined lattice, basis and composition, as well as infinite translational symmetry. Real materials differ as they are not infinite nor perfectly homogeneous, have surfaces, impurities, many types of defects, grain boundaries along with pores and cracks at larger length scales.

Aside from this evident approximation, calculating G accurately for a single crystal at finite T and P is still non-trivial and relies on approximations. The problem is typically recast by calculating the Helmholtz free energy F of a given phase, from which the Gibbs free energy is obtained using a Legendre transformation,

$$G = F + PV. \tag{4.6}$$

The calculation of F must account for the total internal energy of the system, as well as its entropy, which are influenced by several degrees of freedom, namely

electronic, vibrational, configurational and magnetic ones [220, 221]. The coupling of each degree of freedom also contributes to F. The calculation of the configurational free energy (different atomic arrangements on a parent lattice) increases the number of structures that needs to be sampled very significantly. Therefore, one common approach to calculate F is to select a fixed atomic decoration to represent a phase for the calculation of the other degrees of freedom and to calculate the configurational contribution separately. The free energy is then decomposed into different terms by means of the free energy Born-Oppenheimer approximation [222],

$$F = E_0 - TS^{\text{conf}} + F^{\text{vib}} + F^{\text{elec}} + F^{\text{mag}}. \tag{4.7}$$

Where, E_0 is the electronic energy at T=0 K of the representative structure, S^{conf} is the configurational entropy, F^{vib}, F^{elec} and F^{mag} are the vibrational, electronic and magnetic contributions to the Helmholtz free energy respectively. F^{vib} and F^{elec} account for finite temperature effects of the structure.

E_0 is computed using *ab-initio* calculations, often DFT, from which contributions to F^{elec} and F^{mag} can also be obtained. However, these require further DFT calculations, making the full calculation of F an intensive process. These are beyond the scope of this work, and the reader is referred to Reference [223] for further details on these, notably F^{elec} and F^{mag}. Separate methodologies exist for the calculation of the other terms, S^{conf} and F^{vib}.

Computing G through Equations (4.6) and (4.7) involves high accuracy calculations that take into account finite temperature and pressure effects, thus matching experimental conditions. However, this leads to a large compute time. For example, the special quasi-random structures (SQS) [224] and coherent potential approximations (CPA) [225, 226] methods can be used to estimate S^{conf}. They require several DFT calculations, of several supercells of the parent structure for the former or many primitive cells with embedded components for the latter. F^{vib} is commonly calculated within the quasi-harmonic approximation, which requires a knowledge of the system's phonon spectrum. This can be computed using the finite displacement supercell method [227, 228], requiring further *ab-initio* calculations, which are costly due to the number of atoms in the supercell, as well as the tight convergence criteria required.

In the context of a high-throughput study, computational efficiency is of the essence

and a balance between speed and accuracy is paramount. To increase the throughput of this work, here we only compute E_0, which is taken as the sole contributor to evaluate G. This means that the all results are formerly at T=0 K and P=0 Pa. E_0 is one of the main contribution to G, notably at temperatures neighbouring room temperature, particularly for binary and ternary systems. The approach of only using the total energy at T=0 K is a common one for high-throughput searches, since a single DFT calculation is required per phase [229, 230, 231, 232, 233]. Equations (4.6) and (4.7) are simply given to frame the current work in the wider context of material phase diagrams.

Since finite temperature effects are not considered, the superscript will be dropped for E_0. Note that G, U, E and the enthalpy H ($= U + PV$) are all equal at T=0 K and P=0 Pa. Determining which phase is stable is equivalent to minimising any of these quantities. In what follows, only the enthalpy H and the energy E will be used as an indicator of phase stability. Finally, the description given so far is based on thermodynamics and assesses the energetic stability of a compound. An issue with computational studies is that due to the virtual nature of the work, many (in fact most) compounds studied do not exist in the physical world. Other than thermodynamic reasons for the instability of a material, a phase studied may also be dynamically unstable, namely, its phonon spectrum may present imaginary frequencies. This would imply that atomic displacements about some positions of the equilibrium structure are favourable, and hence the structure would deform [234]. As for F^{vib}, this requires the calculation of phonon frequencies, leading to further DFT calculations. This step is not included as part of the assessment of stability here. It could be added as a final step after a high-throughput search based on thermodynamic stability. The latter is performed through the construction of convex hulls presented in the following subsection.

4.3 Convex Hull Construction

The focus is now turned towards the systems studied in this work, namely unary, binary and ternary alloys. The terms phase, compound and material are used interchangeably to designate systems of atoms with different compositions and/or underlying structure. A given crystalline material of K species (K=1 for unary, etc.) will be represented by

its underlying primitive unit cell with N atoms and total energy E. This is computed using DFT. As argued above, the systems that are considered thermodynamically stable will minimise $E_{\text{atom}} = E/N$. The division by the number of atoms is necessary as the systems are no longer infinite in size, as assumed in Section 4.2. A unit cell with a fixed number of atoms is used to represent each system. For different phases at a fixed composition $\{x_k\}$, it is easy to identify the most stable compound, corresponding to the one with the lowest value of E_{atom}. This is the stability criterion for unary systems. Binary and ternary systems first need to be lower in energy than the stoichiometric linear combination of energies of their constituent elements. This corresponds to having a negative enthalpy of formation H_f, defined as,

$$H_f = E_{\text{atom}} - \sum_{k=1}^{K} x_k E_{\text{atom},k}, \tag{4.8}$$

where $E_{\text{atom},k}$ is the energy per atom of the stable elementary phase of specie k. Thus, H_f quantifies the energetic gain of forming a mixed phase with respect to the elemental ones. The picture of stability is not as straightforward when considering the full range of compositional space, since binary and ternary compounds can decompose into two or three other phases of different compositions. This is the case if the appropriate linear combination of the energies of these phases is lower than the energies of the decomposing compound. The T=0 K phase diagram corresponds to the convex hull of a plot of the formation enthalpy against composition. The former is made up of the line segments of the smallest polygon that contains all the points in composition/enthalpy space (with $H_f \leq 0$). The points on these lines are considered stable, as the convex hull defines the stability criterion at each composition. It is a set of simplices, so lines (binary phases) or planes (ternary phases), delimited by points on the diagram. The points of the plot correspond to different phases, have coordinates of $[x_1, .., x_{k-1}, H_f]$. One composition coordinate out of K is not needed, since it is implicit from the other coordinates, as they sum to 1.

The geometrical construction of a convex hull is now described for the example of a binary system A-B. It follows the methodology used for AFLOW-CHULL [235], presented in more detail in Section 4.4.2, which is based on the Qhull algorithm [236]. In this work, the Python wrapper PYHULL [237] was used to construct convex hulls.

Since the enthalpies of formation of the lowest energy phases of A and B are zero and an arbitrary A-B compound must have a negative formation enthalpy to be stable, one line of the binary convex hull joins the two elemental phases across the full composition range at H_f=0. All points with $H_f > 0$ are not considered for the analysis. Note that this single line will constitute the full hull, if there are no binary compounds with negative enthalpies of formation. If such binaries do exist, however, the following step will consist in finding the outermost points in the $H_f < 0$ region of the phase diagram. Only the lowest enthalpy points at each composition are considered for the construction. A first construction is made using the elemental points and the lowest enthalpy point. If points exist below these lines (lower H_f at a given composition), for each facet, the points furthest away will be identified. They could be the same point for several facets. These points are added to the hull construction and new lines are formed between them and the end points of the facets they are closest to. All the new lines, along with the ones that had no outside points, form the new construction. Since again, points can lie outside, this process is performed iteratively until an actual convex hull is formed. The steps described in one iteration are illustrated in Figure 4.1. The result is a set of lines that "wrap" all the points, which are either inside or on the edge of the convex hull. Thus, the points of the convex hull (and their ordering) are the defining features of the convex hull. These points are associated with the stable phases of the binary system, and they do not decompose into other phases.

In the case of ternary convex hulls, three-dimensional plots are required, with the vertical axis typically used for the H_f coordinate and the horizontal plane for the compositional space. The latter is typically represented in the form of an equilateral triangle, with pure elemental phases at the vertices, binary phases along the edges and ternary phases in the centre. A schematic ternary diagram is show in Figure 4.2 to demonstrate how the composition is typically read from them. The associated three binary convex hulls are constructed first, to identify which binary points are used to build the ternary phase diagram. These points are then retained, along with the minimum enthalpy points at each ternary composition. The construction of the ternary convex hull is identical to that of the binaries, but the facets of the hull are no longer lines bound by points, but planes delimited by three points. An example of ternary convex hull is shown in Figure 4.2.

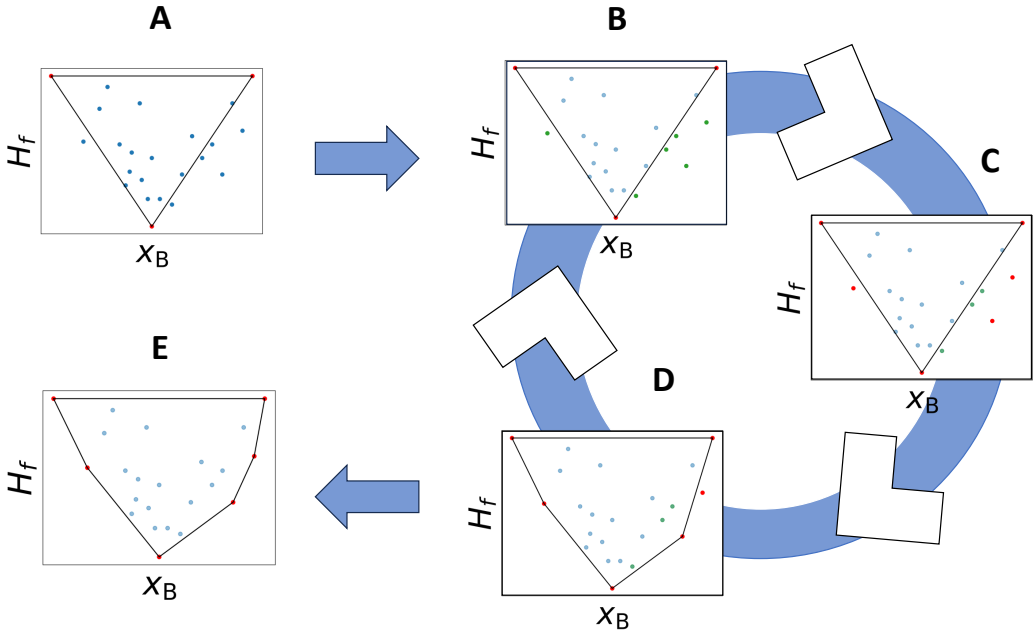


Figure 4.1: Graphical illustration of the procedure followed to build a fictional A-B binary convex hull. H_f is plotted against the proportion of B content, x_B. Black lines show the simplices of the hull, the blue points represent different binary phases. In step A, the minimum elemental phases and the global minimum are added to the convex hull, are highlighted in red and are joined. Steps B-D show the procedure of creating new facets, (B) by first identifying the points outside the construction (in green), (C) finding the furthest points from each facet (new red points), (D) and joining these with the other hull points. This procedure is repeated until reaching the final convex hull (E).

The convex hull now determines the stability criterion for materials. For a material of composition x_k to be stable, it needs to have an enthalpy of formation below the convex hull. In other words, on the phase diagram, a point placed with a certain enthalpy and composition must lie below (lower H_f) the facet of the convex hull of that composition. If it does, it will redefine the convex hull and be a stable phase, like the other points that remain on it. If it is not, it will lie at a distance δ above the hull. This is the difference H_f of the compound and the convex hull at x_k (ignoring the "top" facet delimited by the elemental points),

$$\delta = H_f - H_{\text{hull}}, \tag{4.9}$$

where H_{hull} is the enthalpy at the hull. The convex hull can be used to forecast the decomposition reaction that determines which phases the compound would separate into. Indeed, such a compound would decompose stoichiometrically into the compounds

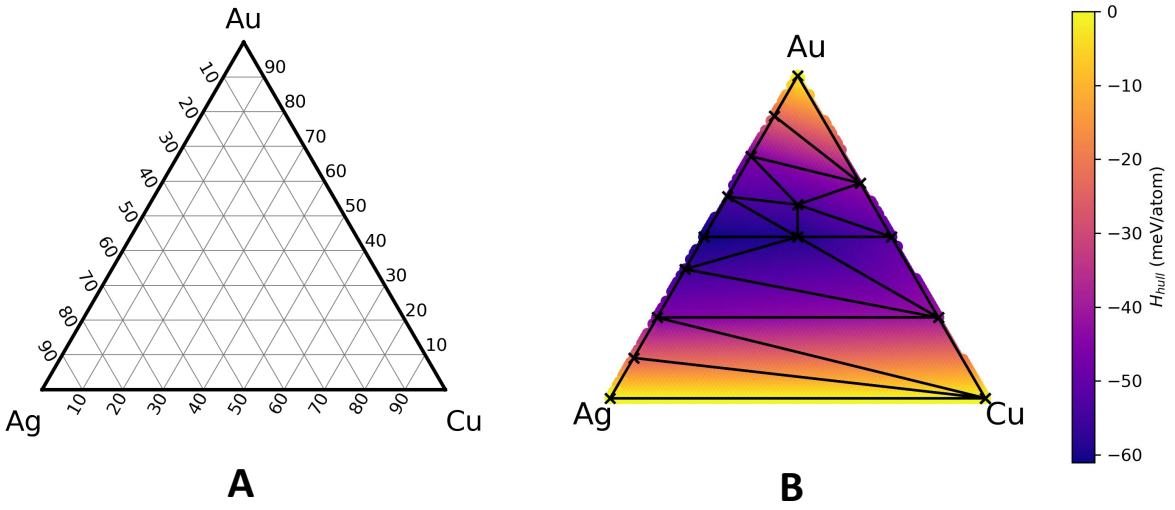


Figure 4.2: Example of a ternary convex hull for the Ag-Au-Cu system. (A) shows how an equilateral triangle is used to define the composition space of ternary compounds. (B) shows an example of a 3D ternary convex hull viewed from the top, with black crosses indicating the position of hull points and black lines delimiting the facets. The colour map shows how the depth of the hull, denoted H_{hull}, varies across the composition range. The data taken is taken from this work, presented in the next chapter.

that delimit the facet or facet edge closest on the phase diagram.

4.4 AFLOWlib Repository and AFLOW Standards

The Automatic FLOW for Materials Discovery library (AFLOWlib) is a large online repository containing computational properties of over 3.5 million compounds [238] and serves as the reference database for much of the work presented here. At its core, AFLOW is a software designed for the computational high-throughput search of novel materials. It thus provides a consistent and robust workflow to generate structures and handle the DFT calculations automatically. The online repository collects the results of all the calculations, which are performed across a wide range of elements (H to Bi excluding Ar) and for combinations of 2 to 7 different elements across different stoichiometries.

4.4.1 High-Throughput Framework

The AFLOW package, therefore, enables the running of high-throughput calculations in a standardised format. The starting point is the structure generation. Experimental compounds present on the Pauling File [239], ICSD [23] and the Navy Lattice Crystal

database are used both directly as candidates, but also to create structure prototypes, that could be decorated to create virtual compounds. Initially, there were approximately 400 such prototypes. Most compounds are binary and ternary derivative structures from *fcc*, *bcc* and *hcp* structures. For the *fcc* and *bcc* derived structures, supercells up to 20 atoms are used, and up to 24 atoms for the *hcp* structures. These make up the bulk of the compounds on the repository, since many derivative structures are created for different stoichiometries (approximately 20 for binaries and 10 for ternaries) of each binary and ternary present in the database. Based on this large library of structures, the AFLOW software follows a standardised procedure to perform HT calculations. It has been performed across the whole library and the results are available on the online repository. Note that it can be run locally as well for individual structures or for a user-specified set of structures. The compounds selected in any HT search are first relaxed, followed by a static and band structure calculations using a DFT code (VASP). Crucially, errors that occur in such runs are handled automatically. As the introductory paper outlines [238], these are typically related to the limited hardware resources or errors that arise due to problematic input parameters. Another central point of the framework is the use of robust, general and consistent input parameters employed to run electronic structure calculations. This notably led to the development of standardised Brillouin Zone integration paths [240].

All calculations on AFLOWlib are carried out with VASP [30, 49, 50, 51], with PAW pseudopotentials and GGA exchange-correlation functionals, as parameterised by Perdew-Burke-Erzenhof (PBE). Structures are fully relaxed (lattice parameters and atomic positions) twice to reach their equilibrium (close to null forces acting all atoms), using an energy convergence criterion of 1 meV/atom. The plane-wave kinetic energy cut-off and the k-mesh density chosen for these calculations depends on the systems studied. VASP provides recommendations for suitable species-dependent cut-off values that provide accurate calculations within a defined range. AFLOW uses a default of 1.4 times the maximum recommended value of all species present. For the k-mesh density, it is determined with respect to a central quantity, the number of k-points per reciprocal atoms N_{KPPRA}. The number N_i of k-points along each reciprocal lattice vector $\vec{b}_i$ is determined from N_{KPPRA} and N, the number of atoms in the unit cell, from the inequality

$$N_{\mathrm{KPPRA}} \leq \left[\prod_{i=1}^{3} N_i\right] \times N. \tag{4.10}$$

Note that N_i also varies with the norm of the associated reciprocal lattice vector $\left\|\vec{b}_i\right\|$, since there are fewer sampling points along the direction of reciprocal lattice vectors with lower norms. Therefore, under the constraint of Equation (4.10), the sampling density $\left\|\vec{b}_i\right\| / N_i$, is kept as uniform as possible across each direction. This determines the number of k-points along each direction. From these three numbers, the k-points are generated following the Monkhorst-Pack scheme [241], apart from compounds with a lattice possessing hexagonal symmetry, for which a uniform Γ-centred mesh is used. For all binary and ternary compounds, the N_{KPPRA} is fixed at 6,000 for geometric and ionic relaxations and to 10,000 for a more dense mesh for static calculations [19]. Between the relaxation and static calculations, the structure is changed into standard form for consistent categorisation of structures [240]. Occupancies of states at the Fermi level are determined using the Methfessel-Paxton approach [242], with a smearing width of 0.10 eV, for all relaxation calculations. For the static calculations, the occupancies are determined using the tetrahedron method with Blöchl corrections [243]. Finally, all relaxation calculations are initialised with non-zero magnetic moments for all species. If these moments are below $0.025\mu_b$ for certain atoms, the moments are set to zero for further calculations. This concludes the standards used for the AFLOW HT framework. For calculations carried out in this work, many of the standards presented above are followed. The differences will be noted for all relevant sections.

The AFLOW ecosystem also includes many other tools that are of great practical use in HT studies. The ones that are the most relevant for this project are briefly presented.

4.4.2 AFLOW Tools

To access the data available on the repository, a useful Application Program Interface (API) [244, 245], that follows the REST [246] principles, has been developed. The data is stored in a SQL database, which is organised in three main layers, ordered hierarchically. These can individually be queried via the API. Each layer has specific keywords and entries that can be queried. The top layer is the *Project-layer*, for which

the entries correspond to different projects. The three main different projects group unary, binary and ternary system data. The second layer is the *Sets-layer*, where individual entries correspond to different systems, e.g., Ag-Au-Cu. Several materials, with different structures and compositions, but with the same atomic components, are associated with the *Sets-layer*. Finally, the entries of the bottom *Calculation-layer* are associated with a single compound. These have many more keywords than those of other layers, since structural, thermodynamic, mechanical and electronic properties of a given compound can be retrieved. The full list of keywords for each layer can be found in reference [244]. This simple tool is key in enabling the availability of the wealth of data available on AFLOWlib, which would otherwise be accessed via a web browser. The API provides a systematic and reproducible mechanism for rapidly downloading material databases locally.

A second tool directly relevant to this work is the AFLOW-CHULL convex hull construction program, built into the AFLOW software, for which the methodology followed has been outlined in section 4.3. Note that for this study, an in-house code was made for this purpose, to have more flexibility for loading different databases and plotting. The AFLOW convex hull program also takes a long time to run, since it downloads all the relevant data from the repository via the API and performs a structural comparison on all compounds before the convex hull construction.

Another very useful feature of the AFLOW ecosystem is the structure prototype library [247, 248, 249]. By categorising the structures retrieved from experimental material databases (notably the Navy Lattice Crystal database) according to Space Groups, a library of structure prototypes was created. At the time of writing, there are 1100 prototypes present in the database. Each one is represented by the crystal structure of a real material. When creating a virtual material from a given prototype, the atomic species of the prototype are exchanged for those of the desired material. Each prototype is therefore characterised by its lattice parameters and positions of constituent atoms. A space group possesses different classes of symmetrically equivalent sites. These atomic positions are referred to as Wyckoff positions [250, 251, 252] and, depending on their class, they have specific degrees of freedom, expressed as fractional coordinates in each direction. The prototypes present in AFLOW conserve the Wyckoff positions. When a compound is created, in order to conserve the space group of the

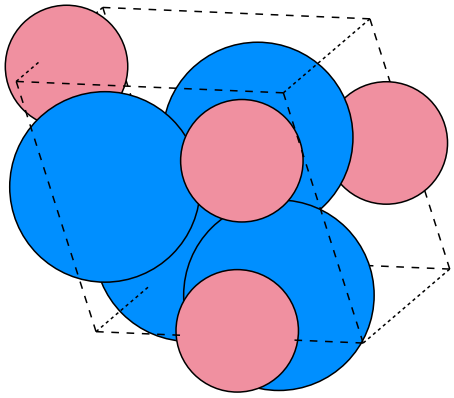


Figure 4.3: Unit cell of CoU. Co atoms are pink and U atoms are blue.

underlying prototype, only certain degrees of freedom are allowed to be modified. This could be a single lattice parameter to scale the volume (for cubic systems). For the atomic positions, only the free parameters of the Wyckoff positions can be changed. To clarify these points, an example is given here. The structure of CoU is shown in Figure 4.3. It is the structure of a 50:50 binary compound with 8 atoms in the unit cell. The underlying lattice is cubic, so that only the volume of the cell can be changed, determined by the lattice parameter a. Here a=6.356 Å. For this cell, there are three sets of Wyckoff positions, for which the fractional positions are given in Table 4.1. In CoU, both Co and U occupy a sites (which have a multiplicity of 4). These sites have one degree of freedom, the fractional coordinate x, which determines all the positions. For Co, it is 0.294 while for U it is 0.0347.

Table 4.1: Table of Wyckoff positions for space group 199. Note that the origin can be either (0,0,0) or (1/2,1/2,1/2) [252].

Multiplicity	Wyckoff Letter	Coordinates
12	c	(x,y,z), (-x+1/2,-y,z+1/2), (-x,y+1/2,-z+1/2), (x+1/2,-y+1/2,-z), (z,x,y), (z+1/2,-x+1/2,-y), (z+1/2,-x+1/2,-y), (-z+1/2,-x,y+1/2), (-z,x+1/2,-y+1/2), (y,z,x), (-y,z+1/2,-x+1/2), (y+1/2,-z+1/2,-x), (-y+1/2,-z,x+1/2)
6	b	(x,0,1/4),(-x+1/2,0,3/4),(1/4,x,0), (3/4,-x+1/2,0),(0,1/4,x),(0,3/4,-x+1/2)
4	a	(x,x,x),(-x+1/2,-x,x+1/2), (-x,x+1/2,-x+1/2),(x+1/2,-x+1/2,-x)

The final feature in AFLOW that is used in this work is the XtalFinder [253] tool that is used to compare different compounds' structures and establish whether they are the same or not. It can also be used to determine the space group of the structure and Wyckoff positions occupied by the atoms. To achieve this, each structure is first mapped onto its ideal prototype using the AFLOW-SYM [254] tool. This determines the symmetry operations that a structure possesses, hence it finds the space group of the structure, along with the occupied Wyckoff positions and allowed degree of freedom values (lattice parameters and fractional coordinates). Note that, since this is performed numerically, tolerance values are required with respect to lattice vectors and atomic positions. The default tolerance is proportional to $d_{\min}/100$, where $d_{\min}$ is the minimum nearest neighbour distance in the compound. Such information is used to define the prototype of the structure consistent with the International Tables for Crystallography (ITC) [251]. From this representation, different structures can thus be compared systematically. Three types of comparisons are then carried out: one based on symmetry, one on the local atomic environment and a general crystal geometry. The first one determines if the structures have commensurate space groups and Wyckoff positions. The second is based on comparing the atomic neighbourhood of identical atoms in each structure. Each neighbourhood is described by the least frequently occurring atom (LFA) atomic geometry (AG), a set of vectors that connect a central vector to neighbouring atoms of LFA type. Comparison is then only performed between structures with the same atoms in their neighbourhoods. Note that this stage is useful, since it is less sensitive to tolerance values than symmetry comparisons of structures, given that the atomic arrangements are sparse. Finally, a full isoconfigurational test is performed by attempting to map one structure (test) onto another (reference), by scaling the volume, creating supercells and choosing an appropriate origin for the test structure. The aim is to check whether the lattices match and the atoms are at the same positions, all within a certain tolerance value for volumes, lattice parameters and atomic distances. These steps are followed not only to determine if two materials are identical (Material-type comparison), but also to compare the underlying structure of two different materials made of different species (Structure-type) and determine whether they have the same backbone structure (ignoring atomic types). The latter is useful for finding unique prototype structures.

Such comparisons are typically performed on a database of M structures, where there are $\frac{M\times(M-1)}{2}$ pairs to compare. To accelerate the process, structures are grouped together based on the symmetry comparison and stoichiometry to decrease the number of comparisons required. The groups are refined during the near isoconfigurational comparison step. Within each group, a full geometric comparison is then performed, in which all grouped structures are compared to a reference structure. Compounds that do not match the reference are moved to a separate group. Within this group, the comparison loop is repeated. The process is therefore iterative. The result is a set of groups of identical structures, each represented by a single compound. This tool is useful for eliminating identical structures, that create redundancies in databases for training machine-learning models and creating prototypes.

This concludes Sections 4.2, 4.3 and 4.4, in which an overview of the relevant theory and methods for the *ab-initio*-based determination of material stability was given. The thermodynamics of compound formation was summarised and linked to established ways to calculate the terms of Equation (4.7). In this work, it will be narrowed down to the determination of the enthalpy of formation H_f of a compound and the construction of convex hulls, the phase diagrams at T=0 K. Finally the AFLOW ecosystem was presented, as it constitutes the reference repository for this work, notably for its rich database of computationally determined convex hulls, and provides a number of tools that are crucial in automating the study. In the following section, a presentation of the ternary Ag-Au-Cu system is given, which is the test system for this work.

4.5 Ag-Au-Cu System

The aim of this first body of work is to create an efficient screening tool for the prediction of the total energy and enthalpy of formation of ternary alloys. In order to design such a tool, a test system is necessary to construct the model, tune it and assess its effectiveness. The system selected here is Ag-Au-Cu, a ternary alloy system made up of the three coinage metals. Alloys made of all three elements, rich in Au-Cu, in fact have applications in the dental industry, where they are used in single units, implants and retainers [255]. All three elements are also used in combination in wire bonding, to link copper wires with metallic electronic components [256, 257]. Along with their

high resistance to corrosion, they are also able to resist high forces, notably when the alloys are age-hardened [255].

This choice of system is motivated by the idea of using a simple system for testing, one that does not display specific bonding patterns and complex coordination, but rather quite simple chemistry, since these elements form compounds with the common *fcc* close-packed underlying structure. All three elemental phases are *fcc* (A1) and their lattice parameters only differs by 11% [258] (Ag 4.086 Å, Au 4.072 Å and Cu 3.613 Å at 293K [259]). They are also group XI elements and all have filled *d* shells. In this section, the common phases of this system and the phase diagrams of the three binary systems (Ag-Au, Ag-Cu and Au-Cu) are described. The section is based on the CALPHAD construction of the phase diagrams carried out in references [260, 261], which make use of experimental work previously published.

Before beginning, some terms, relevant to solid-state phases and phase diagrams, are defined. A solid-state solution is a compound with a crystalline structure whose sites are randomly occupied by the constituent elements. The composition is fixed, however, over the full scale of the crystal. In contrast, for an intermetallic compound, each site has a single element associated with it. Finally, a eutectic point on a composition-temperature phase diagram, corresponds to a point at which the melting point of the mixture of components is lower than that of all of the individual components.

4.5.1 Ag-Au Binary

The Ag-Au binary system is miscible across the entire compositional range, and *fcc* solid solutions (A1) are formed at all compositions and temperatures up until melting. This can be seen in the phase diagram in Figure 4.4a. The melting temperature of Ag is 1235 K and that of Au is 1337 K. There is a continuous liquidus-solidus line between the solid solution and the liquid phase, which is narrow (2 K wide) [262, 263]. The formation of binary alloys is driven by mixing of the two elements (enthalpy and entropy [264]) and there is little effective interaction between them [265]. In the construction of the convex hull, this leads to having degeneracies for many similar structures. It is suggested that at low temperatures (below 200K), stable intermetallics may exist, but that the low rate of diffusion at such temperatures may have prevented the experimental observation of them [266]. *Ab-initio* studies agree that phases with

composition Ag_3Au, AgAu and $AgAu_3$ lie on the convex hull [265, 266, 267, 268], a fact that agrees with the convex hull of AFLOWlib [235], although in all these studies, the compounds are intermetallic.

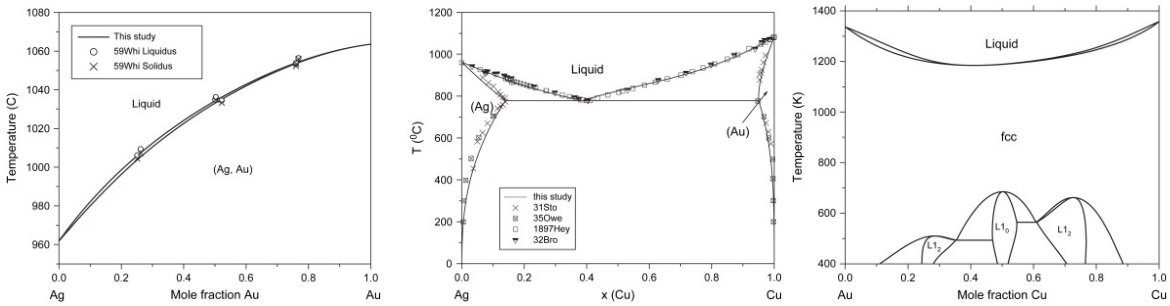
4.5.2 Ag-Cu Binary

The Ag-Cu binary system is mostly immiscible across the majority of the compositional range in the solid state (below 1052 K), as shown in Figure 4.4b. At temperatures beyond 400 K, there is some solubility in each of the elemental phases, which goes up to 14.1% of Cu in Ag and 4.9% of Ag in Cu. A miscibility gap therefore dominates the phase diagram. A eutectic point exists at 1052 K. Note that the melting temperature of Cu is 1358 K. [269]. The experimental and calculated enthalpies of mixing are positive across the entire compositional range [270, 266]. As a result, only phase separated alloys (with regions made only of Ag and others of Cu) exist. No low temperature stable phases have been reported [266]. The DFT-calculated convex hulls are made of a single tie line, joining the two elemental phases, as is the case on AFLOWlib.

4.5.3 Au-Cu Binary

The Au-Cu binary system forms three ordered phases (intermetallics). The CALPHAD computed phase diagram is shown in Figure 4.4c. These phases are the Au_3Cu, equiatomic AuCu and the $AuCu_3$ phases, which respectively form in the $L1_2$, $L1_0$, and $L1_2$ structures, respectively. These correspond to different decorations of an *fcc* lattice. At higher temperatures (dependent upon the composition), a *fcc* solid-state solution, like that in Ag-Au is formed. The liquidus-solidus line is narrow and presents a minimum close to composition with 40% Ag. This system has been very well studied and serves as a test system for binary alloys investigations [271, 266, 272, 268, 273]. *Ab-initio* calculations have notably predicted that another phase is stable at low temperatures, Au_2Cu with the β_2 structure of $MoSi_2$ [266, 272]. Note, however, that this was predicted with LDA and PBE functionals and a later study performed with the hybrid HSE functional, including non-local interactions, found to better match experimental measurements of Au-Cu phases, showed that the β_2 phase is in fact not stable [268]. Note that for the Au_3Cu structure, LDA and PBE calculations have notably

suggested that the stable phase is a $L1_0$ superlattice [266, 274]. This has not been investigated using non-local hybrid functionals. The plethora of studies on the Au-rich compositions shows that it is unclear which phases are stable at low temperatures, and that many *fcc* superlattices lie close in energy [266, 272, 268, 273, 274, 275, 276]. On AFLOWlib, the AuCu and $AuCu_3$ phases are present on the convex hull, but Au_3Cu is undercut by Au_2Cu.



(a) Phase diagram of Ag-Au. (b) Phase diagram of Ag-Cu. (c) Phase diagram of Au-Cu.

Figure 4.4: Binary phase diagrams of Ag-Au (left), Ag-Cu (centre) and Au-Cu (right). The images are taken from the CALPHAD calculated phase diagram from reference [261]. All solid lines are the result of the CALPHAD computed phase diagram of that study. For Ag-Au, dots and crosses correspond to measurements from reference [263]. For Ag-Cu, the experimental references of the legend are as follows: cross [277], crossed-squares [278], squares [279] and triangles [280].

4.5.4 Ag-Au-Cu Ternary

The ternary phase diagram of the Ag-Au-Cu is strongly influenced by the associated binary phase diagrams. It is shown in Figure 4.5. As described above, the Ag-Au and Ag-Cu systems present stable phases, whereas the Ag-Cu system does not. Therefore, Ag-Au alloyed with small concentrations of Cu as well as Au-Cu with small concentrations of Ag can form. At low Au concentrations, due to the miscibility gap of Ag-Cu, no single phase made of all three binaries exist, as compounds form with separate mixes of pure Ag phases with Cu, AuCu or $AuCu_3$. Only at high Au concentrations can an A1 solid-state solution of all three elements be formed [281]. At higher temperatures, the miscibility gap becomes less influential and the ternary solid-state solution is stabilised over a larger compositional range. There are no known thermodynamic *ab-initio* studies of this ternary system at the time of writing (aside from those in HT repositories), showing how understudied ternary systems are compared to the binary

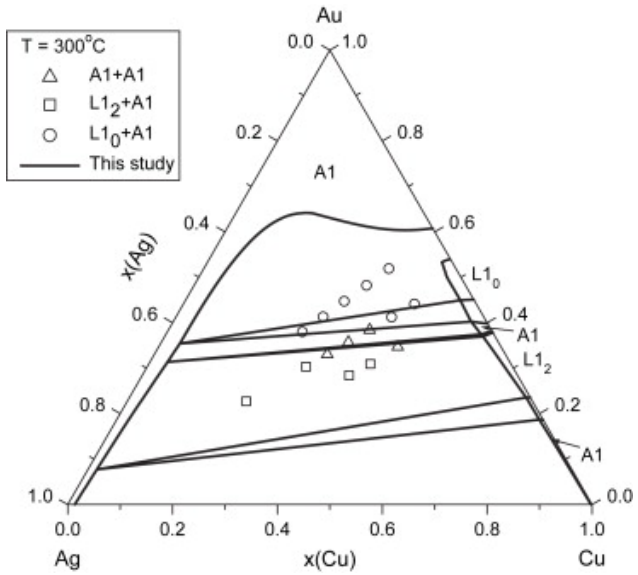


Figure 4.5: Isothermal section of the Ag-Au-Cu phase diagram at 573K. The image is taken from the CALPHAD computed study of Reference [261]. Triangle and squares correspond to measurements taken from [282] and circles from [283].

ones. In AFLOWlib there are no ternary intermetallics that lie on the convex hull, with the Au-rich phase, $AgAu_2Cu$, 3 meV/atom away from the hull, being the closest phase.

Having introduced the main aspects of the binary and ternary phase diagrams for Ag-Au-Cu, the rest of this chapter focuses on the work aimed at building a screening tool, surrogate to DFT, in order to efficiently and accurately predict the total energy and hence the enthalpy of formation of ternary compounds created virtually. Ag-Au-Cu will serve as the test system for the development of this model.

4.6 Fitting SNAP to Binaries

There are two central ideas of this work. One is to be as efficient as possible in building a screening tool for constructing DFT ternary alloy phase diagrams. As explained in the introduction of this chapter, this means using readily available convex hull data from AFLOWlib. This choice presents two main advantages. Firstly, such energy data are already computed, so there is no the need to reperform the DFT calculations. Secondly, no calculations are "wasted" in the sole effort of training a MLIAP. By recycling AFLOWlib data, that are by construction relevant for building a convex hull, the energies of $\sim 10^2$ different compounds are accessible for training. All DFT calculations

are thus of interest in a convex hull construction and the training of the MLIAP can be seamlessly integrated into this process. Such a screening tool could subsequently help accelerate the search for other low-lying energy compounds, by rapidly selecting the most promising ones from a large pool ($\geq 10^4$) of candidates.

The second central idea for this work is to use data from binary phases only to train the model. As explained, this stems from the fact that on online repositories, like AFLOWlib, there are far more data points available for binary systems than for their ternary counterparts. Moreover, there are three binary systems associated with each ternary one. Ternary intermetallics play an important role in the technology sector [284], providing motivation for further search in this space, which has thus far remained rather unexplored. On the contrary, binary alloy phase diagrams have already been thoroughly investigated. This implies that it is less likely to discover new phases there, and that data on existing phases are abundant.

The SNAP MLIAP is used to develop a screening tool. Such a model is capable of predicting the energy (as well as forces and stress tensor) of a compound, based solely on the position of its atoms, including periodic boundary conditions in the case of alloys. Note that the models presented are trained solely on energies. The principal reason for this is that these are the only physical quantities relevant for training available in AFLOWlib (forces and stress tensors not being present). This great restriction reduces the number of data points provided by each structure, since a compound with N atoms provides $3N$ force data points and 6 stress tensor elements, compared to only one energy value. As it will be illustrated, even if forces were included, these would not influence the results much, since structures in this study are close to equilibrium, so the forces almost vanish.

Being a machine-learning model, SNAP requires training data to fit its parameters $\vec{\theta}$. A cross-validation procedure is also required to tune the relevant hyperparameters, namely the cut-off value of the bispectrum components, $J_{\max}$, the cut-off radius, R_{cut}, as well as the atomic weights that appear in the atomic density function in Equation (3.24). Various tests must also be carried out to assess the utility and limitations of the model. The aim is to train such a model on the data of binary compounds from the Ag-Au, Ag-Cu and Au-Cu, in order to make predictions on ternary compounds. At first, models are trained and tested on the individual binary systems.

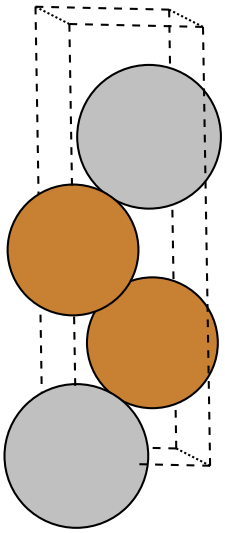
Table 4.2: Data available on the unary, binary and ternary systems associated with Ag-Au-Cu.

Properties	Ag	Au	Cu	Ag-Au	Ag-Cu	Ag-Au	Ag-Au-Cu
# AFLOWlib Compounds	26	39	40	188	275	281	88
# Unique Compounds	11	12	11	168	238	238	88
# Different Space Groups	8	10	9	38	44	43	21

In Table 4.2, data available from AFLOWlib on the compounds for the different unary, binary and ternary systems are given. There are redundancies in the database, which are identified and removed with the help of the AFLOW material-comparison tool [254]. The number of remaining structures, as well as the number of different space groups (SG) represented in each system, are also given in Table 4.2. There are many duplicates in the elemental systems, since these are present in the database of all three binary systems. There are 10 different prototypes used for these on top of the ICSD structures, which are the *fcc* (SG #225), *bcc* (SG #229) and *hcp* (SG #194) structures, along with the *fcc* diamond structure (SG #227), the α-As structure (also Se and Bi) (SG #166), the γ-Se structure (also Te) (SG #152), the β-Sn structure (SG # 141), the face-centered tetragonal structure of In (SG # 139), the α-Ga structure (SG # 64) and the α-O structure (SG # 12).

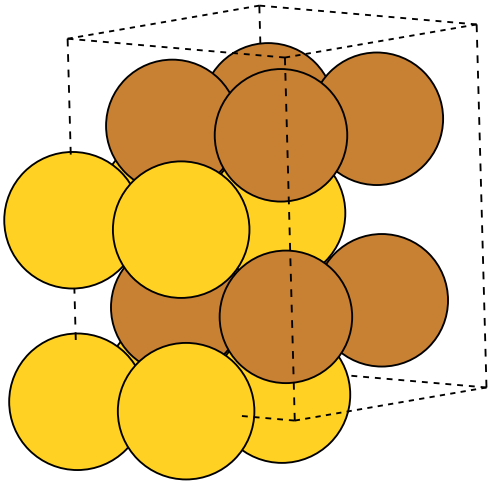
The binary structures are obtained from different types of prototypes. Some of these are based on the unary prototypes, with supercells constructed, if necessary, to accommodate the correct number of atoms, depending upon the stoichiometry. Additionally, certain structures are based on binary prototypes presented in AFLOWlib, such as the Fe_7W_6 structure [285] or the $MoPt_2$ one [286]. The last source of prototypes are *fcc*, *bcc* or *hcp* supercells, which can include certain distortions that break the symmetry or change the symmetry of the parent structure due to the supercell construction. One example of each case is illustrated in Figure 4.6. For the ternary structures, the original prototypes are similar to those of binaries, with the addition of some Heusler structures. Further details are given in Section 4.7.

Before presenting the results, a comparison of this database to those more commonly used for MLIAPs is presented. Two studies are chosen for this purpose, that illustrate two schemes for constructing a database: one based on physical intuition and one



(a) AFLOWlib compound AgCu 126[a].

[a]auid: aflow:c7838ef6fba6e6c3.



(b) AFLOWlib compound AuCu 60[a].

[a]auid: aflow:dc5e8311267c00ee.

Figure 4.6: Examples of binary compounds present in the Ag-Au-Cu AFLOWlib database. Panel (a) shows an Ag-Cu binary compound with space group # 164. Its underlying undecorated structure is *hcp*. Here the unit cell is a 1×1×2 supercell of the conventional *hcp* cell, but the atomic positions are decorated with Ag and Cu atoms. The space group number is hence # 164 rather than that of *hcp* (# 194). Panel (b) shows an Au-Cu binary compound with space group # 164. Note that a 2×2×2 of the conventional cell is shown for clarity. Here the structure has a body-centered orthorhombic cell, with Au and Cu atoms positioned on each of the two positions of the conventional unit cell. It is similar to the *bcc* structure, but with altered lattice constants a, b and c. The space group number is hence # 65 rather than that of *bcc* (# 229).

that uses an automatic scheme. The first is from the work of Chen *et al.* [127], in which a SNAP model was trained as a general purpose interatomic potential for Mo. This illustrates the types of databases employed for a standard use of MLIAPs, namely to run molecular dynamic simulations and predict material properties such as lattice constants, vacancy formation, surface and grain boundary energies, the phonon spectrum and even the melting temperature. The SNAP model produced was able to run accurate MD simulations for large system over long time-scales (∼2000 atoms for 500 ps). The training database was physically informed with $\sim 10^3$ training points (note that in this case only one element is present). These include ground-state phases from the Materials Project repository [20] (8 structures), surface slabs, vacancy, and grain boundary structures (respectively, 11, 24 and 13), along with large structures (54 atoms) that are elastically deformed (67 structures) and used for different sets of

AIMD simulations (169 structures). All of these data classes were necessary to expose SNAP to a variety of atomic configurations, notably with missing atoms (surfaces and vacancies) and unusual angular and radial distributions (grain boundaries). Such data diversity enables the force-field to be robust during the MD simulations. Note that here stress tensors and forces were used for training, enhancing the number of training data points significantly compared to this study. To be more precise, 284 energies, 15,600 force components and 1,704 stress tensor elements were used in their study. The role of the model in this work is not to achieve such type of accuracy, but rather to work as a screening tool for the energy of different structures across a phase diagram. If this work were to use the same training database as theirs, it would only include the Materials Project part of the database [20], namely less than 1% of this data set (considering forces and stress tensor elements).

The work of Gubaev *et al.* [175] provides another suitable example of a MLIAP, in this case, trained to construct phase diagrams. The MTP framework was chosen to accelerate the search for novel stable alloys, as exemplified for the Cu-Pd, Co-Nb-V and Al-Ni-Ti systems. Their force-field is both an efficient screening tool and possesses the ability to perform relaxation. The training database was constructed on-the-fly using an active learning scheme, centred around the extrapolation grade (presented here in Section 4.9). A set of structures (10^4-10^5) were created from *hcp*, *bcc* and *fcc* super-cells and were decorated for different compositions using the methodology presented in Reference [287]. The maximum number of atoms varied from 8-20 depending on the system. The MTPs were then trained on-the-fly as the structures were being relaxed. This led to having 523 and 383 training structures for Cu-Pd and Co-Nb-V respectively. For Al-Ni-Ti, two different MTPs were necessary, one with 2,393 and the other with 976 training structures. Based on details provided, this would put an upper bound of 3,369, 121,284 and 20,214 on the number of energy values, force components and stress tensor elements, respectively, to train the model. It is interesting to note that for this system, the AFLOW structures are also used in the training set, highlighting the additional diversity that such structures bring. The key difference with this work is that many of the structures used for the training of the MTPs are out-of-equilibrium, as they appeared in the relaxation path and were judged by the active-learning scheme as being the most relevant for training a robust force-field. In this case, all structures

included are at equilibrium, do not need to be generated and are directly relevant for the convex hull construction. Note that in both cases outlined above, forces and stress tensors are used for training, as opposed to this study, as such quantities are not given on AFLOWlib.

It should be reminded that the aim of the SNAPs model presented here is not to act as a robust force-field suitable for accurate molecular dynamics simulations across the phases of a ternary system, but rather to be used as a screening tool in the context of the high-throughput search for stable materials. In high-throughput studies that make use of a machine-learning energy screening tool instead of, or in combination with, DFT, values for the stability criteria, δ, go up to 100 meV/atom [162, 203, 288, 289]. As such, a model with an upper-bound error of 100 meV/atom is sought. Note that in AFLOWlib, the stability criterion is set to $\delta = 0$ meV/atom, as only compounds on the convex hull are deemed to be stable. This comes as a consequence of the choice of training database. In fact, the training structures are not generated using scientific intuition or an automatic scheme, but are taken directly from a readily available online repository, AFLOWlib in this case.

Finally, for the Ag-Au-Cu test system, since the initial aim is to explore the validity of using SNAP as a screening tool, total energies are not taken directly from AFLOWlib. The compound structures are, but the energies are recomputed with VASP [30]. The reason for this is that, while AFLOWlib provides an excellent reference for convex hulls, since the calculations providing material properties were carried out in a very high-throughput fashion (millions of calculations performed across the repository), some of the convergence criteria are slightly loose. This results in small inconsistencies within the database, which leads to small number of energy errors of up to 10^{-2} meV/atom. This motivated the use of tighter convergence for the DFT calculations, notably by using a plane wave cut-off energy of 600 eV for all compounds (compared to 516 eV for Ag-Au-Cu and 479, 479 and 324 eV used for Ag-Cu, Au-Cu and Ag-Au binary compounds respectively). To illustrate the impact this has on the results, the convex hull of Ag-Au based on the compounds from AFLOWlib and used in this study is shown in Figure 4.7. Convex hulls constructed from AFLOW data and recomputed data are shown. The hull is significantly altered, notably by the poor convergence of one point at the 50:50 composition. Even other points show deviations between the recomputed

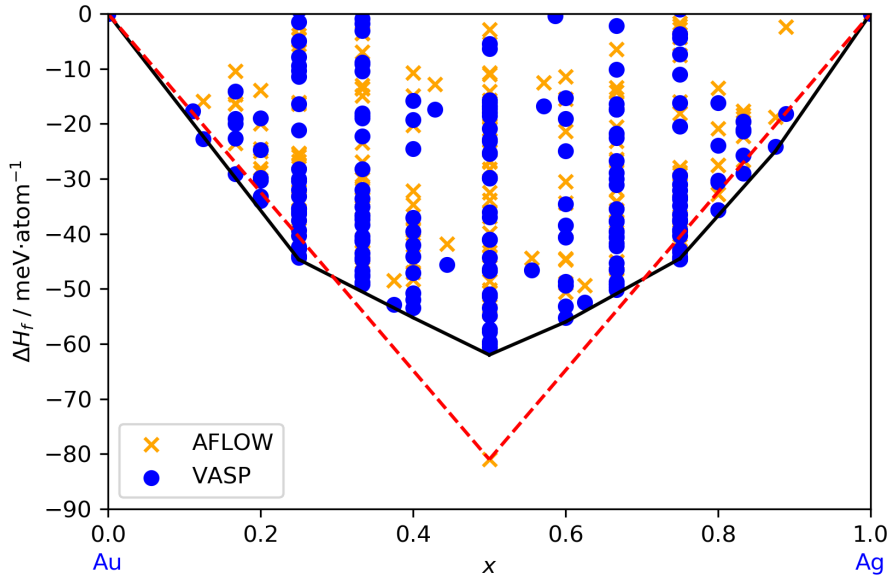


Figure 4.7: Ag-Au convex hull constructed from AFLOWlib (red dashed lines) and recomputed VASP data (black lines). The proportion of Ag, x, is given on the horizontal composition axis. Blue points are data computed with VASP and yellow crosses are from AFLOWlib data. Only the negative portion of the convex hull is shown here.

and AFLOWlib values. Note that this convex hull constitutes a worse case scenario, notably because of the low plane-wave energy cut-off value used for this binary system and the shallowness of the hull in question, and most AFLOWlib convex hulls show less deviation to the recomputed ones. Nevertheless, to avoid convergence issues affecting the outcome of this study and to enable the easier and consistent integration of new data points for training and testing, the higher fidelity DFT-recomputed data were used.

4.6.1 Fitting Binaries

The first test presented to SNAP was that provided by individual binary systems, along with their associated unaries. The aim is to establish SNAP's accuracy in fitting and predicting the energies of binary compounds. Unlike in the case of ternary systems, where the ternary compounds constitute the test set, for individual binaries there is no obvious choice of test set to robustly assess the quality of the predictions. As a result, only a cross-validation (CV) set is used, instead of a test set, to assess the ability of the model to generalise to unseen data. This implies that data used for hyperparam-

eter optimisation forms the set (see Section 4.8 for caveat). An important feature of the dataset studied is the presence of high-enthalpy structures, which have unusual structures. These are notably unaries and binaries that were made from prototypical structures that have geometrical arrangements (radial and angular distributions) that are very different to those typically found in alloys (close-packed structures). "Hollow" structures, with cages of atoms around a large portion of vacuum are notable examples of these, which have structures based on oxides such as ReO_3 [290, 291]. In order to avoid such structures strongly biasing the model error metrics, 10 sets of training / cross-validation are created. These are created based on an 80/20 random split of the full binary dataset. This leads to training 10 different SNAP models, trained on each one of the training sets, each with its own parameters $\vec{\theta}$. Thus, an ensemble of 10 SNAP models are used. This is, in fact, kept for the ternary system study as well. A full discussion of the methodology followed to determine the hyperparameters and the values used for all systems is given in Section 4.8. Predictions from this model in further sections (when predicting the bulk modulus and ternary energies), are performed by averaging the predictions of all the 10 models.

Table 4.3: Summary of the average errors over 10 SNAP models for the three binary systems.

Error (meV/atom)	Ag-Au	Ag-Cu	Au-Cu
Training MAE	2.63	2.82	4.30
Training RMSE	4.37	4.01	6.25
CV MAE	6.85	7.27	8.57
CV RMSE	13.83	15.15	15.73

The averaged results of the mean absolute errors (MAE) and of the root-mean-square errors (RMSE) are given in Table 4.3 for the three different binary systems. The errors are consistent in order of magnitude with typical MLIAPs trained on similar dataset sizes [127]. Some errors may be slightly higher than values reported for current state-of-the-art MLIAPs [292, 293, 294], however the dataset here is smaller and not specifically curated. All binary systems are fitted to very similar accuracies. The cross-validation RMSEs for all systems are higher than the training RMSEs by more than a factor of two. This highlights the fact that high-error compounds, which correlates

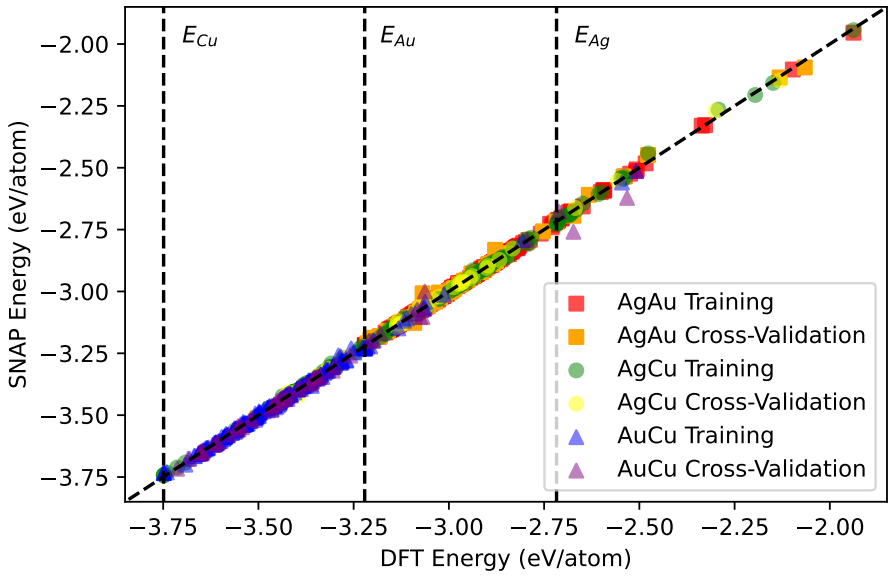


Figure 4.8: Parity plot of one set of training and cross-validation sets of each of the three binary systems. There is one SNAP model of the ensembles associated with each binary system. The dashed parity line is a guide to the eye. The vertical dashed lines give the values of energy of the lowest energy unary compounds, as marked by E_{Ag}, E_{Au} and E_{Cu}.

with high-enthalpy ones, are more difficult to fit for this binary system. These are typically outliers, and they highly influence the RMSE. A typical parity plot for one of the ensemble SNAP models of each binary is shown for illustrative purposes in Figure 4.8 for the training and CV sets. In conclusion, the ensemble of SNAP models is capable of accurately predicting the energies of binary compounds. Before testing the predictions on ternary systems, tests on the prediction of bulk moduli are performed.

4.6.2 Ag-Au Bulk Moduli

As an extra study, the SNAP models were used to predict the bulk modulus of certain binary compounds, selected from the database. This can be obtained by fitting the Birch-Murnaghan equation of state [48] [Equation (2.29)] to the energies obtained following the compression and expansion of the cell of a compound. The test is performed with the ASE package [295]. Here, this corresponds to isotropic compression/expansions, such that only the volume of the cell is changed. The trend of the energy against volume (EV) curve is close to parabolic. The bulk modulus B_0 is obtained from the curvature of the plot. It is of interest to check whether SNAP is capable

of reproducing qualitatively the trend and quantitatively the value of the bulk moduli, and correctly predicting the minimum volume V_0 and energy E_0 of the cell.

In order to test this aspect, a selection of binary compounds are taken from the set and their cells are compressed and expanded for changes in volume between -10% and 10% in 15 steps. The energies of these compounds are computed with DFT. SNAP is then used to replicate the *ab-initio* energy/volume curves obtained. Note that this test is biased, since the systems considered are present in the training/cross-validation of the ensemble of SNAP models trained. However, the compounds chosen are low-enthalpy ones and, hence, possess chemical environments resembling those of other compounds. It is expected that the exclusion of these points in the training of the ensemble would not change the results presented. Results are given for 7 binaries of the Ag-Au system. As a reminder, the ensemble predicts energies by averaging the predictions of all 10 models.

Table 4.4: Percentage differences of V_0, E_0 and B_0 as fitted with the Birch-Murnaghan equation between DFT and SNAP ensemble models trained on different datasets.

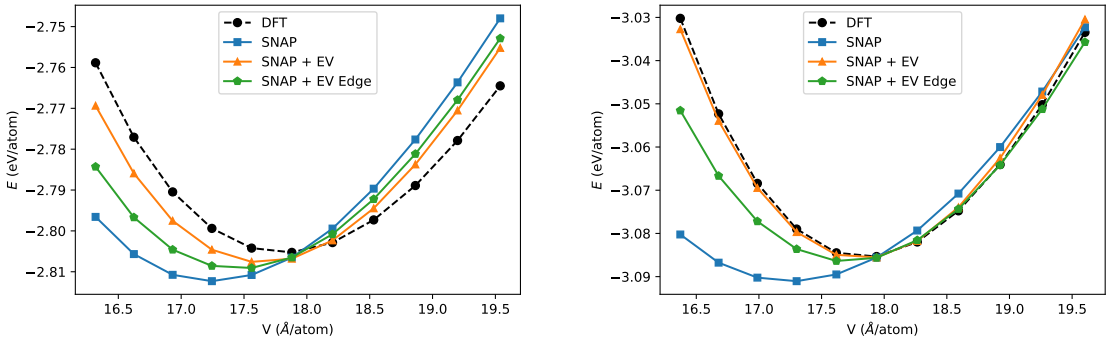
Model	V_0	E_0	B_0
SNAP	3.6	0.2	33.3
SNAP + EV	0.6	0.1	5.3
SNAP + EV Edge	1.2	0.1	14.6

The average percentage differences between the DFT and SNAP predictions for V_0, E_0 and B_0 are given in Table 4.4. There are three different training/cross-validation datasets used to construct the SNAP ensemble models. The first (referred to as SNAP) is the standard Ag-Au binary dataset. The second dataset (SNAP + EV) is enhanced relative to the first, with data points from the energy against volume curve of *fcc* Ag and *fcc* Au (15 from each, sampled uniformly between -10% and 10% volume change). The final one (SNAP + EV Edge) is similar but only includes the points on the "edge" of the elemental EV curves, i.e., only the ones with the most compression and expansion. This corresponds to only four extra points being added to the dataset. Note that in the cases where compressed/expanded compounds are present, such compounds are forced into the training set and are therefore never present in the cross-validation step. Two illustrative EV curves predicted by these models along with the DFT reference

are shown in Figure 4.9.

Overall, all models show predictions consistent qualitatively with those of DFT. Quantitatively, E_0 is on the right scale with percentage errors $\leq 1\%$ and is the quantity best predicted by SNAP. This is expected since SNAP is trained to specifically predict energies and thus, while the exact position (V_0) of the minimum compound is predicted incorrectly, the energy scale will still be accurately described. Since the position of the minimum is better predicted with the SNAP models that make use of the unary EV points, predictions for E_0 are better for both of these. The predictions of V_0 are generally well reproduced by the two EV-enhanced models. Note that the estimation of the minimum point is sometimes very accurate ($\leq 1\%$), but this is not consistent across all systems, with differences of up to 5.57% reached for the SNAP model for one system. The SNAP model generally tends to underestimate the volume. The bulk modulus, dependent upon on curvature and sensitive to small changes, is the least well predicted of all properties. For the SNAP model, the maximum deviations reach 72.19% for one system, while it reaches 9.17% and 31.88% for the SNAP EV and SNAP EV+Edge models. This quantity is the one most influenced by the presence of elemental compressions and expansions in the dataset. Indeed, their presence reduces the error significantly. All models predict the energy of the true minimum energy compound very well, since it is information present in the ensemble training.

This shows that adding only two data points per element from the unary EV curves enhances predictions of equilibrium volumes and bulk moduli. The inclusion of such points could be easily incorporated into the database, if elastic property screening is required. Note that here the maximum difference in bulk modulus between the binaries studied is only of 37.31 %. One last point to note is that here the SNAP predictions (trained on binaries only), captures the parabolic behaviour of the curves very well, notably at low volumes. For many choices of hyperparameters (notably for the atomic weights), this behaviour is poorly reproduced by SNAP, typically for compressions. When atoms are too close, the SNAP predictions are poor and the energy typically decreases too much. Careful hyperparameter optimisation, as described in Section 4.8, prevents this behaviour.



(a) (b)

Figure 4.9: Energy against volume curves for two Ag-Au binary compounds as predicted by DFT (black dots joined with black lines) and different SNAP models (coloured markers). The lines are guides to the eye.

4.7 From Binaries to Ternaries

The study solely on binaries showed that SNAP could be used to fit the energies of binary compounds in the training set and predict the energies of unseen compounds, in the cross-validation set, to an accuracy of the order of 10^{-2} eV/atom. The aim of this work is centred around using such a model to predict the energies of ternary compounds. It is not evident at first that such an approach could work. This is due to the lack of certain interactions in binaries compared to ternaries. While 1-body and all the different 2-body interactions (Ag-Au pair, Au-Cu pair and Ag-Cu pair) are present in binary systems, the 3-body interactions that involve the three different species (Ag-Au-Cu) are of course never present for binaries, by definition. The hypothesis of this work is that such interactions can be sufficiently well-described by SNAP from its interpolation from other N-body interactions present in the training set. Note that N-body interactions up to 4-body are included in SNAP. Validation of this hypothesis would imply that a SNAP model trained on a fixed database of binaries could be used as a screening tool for ternary compounds.

In order to test the working hypothesis, the energies of several sets of ternary compounds were predicted. The first set was composed of the ternaries present in the AFLOWlib's convex hull database. These structures are similar in nature to the binary ones presented in Section 4.6, since the majority of structures are decorations of *bcc* and *fcc* structures / superstructures, with various distortions added to break symmetry. In addition, a small number of less-common structures based on certain

prototypes are added, including Heusler alloys [296] (note that these only comprise atomic neighbourhoods that are the same as those found in *bcc* compounds). These compounds are therefore structurally similar to the binaries. In Section 4.7.1, their energies are predicted with SNAP. The bulk moduli of a subset of these compounds are also predicted in Section 4.7.2. The next set of compounds tested is made up of new prototypes, not present in AFLOWlib's convex hull database. These are constructed based on AFLOWlib's prototype library, but are from a range of different original compounds, notably non-alloys. Their energies of these are tested pre- and post-DFT relaxation. The results are presented in Section 4.7.3. Finally, in Section 4.7.4, SNAP is tested on two different ternary systems (Mo-Pt-Ti and Cd-Hf-Rh).

4.7.1 Ternaries on the Convex Hull

The SNAP ensemble model is now trained on the full dataset of binaries and unaries (rather than three separate models per binary). A full hyperparameter optimisation is performed, as described in Section 4.8. The ensemble is constructed by training 10 SNAP models on 10 different, randomly segregated 80%/20% training/cross-validation splits of the full binary/unary database. This amounted to a total of 678 compounds. Predictions on the new compounds are made by averaging the predictions of all 10 models. The training MAE (RMSE) is 2.36 (3.72) meV/atom, while the cross-validation MAE (RMSE) is 5.62 (12.21) meV/atom. This is consistent with the results in Table 4.3. The slight decreases in error observed notably for the RMSEs as compared to the single binary models is a result of the increase of the number of features (bispectrum components) used to train the model to account for the larger dataset. When using this ensemble to predict the energy of all binaries (training and cross-validation sets), the average RMSE across all models is of 6.46 meV/atom.

The energies of the ternary compounds are predicted with excellent accuracy, with a MAE of 3.56 meV/atom and a RMSE of 4.92 meV/atom. Figure 4.10 shows a parity plot of the predictions made by the ensemble model for all binaries (training and cross-validation) and the ternaries. The parity plot shows SNAP's good performance. Such low errors indicate that the model trained is capable of very accurately predicting the energies of ternaries, despite only being trained on a limited, fixed, dataset of binary compounds. The fact that the error is so low on what is effectively a test set, and

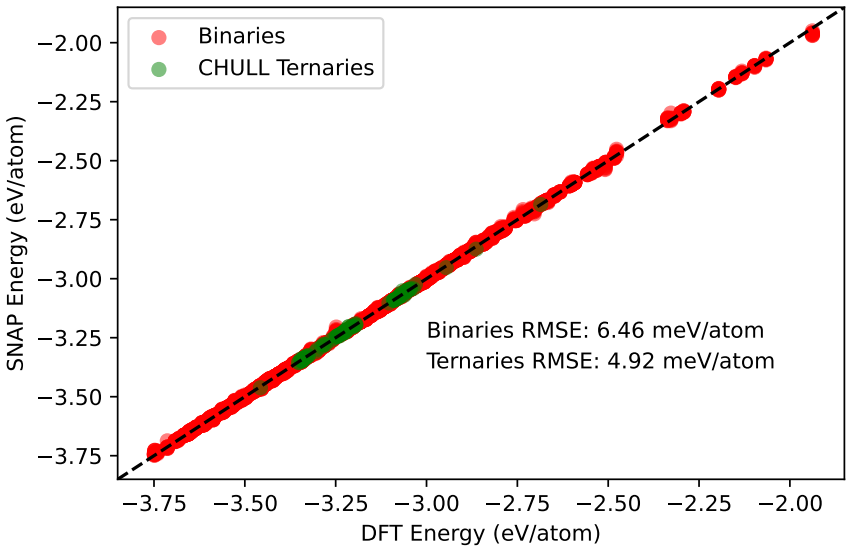


Figure 4.10: Parity plot of the predictions made by the ensemble SNAP model trained on binary compounds and tested on ternary compounds for the Ag-Au-Cu system. The dark dashed parity line is a guide to the eye.

is even lower than that of the cross-validation set, does indicate that the ternaries in the CHULL database do not present much more structural diversity than what is present in the binary database. This is also highlighted by only a small difference between the MAE and RMSE for this set, a fact that indicates the absence of any large outliers. Nevertheless, such structures are AFLOWlib's candidates for their convex hull construction of the Ag-Au-Cu system, and it is non-trivial that a model trained on binaries could perform as well on ternary systems. Before testing the model on other sets of ternary compounds, the model is used to predict the bulk moduli of the ternaries presented in this section.

4.7.2 Ternary Bulk Moduli

As in Section 4.6.2, the SNAP models are used to make predictions of the energies of the equation of state points corresponding to compressed and expanded compounds. This time, the tested compounds are 10 randomly selected ternary compounds with a range of compositions. The three same variant SNAP models as before are used for predictions. The only difference here is that the SNAP + EV and SNAP + EV Edge training sets include the compressed/expanded structures of all three unary compounds

(Ag, Au and the additional Cu), which are always included in training. For the former this corresponds to 45 extra points, while for the latter only the most compressed and expanded structures are retained, leading to an additional 6 training points. An example of the DFT and the three predicted EV curves are shown in Figure 4.11. The quality of the predictions relative to DFT are assessed as in Section 4.6.2 and the results are given in Table 4.5.

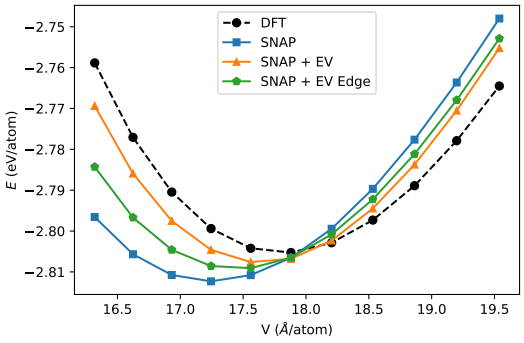


Figure 4.11: Example of an equation of state plot for one ternary compound of AFLOWlib's CHULL database. The different curves correspond to predictions made by DFT (black dots joined with black lines) and three SNAP models constructed differently (coloured markers). The lines are guides to the eye.

Table 4.5: Percentage differences between DFT and the three different SNAP models for V_0, E_0, and B_0 for the 10 ternary compounds. These are obtained following fittings with the Birch-Murnaghan equation.

Model	V_0	E_0	B_0
SNAP	1.2	0.1	14.1
SNAP + EV	0.3	0.1	4.4
SNAP + EV Edge	0.3	0.1	6.8

The results are largely in line with those obtained for Ag-Au compounds, since the errors are of the same order of magnitude for each quantity. For, V_0 and B_0 there is even a slight reduction in error, close to a factor of two. This is due to the high structural similarity of these ternary compounds with the bulk of the binary compounds, highlighted in the previous section by the low test error, as well as the larger dataset used in the SNAP models. In this case, the SNAP + EV Edge results are almost as accurate as the SNAP + EV ones. For E_0 the results worsen for the SNAP + EV and SNAP + EV Edge models, a fact that can be attributed to the compounds not being present in the training/cross-validation sets, as they were in the analysis in Section 4.6.2. This is because the energy is the target property of SNAP. It is interesting to note that in Figure 4.11, all models give the same energy prediction for the true DFT ground state compound, even if they do not agree on which point is the lowest in energy. In contrast, the disagreement between models for highly compressed/expanded

compounds is much larger. This hints at how disagreement between models, a quantity readily available for ensemble models, could be a good indicator of the reliability of predictions. This idea is further explored in Section 4.9.

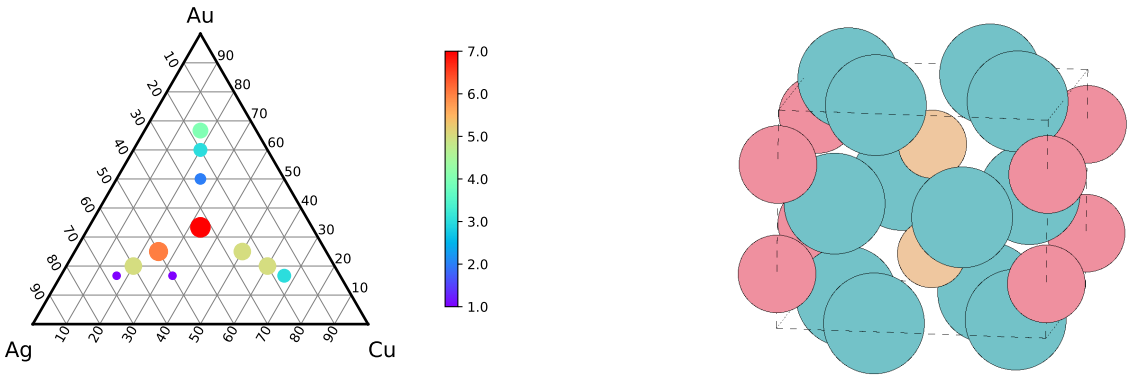
In conclusion, the SNAP ensemble models trained on binaries are highly suitable for predicting the total energy and the equation of state points of the ground-state ternary compounds, close to the convex hull. For the EV predictions, the addition of only 6 unary EV points increases the accuracy of the predictions. The next section describes the limitations of SNAP for predictions on a newly created set of ternary compounds.

4.7.3 Prototype Ternaries

The ensemble SNAP has thus far been tested on equilibrium, low-enthalpy ternary compounds. While this is very promising in the view of using it as a screening tool, further tests are required to assess its limitations. To do so, a new ternary test set is constructed. AFLOWlib is the starting point for these new structures, but rather than using those present in the CHULL database, the compounds are generated using AFLOWlib's prototype library. This offers a more diverse set of structures, even more than for the binary CHULL database, since prototype structures not only originate from the structures of alloys but also from oxides, carbides and other compounds containing main-group elements. The presence of such elements induces the presence of strong valency in the bonding of a material, and hence geometries and atomic coordinations that differ from those found in alloys that are typically close packed.

A selection of 42 different ternary compounds are generated from these prototypes. There are two sources of prototypes. One is made up of "special" ternary prototypes used to generate the convex hull compounds, which have no degrees of freedom aside from the volume of the cell. A subset of these prototypes was used by AFLOW to generate their CHULL database. The structures in this database are as described in Section 4.7.1. Note that the difference to the CHULL database is that the structures here are unrelaxed, and hence in principle are out-of-equilibrium. The second source of prototypes is the AFLOWlib ANRL database, which is made up of structures based on a large range of experimental compounds and is more diverse than the prototype database. For these structures, more structural degrees of freedom can be tuned (lattice

parameters and Wyckoff position parameters). Here, default values are selected, which are those of the underlying compound of the prototype, aside from the volume. For all the compounds generated from these prototypes, only the volume is chosen. It corresponds to the average weighted ground-state unary volumes (similar to Vegard's law). Then the structures are all decorated with the three elements. The prototype imposes a stoichiometry (e.g., $A_1B_1C_2$), however the choice of species substitution is free (e.g., $Ag_1Au_1Cu_2$ or $Ag_1Au_2Cu_1$ are both possible). This is chosen randomly for each prototype. Figures 4.12 below show the distribution of stoichiometries within the prototypes, as well as an example structure used as a prototype. Twenty-six different space groups are represented, with the repeat ones coming from the "special" prototype database, in which space group # 221 is notably present 7 times.



(a) (b)

Figure 4.12: Left: Number of prototype ternary compounds at different stoichiometries. Right: Example of a compound used as a prototype for the ternary prototypes. This is the Nb_4CoSi [297] structure from AFLOWlib.

For all compounds, full *ab-initio* relaxations are performed. The relaxations are terminated for forces below 10^{-3} eV/Å. Thus, for the initial non-relaxed (NR) ternary compounds, the fully relaxed (R) compounds and all those met during the relaxation path, the DFT energies are computed. While it may seem that this produces a vast dataset for testing the model, only the end points are of particular interest. Most compounds encountered during the relaxation are highly redundant and are very similar to one another. Therefore, only the NR and R ternary compound sets are tested. The same model as in Section 4.7 is used to predict the energies of these two sets. The results are shown in Figure 4.13. The MAE (RMSE) on the NR ternary set is of 270.86 (621.22) meV/atom, while for the R ternary set it is of 18.28 (30.20) meV/atom.

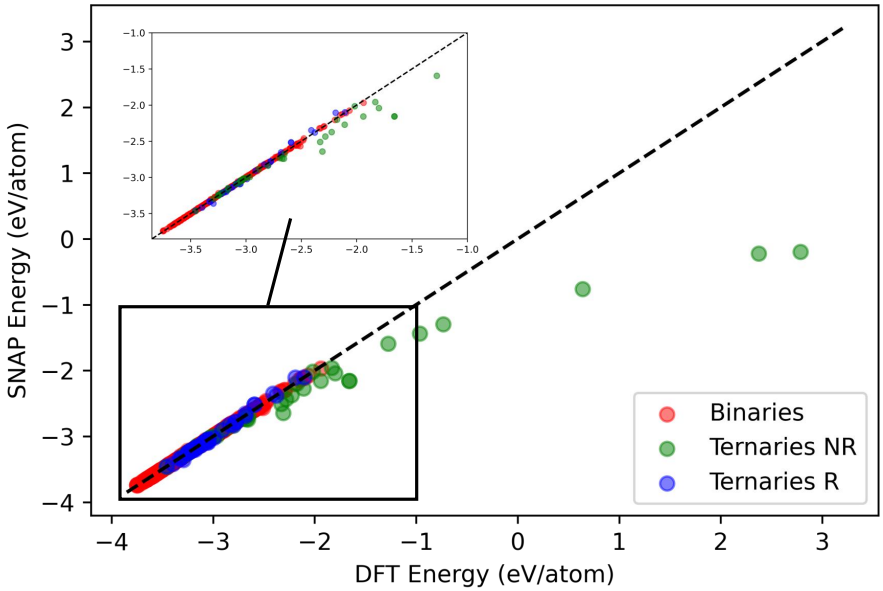
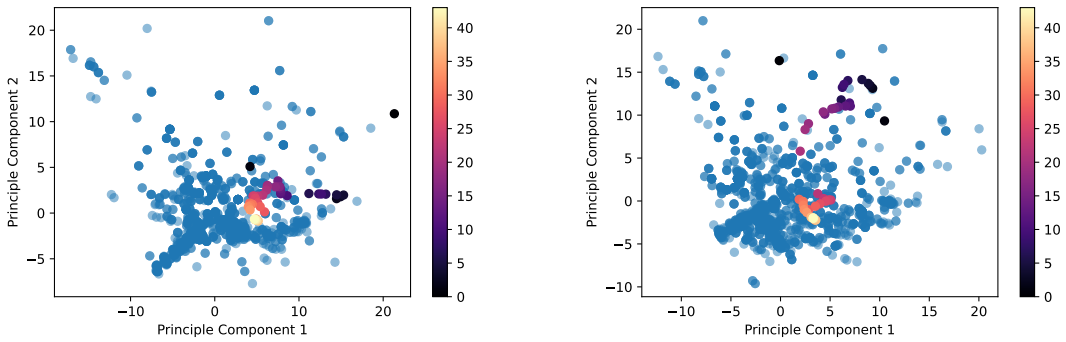


Figure 4.13: Parity plot of the predictions made by the ensemble SNAP model trained on binary compounds and tested on ternary compounds for the Ag-Au-Cu system. The dark dashed parity line is a guide to the eye.

Compared to the previous results, the errors are now larger, notably by two orders of magnitude for the NR set. This is a key result, as it implies that the ensemble SNAP model is not able to accurately predict the energies of unrelaxed compounds. However, following *ab-initio* relaxation of these compounds, the error on the predictions made by SNAP becomes suitable for a screening tool. While the error on the R set is higher than that on the CHULL ternary set, the latter had uncommonly low errors for a test set. The RMSE on the relaxed ternary prototypes is not out of line with the that on the cross-validation of the binary compounds. This is a highly insightful result, since it implies that the equilibrium status of a material is the most important feature that determines whether it is well described by SNAP or not. In other words, if the forces on individual atoms vanish, then SNAP can be used to screen its energy. This is perhaps surprising, since one might think that the structure adopted by a compound may severely influence SNAP's predictive performance. This is only the case if such a structure is unrelaxed. To illustrate this point, one can note that some ternary prototypes may be poor guesses for stable compounds and lie at high energies. In the

parity plot in Figure 4.13 these are the blue points with the highest energies. However, the deviation of such points from the parity line is far lower than that of the green points.

To further illustrate this aspect, a species-wise two-dimensional principal component analysis (PCA) was carried out on the feature space of the training set. The bispectrum components (features) of all compounds in the training set and of all the structures encountered in the relaxation path of one of the ternary prototypes were transformed following this PCA. Results for the Ag and Au feature spaces are shown in Figure 4.14. The notable feature of these plots is that for the starting structures, colour-labelled in dark, the points are in a sparse region of the plot. Therefore, the chemical environments present are different to those commonly found in the training set. On the contrary, for the compounds close to the relaxed state, marked in light colours, the points are in a much denser region, which is expected to be well described by SNAP. This further strengthens the argument that SNAP performs poorly on out-of-equilibrium compounds, rather than on ones with uncommon structures.



(a) (b)

Figure 4.14: Principal component analysis (PCA) plots obtained from species-wise transformations of the training binary compounds. Points are obtained following the projection of the feature vectors of the binary alloys (blue points) and relaxation path compounds (coloured points) of a ternary prototype. The plots for Ag (left) and Au (right) are shown.

These findings beg the question of whether the SNAP ensemble model can be used to perform relaxation. If this were successful, then SNAP could be used to first drive a relaxation when presented with a new ternary compound and then, once relaxed, predict its energy. By averaging the weights of all models of the ensemble (equivalent to the prediction of all 10 models, since SNAP is linear), a SNAP-driven relaxation

is carried out with LAMMPS. The energy and force convergence criteria were set to 10^{-3} eV and 10^{-4} eV/Å, respectively. First the atomic positions were relaxed, before performing a full relaxation (positions + cell parameters). This is repeated five times to ensure adequate convergence. The DFT total energies of the SNAP-relaxed (SR) cells are then computed. For three systems, the SNAP-driven relaxations resulted in non-physical systems, with highly overlapping atoms and their DFT energies could not be computed. The results are shown in Figures 4.15 and 4.16. The former shows the total DFT energies of the SR ternaries as compared to the R and NR sets. For most cases, the SR compounds have the same or a (much) higher energy than the NR compound. This is visible in panel (a) of Figure 4.16. In a handful of cases, the SR energy is significantly lower than for the corresponding NR one, but remains $\sim$ 100 meV/atom higher than the *ab-initio*-relaxed system. In one case, the SR compound has a lower energy than the relaxed compound. For this case, the atoms are so rattled in the SNAP relaxation that a new structure was reached. Note that it is not relaxed with DFT (forces of the order of 10^{-1} eV/Å). However, this compound, even after being relaxed with DFT, has an energy above the convex hull of nearly 197 meV/atom. Overall, these structures remain out-of-equilibrium and their energies are poorly predicted by SNAP, similarly to those of the NR case, as can be seen in panel (b) of Figure 4.16.

This shows that SNAP is not able to drive adequate relaxations, when trained on the energies of the binary compounds. The reasons for this could be expected from the PCA presented above, since the energy and hence force predictions made by SNAP, notably at the start of the relaxation, are inaccurate due to a lack of knowledge of the relevant feature space. This results in a poorly directed relaxation from the start and inevitably to a poorly relaxed final structure. The main conclusion is that SNAP cannot be used to drive relaxation.

One last attempt at improving the relaxing-ability of SNAP consists in training SNAP on the forces on atoms of the binary compounds. These are only available because the energies are recomputed for the Ag-Au-Cu system. In fact, they are generally not available from AFLOWlib, unlike the energies, and hence could not be readily integrated in the training of a screening tool based solely on AFLOWlib data. However, for completeness, forces are used for fitting here. Results from the fitting of the binary and unary compound forces are show in Figure 4.17. The results of two

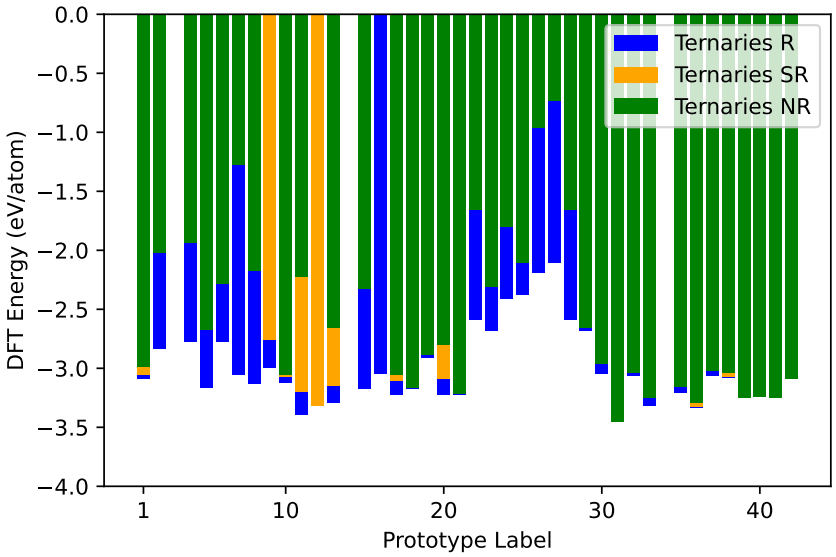
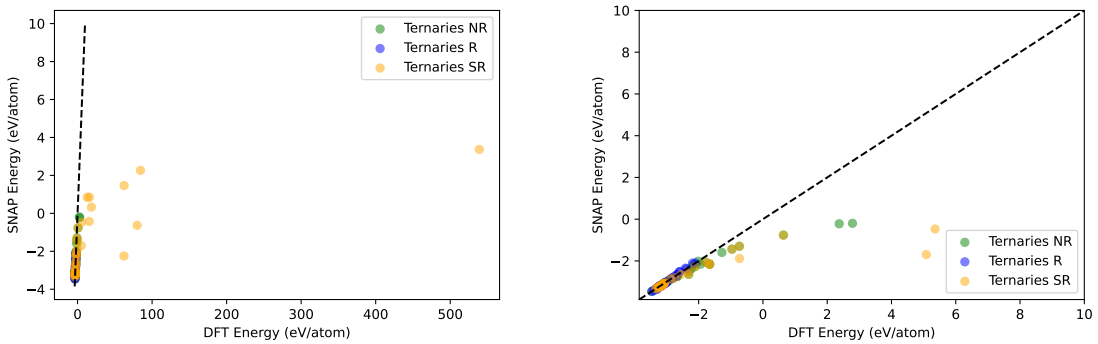


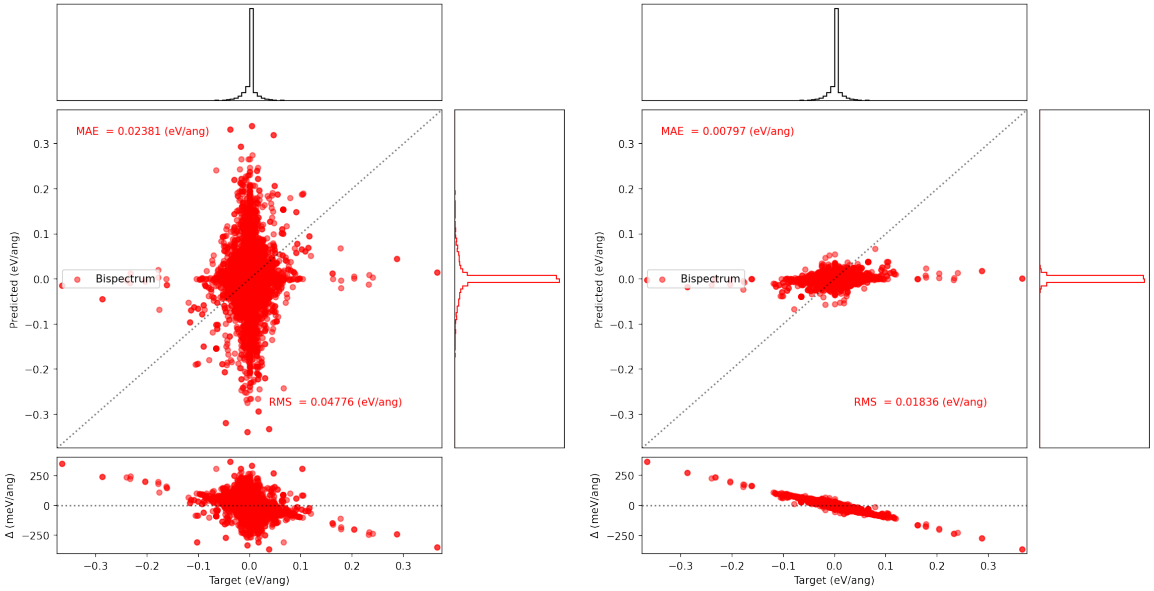
Figure 4.15: Plot showing the results from the SNAP driven relaxations. A histogram of the DFT energies of the non-relaxed (NR), SNAP-relaxed (SR) and relaxed (R) ternary prototypes is displayed. The missing bars are associated with prototypes for which the DFT relaxation failed after the SNAP-driven relaxation.



(a) (b)

Figure 4.16: Plots showing the results from the SNAP driven relaxations. Panels (a) and (b) [zoom of (a) up to 10 eV/atom] show the SNAP predictions for the three sets, the non-relaxed (NR), SNAP-relaxed (SR) and relaxed (R) ternary prototypes. The dashed black parity lines are a guide to the eye.

models are illustrated (one trained on energies and the other on forces only), and in both cases, even when the forces are fit (i.e., the training set is solely comprised of forces), the results are poor. This is due to the fact that the force components are generally very small (10^{-2} eV/Å), since the compounds are at equilibrium. Adding the convex hull binary compound forces to the training process would hence not improve the force predictions and improve the relaxation process.



(a) (b)

Figure 4.17: Parity plots of the fitting of forces for the binary compounds when fitted on energies and forces (a) and only on forces (b). In (a), both energies and forces are fitted with equal relative weighting, where the weights on energies and forces are 1 and $\frac{1}{3N}$, respectively. N is the mean number of atoms per system. This is to account for the fact that there are $3N$ more force components than energies in the fitting, and is standard procedure when fitting both energies and forces. In the (b), only forces are fitted.

In conclusion, the SNAP ensemble model, trained on binary compounds from the AFLOWlib database can be used as an effective screening tool, provided that the compounds being screened are close to their equilibrium. If they are non-relaxed, the predictions are not quantitatively accurate, but the energy scale and energy ordering of compounds is reliable. The model is not able to drive relaxation due to an absence of relevant out-of-equilibrium structures in the database. Before concluding and discussing how such a screening tool may be used in practice, the procedure followed to build the screening tool is repeated for two new ternaries, Mo-Pt-Ti and Cd-Hf-Rh.

4.7.4 Extension to Mo-Pt-Ti and Cd-Hf-Rh

In order to validate the previous findings, two similarly constructed screening tools are tested for Mo-Pt-Ti and Cd-Hf-Rh. The choice of these two ternary systems is motivated by two main factors. The first is that more complex alloy systems, compared to the noble metal trio Ag-Au-Cu, are sought after. In this regard, both ternaries are

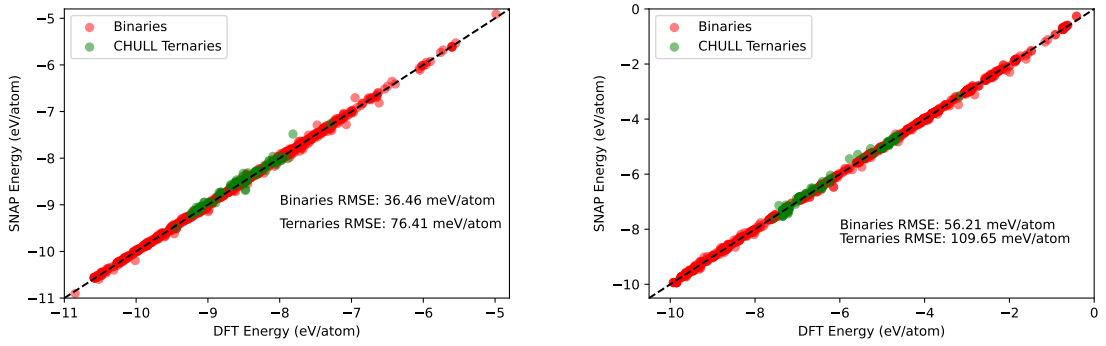
Table 4.6: Number of data points available on the unary, binary and ternary systems associated with Mo-Pt-Ti and Cd-Hf-Rh.

X-*Y*-*Z* Ternary	*X*	*Y*	*Z*	*X*-*Y*	*X*-*Z*	*Y*-*Z*	*X*-*Y*-*Z*
Mo-Pt-Ti	9	9	8	195	807	196	86
Cd-Hf-Rh	26	26	26	615	245	196	96

chosen such as to have an early transition metal (Ti/Hf), an intermediate one (Mo/Rh) and a late one (Pt/Cd). It is expected that the wider diversity in the number of valence electrons, as well as the larger size mismatches between species should lead to more specific bonding between different atomic pairs and hence more specific structural features than close packed geometries. The second important deciding factor is the number of unary, binary and ternary compounds available on AFLOWlib for each system. This determined the final choice. In fact, in AFLOWlib there are many refractory metal (Mo/Hf) binaries, since certain of these systems were the object of a more intensive high-throughput convex hull determination [298, 299] than most systems on AFLOWlib. The Mo-Pt-Ti and Cd-Hf-Rh systems, respectively, have 1,224 and 1,179 binary+unary compounds in the AFLOWlib database. In each case, one binary (Mo-Ti and Cd-Hf) is over-represented, with 3-4 times more compounds than the other two binaries. The details on the number of different compounds in each database is given in Table 4.6. For all binaries, there is at least one experimental compound included in the database, whose structure is present in the ICSD.

Ensemble SNAP models are trained for each ternary system, following the same procedure as previously. 10 models are trained on 80%/20% splits of the binary data, following the standard hyperparameter optimisation to be presented in Section 4.8. The ternary compounds make up the test data. The only important difference here, aside from the ternaries chosen, is that the energies of the compounds are not computed for the study and the AFLOWlib total energy values are directly taken. For Mo-Pt-Ti, the MAE (RMSE) is 18.0 (27.9) meV/atom for the training set, 29.8 (58.7) meV/atom for the CV set and 64.0 (76.4) meV/atom for the test set. For Cd-Hf-Rh, the MAE (RMSE) is 27.4 (42.9) meV/atom for the training set, 44.4 (89.4) meV/atom for the CV set and 87.2 (109.7) meV/atom for the test set. The parity plots of the ensemble predictions are shown in Figure 4.18. Overall, the relaxed convex hull compound

points follow the parity line well. The binaries' RMSE corresponds to the error on all binaries (training + cross-validation sets). The errors for these systems are one order of magnitude higher than those of the Ag-Au-Cu system. This reflects the larger energy range fitted, the increased complexity of the systems considered, as well as the increased noise of the data from the looser DFT convergence criteria employed by AFLOWlib. Note that, while the errors are higher, this is consistent across training, cross-validation and test sets and is not just an increase in the test set error. In fact, the test error is largely in line with the cross-validation error on binary systems. These results suggest that it is still reasonable to use a model trained on binaries to screen ternary compounds. The errors are of the order of 10^{-2} meV/atom, but the model can be trained directly from AFLOWlib, without the need to re-generate any DFT data. Note that, compared to Ag-Au-Cu, the total-energy percentage error is similar. In fact, for Ag-Au-Cu, the percentage error relative to the average energy per atom was of 0.19% for the training set and 0.15% for the test set. These must be compared to 0.40% and 0.86% for Mo-Pt-Ti and to 0.74% and 1.57% for Cd-Hf-Rh. It is important to highlight that the ternary convex hull diagrams of Mo-Pt-Ti and Cd-Hf-Rh exhibit a much greater depth compared to that of Ag-Au-Cu. The lowest points on these diagrams have enthalpies of formation measuring 951 meV/atom and 921 meV/atom, respectively, whereas for Ag-Au-Cu, the enthalpy of formation is merely 61 meV/atom. As a result, the absolute error in SNAP becomes less significant in this context.



(a) (b)

Figure 4.18: Parity plots showing the predictions of the ensemble SNAP models for Mo-Pt-Ti (left) and Cd-Hf-Rh (right). Predictions are shown for the binaries (training+cross-validation) and for the ternaries (test). The dashed black parity line is a guide to the eye.

Before presenting the conclusions of this work, the process for hyperparameter optimisation is presented. Finally, various error metrics that could be used to detect the degree of extrapolation of a prediction are tested in Section 4.9.

4.8 Hyperparameter Optimisation

This section focuses on the methodology followed to optimise the model hyperparameters. These correspond to model parameters that are not determined during the fitting procedure of the training dataset, i.e., the training process. In that step, only the linear $\vec{\theta}$ parameters appearing in the definition of the ridge regressor are determined from the least-mean-squares fit. All other parameters are tuned through the use of the cross-validation (CV) set, not employed for training. In the context of the ensemble of SNAP models, there are three categories of hyperparameters. The first concerns those that impact the construction of the feature vectors, the bispectrum components. These are the cut-off radius R_{cut} and the atomic weights $\{w^{\alpha}\}$ (α denotes the atomic species) that appear in the atomic density function in Equation (3.24), J_{max}, that determines the highest order of bispectrum used for the construction of the feature vectors and hence influences the final number of components and that are used. Technically, r_0, from Equation (3.25), is also a hyperparameter but in this study it is not tuned, which is common practice. The second class of hyperparameters are made up of those that concern the machine-learning model. In the case of ridge regression, there is only one, namely the regularisation term λ. Finally, the last category concerns the parameters pertinent to the construction of the ensemble. These are N_{models}, the number of models within the ensemble, and the proportion of the binary dataset used for training/cross-validation.

The process for hyperparameter tuning is determined from tests on the individual and combined binary Ag-Au-Cu datasets. Repeated k-fold cross-validation (Monte Carlo CV) is used to split the dataset into training and cross-validation sets. It is chosen since it is a robust cross-validation technique [300]. This split is performed N_{models}=10 times, such that the hyperparameter optimisation did not include a bias from one particular random split. Note that these splits are retained for the construction of the ensemble. The choice of the value of N_{models} is from usual practice

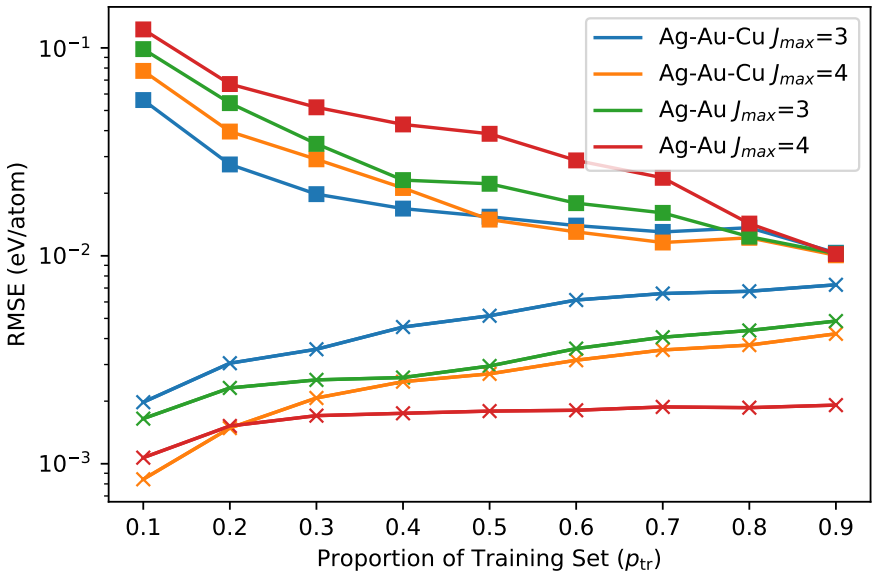


Figure 4.19: Learning curves of the SNAP ensemble models for Ag-Au-Cu (three binaries) and the single Ag-Au binary system. Crosses represent errors on the training set, while squares are for the cross-validation sets. The lines are guides to the eye. The RMSE scale is logarithmic.

in machine learning operations [300], and is not specifically tuned here. The optimisation process consists in selecting different hyperparameters, training N_{models} models on N_{models} different training sets and calculating the error on each model's associated cross-validation set. The errors are averaged and the hyperparameters that generate the lowest errors are selected as the optimal ones. Individual hyperparameters are optimised in different steps.

At first, J_{max} and the training/cross-validation proportions, quantified by p_{tr}, the proportion used for training, were tuned together through the use of learning curves, shown in Figure 4.19. It is optimal to have a low cross-validation error without any overfitting. This means having a lower and stagnant training error. The curves for Ag-Au J_{max}=4 clearly show this trend, since beyond p_{tr}=0.4 the training error is stagnant. Clearly for Ag-Au, J_{max}=3 is optimal, which corresponds to 62 (31×2) components in the feature vectors. For Ag-Au-Cu, both J_{max}=3 and 4 seem suitable, with similar cross-validation errors beyond p_{tr}=0.4. In this case, J_{max}=4 is chosen, since the learning curves suggest that in the case additional data were available, further decreases in training and cross-validation errors could be achieved, unlike the J_{max}=3 case. This corresponds to 168 (3×56) components. The optimal value for p_{tr} is 0.8, since beyond

this value, the number of compounds in the cross-validation set becomes too low for reliable error quantification. Below these values, cross-validation errors are higher. The results obtained for Ag-Au also apply for the other binaries. The values for $J_{\max}$ are consistent with those found in other studies with similar data set sizes [207, 301].

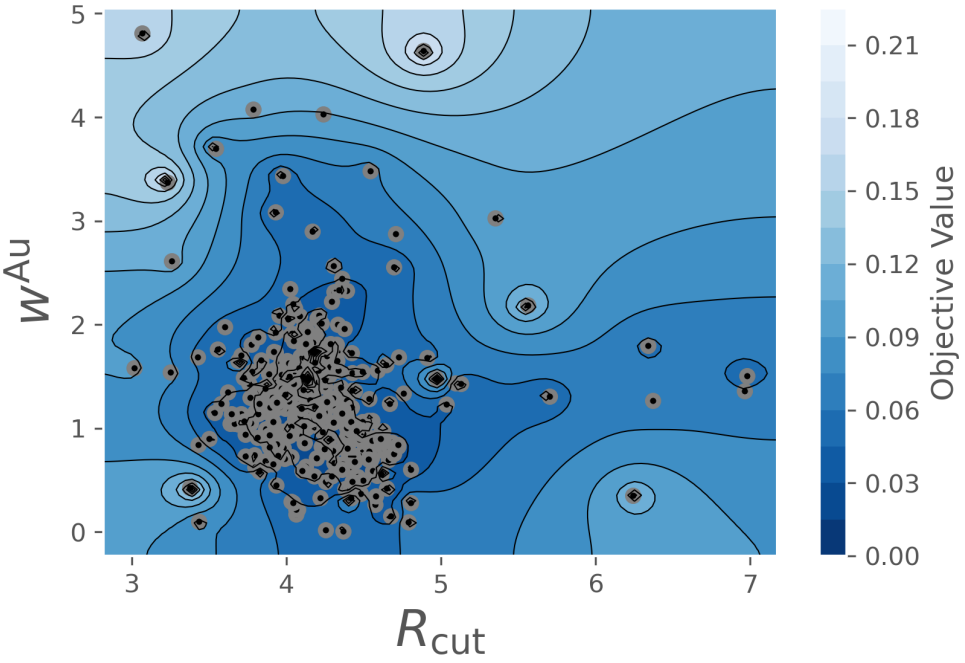


Figure 4.20: Contour plot generated by Optuna [302] showing the parameter relationships for one pair of hyperparameters, R_{cut} and w^{Au}. The colour scale maps the cross-validation RMSE, which is the objective value.

The last three hyperparameters tuned are R_{cut}, $\{w^{\alpha}\}$ and λ. These can all be floating point values, and the precise optimisation of all together can lead to large decreases in the RMSE. Since the search space for the optimal values is large (5-dimensional for three species), it is crucial to use efficient, guided search algorithms, such as to only spend search time in optimal areas of the search space. The algorithm of choice is the Tree-structured Parzen Estimator (TPE) [303, 304, 305]. It is used as implemented in Optuna [302], a Python library. This is a type of Bayesian optimiser, which is the state of the art for hyperparameter optimisation [165, 207]. Searches were run to optimise R_{cut} in the range 3 Å to 7 Å, $\{w^{\alpha}\}$ from 0 to 5 and λ from 10^{-6} to 10^{0} using logarithmic sampling. It was discovered that λ had very little impact on model performance ($\leq 0.1\%$), and only slowed the search. Therefore, it was fixed at 10^{-2}, a typical optimal value observed across the first test runs. For R_{cut} and $\{w^{\alpha}\}$, 50 different runs were performed with 400 search steps (different sets of hyperparameters), since

some runs become stuck in local minima with unsuitable hyperparameters. Among all the results, the hyperparameters that minimised the cross-validation set RMSE were chosen. The RMSE to hyperparameter mapping is represented in the contour plot in Figure 4.20, which is a result of the best optimisation for the Ag-Au-Cu model. The large number of points in the regions where the objective value is low shows how much of the compute effort is aimed at studying the region in which the CV RMSE is lowest.

Table 4.7: Summary of the optimal hyperparameters for the SNAP models trained for the individual binary systems. Here w^X and w^Y refer to the weights of the X and Y species of the X-Y binary system.

Hyperparameter	Ag-Au	Ag-Cu	Au-Cu
$J_{\max}$	3	3	3
R_{cut} (Å)	4.144006	4.358127	4.452703
w^X	0.526867	0.225576	0.341620
w^Y	0.701251	0.174029	0.193797

All hyperparameters are optimised following the methodology presented in this section. N_{models}, p_{tr} and λ are always constant at 10, 0.8 and 10^{-2} respectively. For all other hyperparameters, the final values for all the models trained are presented in Tables 4.7 and 4.8. In the final subsection of this chapter, different error metrics are presented, and their utility is assessed.

Table 4.8: Summary of the hyperparameters used for the ternary SNAP models. Here w^X, w^Y and w^Z refer to the weights of the X, Y and Z species of the X-Y-Z ternary system.

Hyperparameter	Ag-Au-Cu	Mo-Pt-Ti	Cd-Hf-Rh
$J_{\max}$	4	4	4
R_c (Å)	4.647073	4.917926	6.436145
w^X	0.305086	0.366659	3.107010
w^Y	0.245647	0.161844	4.293181
w^Z	0.418890	0.35940283	3.193081

4.9 Error Metrics

For this final section, the attention returns to the Ag-Au-Cu ternary system. In Sections 4.7.1 and 4.7.3, an ensemble SNAP model was presented and was used to predict the energies of three classes of ternary compounds: those present in the AFLOWlib database (CHULL ternaries), non-relaxed prototype ternary compounds (NR ternaries) and the corresponding DFT-relaxed compounds (R ternaries). One major attribute of the SNAP models is that it cannot accurately predict the energies of out-of-equilibrium compounds. Therefore, to make use of it as a screening tool, it is necessary to only employ it for equilibrium compounds. The equilibrium status of a compound is not known *a priori*, since it requires the knowledge of the DFT forces acting on the constituent atoms. Such calculations are what the screening model aims to circumvent. Therefore, it is necessary to identify such compounds without the use of DFT. One way to do so is through the use of error metrics, which are able to quantify the error made by a SNAP model when it makes a prediction. Crucially, these should not rely on DFT, so the real SNAP error with respect to DFT is not an error metric. Instead, they should rely on the SNAP model or the feature vectors that represent individual compounds. Three such error metrics are studied here. They will first be presented before assessing how well they correlate with the real SNAP error with respect to DFT, denoted as ϵ. This is in the context of making a prediction on a new "unseen" compound, which is not in the training database. The aim is to evaluate how accurate the model prediction will be before any DFT.

The first of these error metrics is the minimum Euclidean distance $d_{\min}$ of a compound's feature vector to any in the training set Ω_{tr}. The idea behind this metric is that, if for a new material there is a compound in the training set that is similar (in feature space), then the SNAP prediction will be good. It is assumed that SNAP has "learnt" the relevant geometry to energy mapping from the structure in the training set. For a compound labelled i, it is defined as

$$d_{\min} = \min \left\{ ||\vec{B}_j - \vec{B}_i|| \right\}_{j \in \Omega_{\text{tr}}}, \tag{4.11}$$

where $\vec{B}$ is the bispectrum component vector. Note that there are several ways to define $d_{\min}$, since the bispectrum components can be for the global structure one or

for a single atom. For a system with N atoms and K species, there are N bispectrum vectors. In the definition of SNAP, there are separate fitting coefficients for each atomic species that are multiplied by the corresponding atomic bispectrum coefficients. From this first standpoint, the relevant feature space is species specific. Alternatively, for each compound, all bispectrum of the same species are summed together, and a concatenation of all bispectrum is performed, yielding a single vector per compound. In this case, the feature space becomes a global structural one that smears the individual atomic contributions. If $\vec{B}$ is taken as being atom-specific, $d_{\min}$ has a more physics-intuitive definition, since individual chemical environments are compared. In the case the total $\vec{B}$ vector is used, individual atomic contributions are washed away in the sum. However, the definition of $d_{\min}$ better reflects the mathematical grounding of the problem. This is because ultimately, it is the total, concatenated vector that is used in the fitting procedure, given by Equation (3.32).

In the case of the total vector, Equation (4.11) uniquely defined $d_{\min}$ directly. For the species-specific features, if there is more than one atom per species, one value must be selected. Since the aim is to quantify how much uncertainty there is in the SNAP prediction, the maximum value is chosen, as even if one chemical environment is poorly described (far from the training set), the overall prediction may be poor. Finally, since $d_{\min}$ is species specific, so there are K different values.

The second error metric is based on the model built in this work, the ensemble of SNAP models. The N_{models} different SNAP models, trained on different random subsets of the training data, make a collective prediction by taking the average of each of their individual inferences. Another quantity of interest is the standard deviation of their predictions σ. This quantifies the degree of disagreement between the models. It is postulated that this may be a good error metric. In the machine-learning field, it is named query-by-committee [300]. It is different to $d_{\min}$ in that it is non-species-specific, as there is only one value per compound (since only the total energy of a compound is predicted). It also makes use of the energy-to-structure mapping, implicitly encoded in the individual SNAP models, rather than just relying on the feature space description. It also comes at essentially no extra computational cost, whereas $d_{\min}$ requires many Euclidian distance calculations, as well as the use of a sorting algorithm to determine the minimum distance. The σ metric has notably been used in the context of hyper-

active learning for GAP models [306], as a proxy to the standard deviation of the posterior-predictive distribution.

The final error metric tested is one that has been used quite extensively in the training of MLIAPs [106, 171, 175, 307, 308]. It is the extrapolation grade γ, first introduced by Podryabinkin *et al.* [106]. Its definition is more technical than that of the other two metrics and comes from the statistical theory of linear-least-squares method. It is explained in some detail here. The metric relies on the concept of D-optimality design [309], extensively used in experimental design. In this field, one is interested in estimating how p experimental parameters influence an analytical response in an experiment. When a linear model relating this response to the parameters is assumed, the concept of D-optimality is used to determine which candidate experiments, performed at specific conditions for the p parameters, would be optimal to fit the model. It is done to minimise the number of experiments performed, since these may be expensive. Let us consider the case where there are N_{exp} possible candidate experiments, determined from a grid sampling of the all the possible combinations of values the p parameters may take. The experiments with the values of the parameters used make up the $(N_{\text{exp}} \times p)$ matrix of candidate points. Each row represents an experiment with p values for each parameter. The selection of the best m candidates for the model results in a $(m \times p)$ model matrix $\mathbf{X}$. The D-criterion is used to select this matrix and hence best candidates. This criterion was proposed by Smith *et al.* [310] and postulates that the optimal model matrix, out of all those that may be chosen, is that which maximises the determinant of the information matrix $\left(\mathbf{X}\mathbf{X}^T\right)$. This corresponds to minimising the influence of experimental error on the determination of the model coefficients. In the case of MLIAPs, these errors correspond to noise arising from numerical errors in the DFT calculations and from the fact that atomic descriptors are local and hence the atomic neighbourhood used is truncated at R_{cut}. This implies that non-local interactions are not accounted for.

The number m of experiments to be performed has no optimal value. It can be chosen freely, and it typically depends on external constraints. The lower bound is the number of parameters p, such that $\mathbf{X}$ is a $(p \times p)$ square matrix. This is chosen here.

We now bring the discussion of the D-optimality criterion to the case of training databases for MLIAPs. Now, $\mathbf{X}$ is made of different configurations rather than candi-

date experiments. These configurations may be feature vectors for the whole structures or individual atoms, as will be discussed. Now, m represents the number of compounds in the training database and p the number of components in the descriptors. Finding the D-optimal matrix $\mathbf{X}$ requires the use of efficient algorithms, since a brute-force calculation of the determinants of all possible information matrices is very costly. In the procedure followed, the MAXVOL [311] algorithm is employed. It is designed to find the optimal square submatrix of size $(p \times p)$ from a large $(m \times p)$ rectangular matrix, where optimal means largest determinant. The authors of Reference [106] claim that the problem of finding the optimal $\mathbf{X}$ matrix can be recast [171]. The aim is to determine which p configurations, $\vec{B}_i$, should make up $\mathbf{X}$. The algorithm iteratively updates which square matrix $\mathbf{X}'$ is the optimal one. To know whether a row i of the full dataset $\mathbf{X}$ should substitute a configuration (row) in $\mathbf{X}'$, the extrapolation grade γ is used, defined as

$$\gamma = \max_{1 \leq j \leq p} |c_j| \,, \tag{4.12}$$

where c_j are the components of $\vec{c}$, namely

$$\vec{c} = \vec{B}_i^T \left(\mathbf{X}^T \mathbf{X}\right)^{-1} \mathbf{X}^T = \vec{B}_i^T \mathbf{X}^{-1}. \tag{4.13}$$

If the value γ is larger than 1, then a model based on $\mathbf{X}'$ extrapolates on configuration represented in row i, therefore $\vec{B}_i$ should be substituted into $\mathbf{X}'$. It should, in fact, be exchanged at the row j of $\mathbf{X}'$, corresponding to the index of the maximum component of Equation (4.12). In other words, the configuration represented by $\vec{B}_i$ is more suitable than the configuration $\vec{B}_j$ for the model training.

The exchange of rows based on the value of γ implies that the problem becomes that of optimising the determinant of $\mathbf{X}$. MAXVOL, hence, is an algorithm for finding $\mathbf{X}$ by using several trial matrices $\mathbf{X}'$ and testing which configurations should be placed in it based on calculations of γ for all configurations. This is repeated until all configurations not present in $\mathbf{X}'$ have values of $\gamma < 1$. In this case, the model trained on $\mathbf{X}'$ will interpolate when making predictions on all configurations not included in the training set.

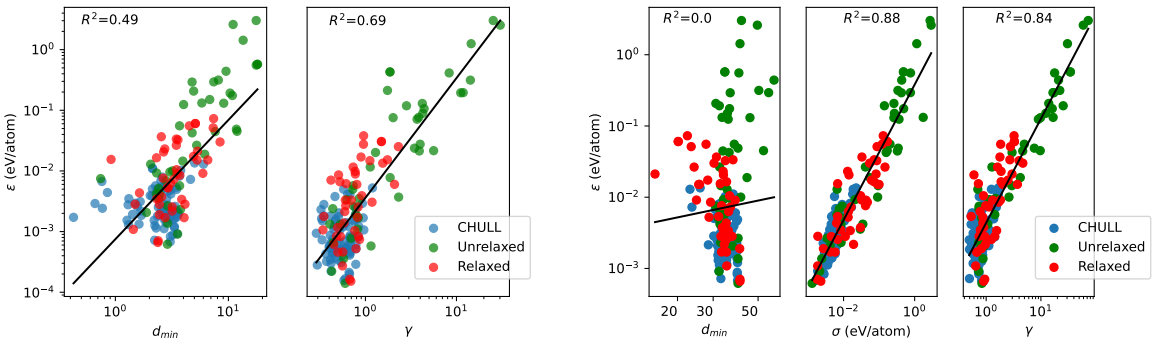
The algorithm is run to obtain the active-learning set, the matrix $\mathbf{X}$ and its in-

verse along with the *active set*, which are the associated p configurations. These are required in Equation (4.13). In the many studies of A. Shapeev and collaborators, the extrapolation grade is used for the construction of the MLIAP in an active learning scheme, as described in Section 4.6. Once an initial training set is constructed and the MLIAP is used on new configurations, γ is computed to determine whether the model is performing an interpolative prediction, as desired, or not. In practice, if γ is beyond a threshold value, the model is retrained by incorporating the new configuration encountered.

In this study, very similar information is sought. The aim of an error metric is to quantify the uncertainty of a prediction, and, if the SNAP model is highly extrapolating, the uncertainty will be expected to be high. It has been shown that γ correlates with the MLIAP error with respect to DFT in the case of MTPs [106]. As for the minimum distance metric, the γ extrapolation grade can be made global or species-specific. The former is done if whole structures are considered, instead of single atomic environments, and hence the feature vectors used to populate $\mathbf{X}$ are the total ones, the sum of all the bispectrum components of the constituent atoms. In the case of species-specific, the computation of γ is performed separately for each species and the bispectrum of each atom are used in $\mathbf{X}$. Note that these correspond to the Query Strategies (QS) 1 and 3, respectively, from Reference [106]. For the species-specific case, QS 3 would be only for energies.

Following these definitions, it is shown how well these metrics correlate with the SNAP error, ϵ. This is performed for the three ternary sets CHULL, NR and R. In order to quantify the correlation, a linear fit on ϵ and each error metric is performed on all compounds of the three sets together. The value of the coefficient of determination R^2 is calculated and used to assess the quality of the correlation. The correlation plots for the three metrics are shown in Figure 4.21. The plots, as well as the linear fits, are performed on a logarithmic scale, since several orders of magnitude are spanned. The values of R^2 are given in the subplots. Note that for $d_{\min}$ and γ, the specie-specific metric are computed in panel (b). The species are chosen based on the best fitting out of all three. For $d_{\min}$ it is Au, while it is Ag for γ, with respective values for R^2 of 0.49 and 0.69. For $d_{\min}$ the R^2 values are 0.35 and 0.34 for Ag and Cu respectively, while for γ the values are 0.66 and 0.63 for Au and Cu respectively. Finally, σ, is,

by construction, non-species specific and is shown in panel (b), along with non-species specific constructions of $d_{\min}$ and γ.



(a) (b)

Figure 4.21: Plots of the ensemble SNAP error compared to DFT against three different error metrics $d_{\min}$, σ and γ for three sets of ternary compounds. The left panel (a) show two species-specific metrics. $d_{\min}$ (left) is computed for the Au feature space and γ (right) is calculated for the Ag feature space. The plots in the right panel (b) show results for total structure feature vectors. $d_{\min}$ (left), σ (centre) and γ (right) are shown. The black lines are the linear fits performed on the whole dataset for each error metric. Their corresponding R^2 values are given.

Overall, all plots show correlation between ϵ and the error metrics, aside from $d_{\min}$ when calculated for with global structure fingerprints. As a general feature for all plots, there is a large group of points in the error range between 10^{-3}–10^{-1} eV/atom. They consist of compounds from the CHULL set, which have a total error range (maximum minus minimum error) of 12.8 meV/atom and the R ternaries, which have an error range of 72.2 meV/atom. Even a significant proportion of the NR compounds are in this range, since some are close to their equilibrium structures. This implies that the correlation is less pronounced for such compounds. The fit becomes better for the NR structures that are far from equilibrium. The poorest-performing metric is $d_{\min}$, which shows low correlation for the species-specific descriptors, notably at low values of ϵ. The metric performs best in the Ag feature space, but the fit still has a lower R^2 than those of γ for any of the three species. When using $d_{\min}$ with global feature vectors, almost no correlation is observed (R^2 of 0.003). The reason for such poor performance can be rationalised by the choice of the training set. Indeed, since the latter is made only of binary systems, by constructing a global feature vector, there are necessarily 56 bispectrum components equal to 0, due to the absence of one of the three ternary

species. When the Euclidian distance is calculated for all test compounds, $d_{\min}$ is dominated by a distance equal to the norm of the bispectrum of a single specie. This is of the order of 10. Such a scale should be compared to the scale observed for $d_{\min}$ for a single specie in Figure 4.21a, which is between 10^{-1} and 1 for most compounds and only reaches 10 for the highest error compounds. The systematic shift induced by the absence of ternaries in the training database therefore erases the finer details arising from the relevant differences between training and test datasets. This could be remedied in several ways by redefining the metric. The global feature vector could be compared to the mean feature of the training database. Alternatively, the selection of a global $d_{\min}$ could be computed from the species-specific values, by taking the average of maximum values. The latter was notably performed for the publication associated with this chapter [31], showing better correlation than even the species-specific metrics. Nevertheless, $d_{\min}$ is not a suitable metric for the model trained, and the two other metrics show better correlation with ϵ.

The extrapolation grade shows good correlation with ϵ for the species-specific metric. The correlation is moderate in the low error region ($\leq 10^{-2}$ eV/atom) for the species-specific and the global feature spaces. This is in the range of the model error. Interestingly, in this region, the extrapolation grade is below 1, showing that the model is interpolating, as expected. It may, therefore, be expected that the correlation is less pronounced. As one moves beyond 1, the correlation becomes sharper and it is in fact better for the global structure feature space. Values of γ beyond 2 are reached by many NR compounds, which in the active learning scheme of Reference [308] is sufficient for re-training the model. The most poorly described compounds have values beyond 10, which is cause for terminating predictions due to "dangerous extrapolation" according to the same reference. This is in sharp contrast with $d_{\min}$, which is perhaps surprising, considering it should in principle suffer from the same shortcomings due to the common lack of ternaries in the training database. This effect may be attributed to the fact that the extrapolation grade makes use of the full active learning set, rather than a single point of the training set. As a result, the boundary between interpolation and extrapolation is better defined. This boundary is best defined for the global structure feature vectors, which is the one used for fitting. This error metric is therefore suitable for detecting compounds for which the model extrapolates.

The standard deviation of the predictions made by the SNAP ensemble, σ, is the best performing metric and is only defined globally. It is, in fact, even effective in the low error region. The energy scale spanned by σ equally correlates well with ϵ. The slightly increased performance of the standard deviation metric over the others may be attributed to several factors. Firstly, it is inherently a global metric and, hence, takes into account the full compound rather than single atoms. It equally relies on the knowledge of the full range of training binary compounds, since it makes use of SNAP models trained on such databases. The main difference with the other two metrics is that there is also in-built knowledge of how well the model fits the energies. The other two metrics rely solely on the feature representation of the compounds, but not how well these are mapped to the energies. If the model were not able to perform this mapping effectively, due to noise or the assumption that the energy and features are linearly related, then $d_{\min}$ and γ would suffer, whereas σ would not. It should be emphasized as well that the calculation of σ comes at very little extra cost to making predictions. The most computationally demanding step in the construction of a SNAP model is the calculation of the bispectrum components. Here, this is only required once, since the construction of the ensemble of SNAP models is carried out by training different models on different subsets of the same training database. Once the coefficients $\vec{\theta}$ of the SNAP models are computed, the prediction of the mean and standard deviation of the ensemble is fast. It is therefore concluded that the most effective error metric of those tested is σ, as it can be efficiently used to detect compounds that are problematic. The work in this chapter is now summarised in the conclusion and put in the context of a more global workflow aimed at searching novel materials.

4.10 Conclusions

Upon closing this chapter, the main outcomes of the study are summarised. The field of inverse materials design aims to unravel novel materials with specifically targeted key properties. A crucial step in this process is the determination of whether a candidate material is stable or not, which is not a given in *ab-initio* studies. This can be established by first computing the Gibbs free energy of a compound, a calculation which requires free energy contributions from enthalpy, as well as many entropic finite

temperature terms, which are computationally expensive. This must then be compared to that of all known competing phases, such as to build a full phase diagram. A more sober approach only requires the calculation of the total DFT electronic energy of each compound and the construction of the relevant T=0 K phase diagram, the convex hull. For a ternary system, this nevertheless requires hundreds of expensive DFT calculations to construct the three binary and single ternary convex hulls. In the study presented in this chapter, it is shown how a screening tool, built on pre-trained data can be used as a surrogate to DFT.

The ternary screening tool is an ensemble of SNAP MLIAPs, trained on readily available unary and binary convex hull data from AFLOWlib. The choice of the training database is justified by the wealth of data present in binary phase diagrams (as opposed to ternary ones), as well as using pre-existing data that is directly relevant for the convex hull construction. A model is trained for the Ag-Au-Cu ternary system, a simple system made of noble metals for which the three binary phase diagrams display rather different features. It is shown that, upon attentive hyperparameter optimisation, an accurate ternary screening ensemble SNAP model can be trained uniquely on binaries, with training and test errors well within the acceptable range (10^{-2} meV/atom) for an adequate screening tool. The bulk modulus of ternary compounds can also be predicted upon addition of equation-of-state unary compounds to the training database.

The screening tool is, however, only effective if predictions are made for compounds at equilibrium. Otherwise, due to the lack of out-of-equilibrium structures in the training database, the machine-learning model must extrapolate, leading to inaccurate predictions. It is emphasized that despite this shortcoming, such high enthalpy, unrelaxed compounds are still estimated to be high in energy by SNAP, validating its use as a screening tool. The models are not able to drive ionic relaxation towards equilibrium for non-relaxed compounds, as the configurations encountered during the relaxation path are also outside the scope of the training data.

The methodology for constructing the screening tool is also deployed to build two other SNAP models, for the Mo-Pt-Ti and Cd-Hf-Rh ternary systems. A key difference for these systems, with respect to the first case, is that the energy data used for the training step is directly taken from AFLOWlib and is not recomputed. The models developed are, therefore, made only on recycled data. While a decrease in performance

is observed, it should be noted that the energy scales in question are much larger than for Ag-Au-Cu and the ternaries modelled are inherently more complex. It is assessed that sufficient accuracy is reached to validate the approach described to build ternary screening tools.

In a final section, several error metrics are tested to identify compounds for which the SNAP predictions are erroneous. It is established that the more intuitive minimum distance to training database metric, $d_{\min}$, is not suitable. The two other metrics tested, the extrapolation grade, γ, and the ensemble standard deviation, σ, are however very effective and display high correlation with the model error. Since the latter is readily available upon construction of the model, it is chosen for the next study.

Two crucial insights are unravelled in this chapter. The first is that the chemical environments encountered in binary alloys are similar to those in ternaries, since a MLIAP trained on binaries is effective at predicting the energies of ternaries. Secondly, the screening tools devised can only perform well for compounds close to their equilibrium. In the context of accelerating the search for stable materials, it is important to combine a database of candidate compounds with an appropriate screening tool. The two ideas highlighted above hint that using the structures of binary compounds, close to the convex hull as prototypes for ternary compounds, may enable the use of the ensemble SNAP model to surf the convex hull. In the next chapter, the model is deployed to search for novel stable phases by scanning the convex hull.

5

Ternary Convex Hull Construction with MLIAPs

5.1 Introduction

The work presented in this chapter was done in full collaboration with Michail Minotakis. Matteo Cobelli provided the code to train the machine learning interatomic potentials and was involved in the project development. This chapter is related to work described in an article soon to be published (pre-print [32]). As such, this introduction is closely related to the corresponding section of the article. Since there is significant overlap between the relevant literature for this chapter and the previous chapter, a full overview is not given here. Instead, only the most pertinent concepts are reminded before elaborating further on studies that are directly comparable to this one.

The first step in high-throughput computational studies consists in identifying stable compounds by finding a stoichiometry and an associated structure that can be formed. In order to assess the stability of a given structure, the appropriate convex hull diagram needs to be calculated. The proximity, δ [Equation (4.9)] between a compound's enthalpy of formation, H_f, and the closest tie line on the convex hull serves as a criterion for evaluating its stability. Lower values indicate a higher likelihood of stability. Threshold values, typically up to $\sim$100 meV/atom, are used as stability cut-offs [203]. One possibility is to predict this quantity directly by using ML models, where compound compositional information is encoded and mapped onto H_f. This is

otherwise known as composition prediction, as it is used to identify which stoichiometries are stable, by fixing structural variations. Models where the feature vector is only based on compositional information have been used to predict the stability of compounds forming a set of prototype structures (elpasolites, perovskites, Heuslers, etc.), which is fixed for the compounds in the training set [312, 25, 230, 203]. Including structural information mainly improves the predictions, if large training datasets ($>10^5$ data points) are used [313]. Graph convolutional neural networks [157, 158, 159, 162] have notably been used to predict convex hull distances accurately, and benefit greatly from structural features [160, 161]. Note that these can also be constructed with compositional information only [314, 315]. One downside to the inclusion of structural information in the models is that the optimised structure is not known prior to the search, so that data for unrelaxed structures has to be used. This can notably be corrected by using machine learning interatomic potentials (MLIAPs), that are capable of performing relaxation.

Such potentials have been successfully applied to predict the energy and forces of alloys [127], and have been used to accelerate and assist the construction or further exploration of binary and ternary convex hulls. Workflows built on these potentials use MLIAPs as surrogate models to first relax and then make energy predictions on a large library of prototype structures. The lowest energy structures are then compared to a reference convex hull obtained from DFT calculations. This process allows one to improve the reference convex hull diagram by identifying structures lying below it. The training of such potentials is crucial for adequate performance, and studies insist on using high-energy structures for the relaxations to be reliable. Work in this area has broadly been split into two categories. In the first, specific MLIAPs are trained for a given system [175, 213, 316, 317, 318, 319, 320], typically using active learning. In the other, MLIAPs are trained on large generic databases, and are used to scan over many phases [163, 321]. The former is more accurate than the latter, but it is not transferable to other phases. Due to their higher accuracy, phase-specific MLIAPs can also be regarded as global structure optimisers, in that they can be used to identify specific stable compositions, and can accurately predict their structure as well. Many other ML global structure optimisers, exist, either in the form of novel workflows [171, 173, 322, 323] or by inserting MLIAPs into the pre-existing state-of-the-art global

structure optimisers [172, 170, 324].

This chapter describes two bodies of work within this area. In a first part, it is demonstrated how the MLIAP model presented in Chapter 4, trained on data available on AFLOWlib [19], is used to screen a library of ternary alloy prototypes constructed from their associated binary systems. The ensemble of SNAP [28] models trained on energy data for the three binary systems associated with a ternary one was able to predict the energies of ternary compounds with a mean absolute error (MAE) of $\sim 10^{-2}$ eV/atom, as long as the structures are fully relaxed. This, not only does provide a fast energy-screening tool for ternary compounds, which only requires *ab-initio* data on binary structures, but it also gives the valuable insight that chemical environments within binary and ternary transition-metal alloys are similar. This is the idea at the heart of the workflow introduced here. A selection of binary structures, those close to their respective convex hulls, are selected as prototypes for ternary alloys. In a high-throughput setup, these are screened using an ensemble of SNAP models, trained on binaries. The lowest-enthalpy compounds are then selected as the most promising candidates, and their energies are calculated using high-fidelity DFT. The ternary convex hull is thus updated.

What makes this workflow different from tailor-made MLIAPs used for convex hull construction is that all the data, both for the prototype generation and for training the SNAPs, are taken from the relevant binary phases of the AFLOWlib database [235]. In other words, there is no need to generate any new data. Despite the training database not being specifically made, either by including important configurations through physical intuition or through active learning, it still has a low enough error on energy predictions to enable a high-throughput search of novel alloys. This is because stable binary and ternary phases share similar local atomic environments. In some sense, the workflow enables an interpolation of the data already available in AFLOWlib, to scan ternary convex hulls and identify stable compositions. Since only a few high-energy structures and no out-of-equilibrium configurations are included in the SNAP training dataset, additional features are introduced in the workflow to increase the robustness of the predictions. These include constraints on the SNAP-driven relaxations (constant volume and the inclusion of a maximum number of steps) as well as using an ensemble of models.

In a second part of this chapter, a similar workflow is built, but with a change in the screening tool. Instead of using the SNAP ensemble model and making specific adjustments to use SNAP as a relaxation driver, the recently developed universal force-field M3GNet [163], a crystal graph neural network, is inserted into the workflow. This state-of-the art universal graph deep-learning interatomic potential was pre-trained on the data from the Materials Project [20], which most notably includes in excess of 185,000 structures resulting from the *ab-initio* relaxation of roughly 62,000 compounds. Across all compounds in the dataset, 89 different elements are represented. Energies, forces, and stresses were included in the training. The model MAE was only of 35 meV/atom on the test set (5% split of the full dataset), which is of the right order of magnitude to serve as a screening tool. The use of the model is in line with the spirit of the workflow. While it is trained on a far wider range of compounds than the SNAP ensemble model, it is pre-trained on a readily available dataset and requires no extra DFT calculations. One major advantage of M3GNet is that it is capable of performing ionic relaxation without any intervention. It can therefore be readily integrated into the workflow developed to screen novel ternary materials.

This chapter is organised in three main parts. The methods and relevant phase diagrams are first presented. In Section 5.2, attention is given to the method used to generate derivative structures for the creation of ternary prototypes. The details of the M3GNet model are also introduced. In Section 5.3, three ternary phase diagrams are described, those of Mo-Ta-W, Al-Fe-Ni and Bi-Fe-X (X being a refractory metal). These systems serve as tests for the workflow developed. In a second part, in Sections 5.4 and 5.5, the workflow development is explained, and the SNAP ensemble model is used as a screening tool. Finally, in a third part, the workflow constructed around M3GNet is described, and the associated results are shown.

5.2 Methods

Since the work revolves around the identification of new ternary intermetallics, the methods presented in the previous chapters are also relevant here. The SNAP interatomic potential and the construction of convex hulls are notable examples. Nevertheless, two new computational tools are central for the development of the workflow and

deserve some attention. The first is a methodology for efficiently enumerating "derivative structures" of a parent structure, presented in Reference [325]. This method provides an algorithm for identifying *all unique* atomic decorations and corresponding supercells that can be generated from a prototype structure. Since the workflow presented makes use of an initially small library of prototypes, the algorithm provides an approach to efficiently extend the number of candidate structures to be screened. In the second subsection, a presentation of the crystal graph neural network with 3-body interactions (M3GNet) [163] is given. This recently developed universal interatomic potential leverages the massive database of the Materials Project [20] to be an effective screening tool and to perform relaxation on materials containing up to 89 different species. The capacity of the deep learning model to do the latter motivates its use as a substitute for SNAP in the workflow, as presented in Section 5.6.

5.2.1 Enumerating Derivative Structures

In this subsection, the algorithm in Reference [325] is presented. Some relevant elements of group theory are first presented to aid the core discussion. The following subsections break down the steps followed by the algorithm. Since this algorithm is completely independent of the other methods so far presented, the mathematical notation used for this specific subsection bears no relation to any other sections.

Elements of Group Theory

There are first some elements and definitions from group theory that need to be explained; firstly, a formal definition of a group. A group is a set G together with a binary operation (operation on two elements), also called a multiplication, such that

$$* : G * G \rightarrow G \tag{5.1}$$

In other words, when the operation is performed on two elements of G, the result is also an element of G. The following conditions must also be satisfied:

- The group multiplication must be associative.
- The set of the group must have an identity element.

- Each element of the group must have an inverse.

An example of this is ($\mathbb{Z}$,+), the group is the set of all integers under the addition operation. If one takes any two elements of the set and adds them together, the result will also be part of the set. Associativity is respected for the addition of integers. The identity element is 0. Each element has an inverse, for an integer i it is just -i.

An important notion for the discussion to come is that of equivalent labellings. This concept is valid for finite groups. As an example, the cyclic group $\mathbb{Z}_4$ will be used. This is the set of integers $\{0, 1, 2, 3\}$ under the addition operation modulo 4. The Cayley table for this group is given in Table 5.1. This is a square table with all elements of the group as headers of each column and row, with the inside of the table populated by the group multiplication of each element, in the same form as a multiplication table.

Table 5.1: Cayley table of the cyclic group $\mathbb{Z}_4$.

$\mathbb{Z}_4$	**0**	**1**	**2**	**3**
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

In each row and column, all elements of the group are present once but each time in a different order. These correspond to different permutations of the group, which are equivalent to each other and correspond to a change in the order of the group members. Each one is an equivalent labelling of the same group. In other words, writing the different elements of the group in the order of the first, second, third or fourth row is equivalent. Writing $\mathbb{Z}_4$ as (0,1,3,2) however would not be.

Algorithm

Reference [325] describes an algorithm for generating "derivative structures" from a parent structure. This is exactly what is required for the work in this thesis, since there is a pool of prototype structures that provide parent lattices, from which one wants to generate all possible supercells up to a certain size (in number of atoms), with various lattice decorations with three species. Note that in this first piece of

work published by Hart *et al.*, the algorithm described is used to generate all possible and unique superstructures of a parent structure for a given set of k species up to order n (i.e. with n atoms). This implies that the structures generated will have different stoichiometries, and include all possible k-nary structures (binary, ternary, quaternary), bar unaries. Additionally, in the original paper, the parent structure is a lattice (crystal structure with one atom per unit cell). In further work, the authors extend their algorithm to include parent multilattices (crystal structures with cells containing more than one atom) [326], as well as a method for generating structures at a fixed concentration of the k species [287]. These are not presented here, only the first seminal work that describes the initial method is given.

An example of a derivative superstructure is shown in Figure 5.1. The *fcc* lattice is the parent lattice, depicted by two consecutive conventional unit cells. The structure on the right is a binary superstructure and an example of a derivative structure obtained from the *fcc* lattice. It can be seen that different sites from the *fcc* lattice are decorated with two different types of atoms. Decorations (the different ways to place atoms on the various superlattice positions) will also be referred to as colourings or labellings.

The algorithm can be broken down into two main steps. The first involves generating all the unique superlattices (without decorations) of a parent lattice. In this context, a superlattice has the same unit cell as a *sublattice* of the original lattice, in the mathematical sense. However, the points of the *parent* lattice "inside" this supercell are also part of the superlattice and can be decorated. The second concerns the generation of all the unique decorations of these superlattices. This is performed for all lattices with sizes from 1 to n, where n indexes the size of the superlattices and, for the case of single-site lattices, also corresponds to the number of atoms of the derivative structure. The method is fast as it scales as $O(N)$, rather than the $O(N^2)$ that one would obtain by first generating all possible derivative structures in a brute force approach and then comparing them to each other to find the unique ones.

Generating Unique Superlattices

Consider a parent lattice whose lattice vectors are arranged in the columns of a 3×3 matrix **A**. In order, to construct a superlattice or to perform a change in basis and obtain a new basis **B**, the transformation **B**=**HA** is performed, where **H** is a matrix

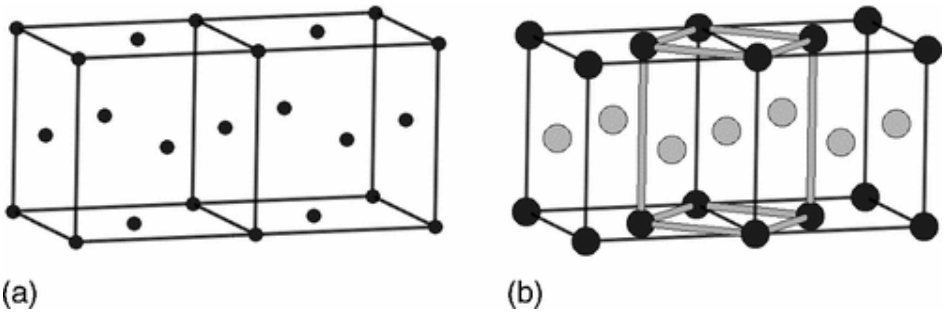


Figure 5.1: Illustration of a binary superlattice (right) of a parent *fcc* lattice (left). Image taken from Reference [325]

with integer entries. The determinant of this transformation matrix corresponds to the index n of the superlattice. All such transformation matrices can be written in Hermite Normal Form (HNF), as

$$\mathbf{H} = \begin{pmatrix} a & 0 & 0 \\ b & c & 0 \\ d & e & f \end{pmatrix}, \quad 0 \leq b < c, \quad 0 \leq d, e < f. \tag{5.2}$$

Note that $|\mathbf{H}|{=}n{=}a \times c \times f$. Transformation matrices written in this form are unique, so they each yield a different superlattice. Indeed, for the case of transformations of order $n{=}1$, all $\mathbf{H}$ written in HNF become the identity matrix ($\mathbf{H}{=}\mathbf{I}$), since all such transformations just correspond to a basis transformation and therefore still produce the same lattices. The fact that transformation matrices written in HNF represent distinct superlattices and have a well-defined form, implies that superlattices of all orders can be easily and systematically created through the use of a few nested loops. Importantly, this step does not require *any* knowledge of the parent lattice, so there is the same number of HNFs at each order for all lattices, as given in Table 5.2.

Table 5.2: Number of distinct HNFs and SNFs for different index numbers n.

n	1	2	3	4	5	6	7
# HNFs	7	13	35	31	91	57	155
# SNFs	1	1	2	1	1	1	3

The set of HNFs constitute the set of unique and distinct undecorated superlattices

provided the parent lattice has no rotational symmetries. If the latter does not hold, then some HNFs will produce equivalent superlattices. In the process of enumerating the HNFs, a symmetry check is therefore performed each time a new HNF is created. The symmetry of the lattice is determined, and the associated rotation matrices are listed as $\mathbf{R}$. If $\mathbf{B}_i$ is a superlattice in the set of already enumerated superlattices and $\mathbf{B}_j$ is a new one, then $\mathbf{B}_i^{-1}\mathbf{R}\mathbf{B}_j$ must be a unimodular matrix of integers for the new superlattice to be added to the list, for all rotations $\mathbf{R}$ and all superlattices $\mathbf{B}_i$. The result of this process is a set of unique superlattices up to a defined order n that need to be decorated.

Generating Unique Decorations

For each superlattice (HNF) created, there are n interior sites (the ones of the parent lattice within the unit cell of the superlattice) to be decorated with the possibility of k different types of decorations (different atoms), which will be labelled with letters (a,b,c,...). This means there are up to k^n possible decorations, which could be equivalent because of the symmetry of the underlying lattice, as well as the translational symmetry of each superlattice. These decorations can be represented by all the n-digit, base-k numbers. The representation of the 16 such decorations are given in Table 5.3 for n=4 sites and k=2 species a and b. Note that this list of decorations is valid for all HNFs of index n. The rest of the algorithm then deals with eliminating the duplicate decorations from this list by making use of group theory, rather than comparing all structures one to one directly. Each of the steps in the elimination process are described below and after each one, duplicates are removed from the list before proceeding to the next one.

Table 5.3: Hash table for the different decorations of a superlattice with n=4 sites and k=2 possible different species, a and b, that can be used as decorations. Some of them may be equivalent.

aaaa	abaa	baaa	bbaa
aaab	abab	baab	bbab
aaba	abba	baba	bbba
aabb	abbb	babb	bbbb

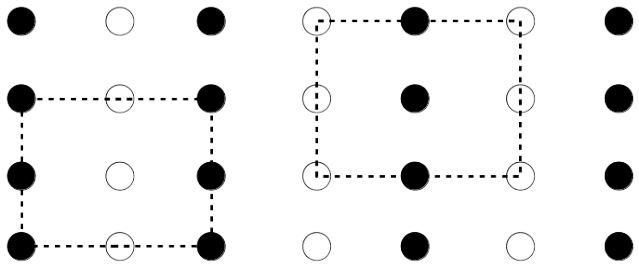


Figure 5.2: Illustration of translational duplicacy. A 2D structure is shown above with a cubic parent lattice and decorated with two atom types a (black) and b (white). Two unit cells of superstructures are outlined by dashed lines. The first one corresponds to a *aabb* decoration, while the second one to a *bbaa* decoration. Both yield the same structure and only differ by a translation of the parent lattice.

Eliminating Translational Duplicates

Translational duplicates correspond to derivative structures generated from different atomic decorations of a superlattice, but that result in the same structures under a translation of the parent lattice. This is illustrated for a two-dimensional lattice in Figure 5.2. In order to identify these duplicates, a group theoretical approach is followed. An example is presented to help give some intuition on the approach. A 2D simple cubic lattice and an associated 2×2 superlattice, as shown in Figure 5.3, will be used for this purpose. The parent lattice L is a group composed of the infinite set of points spanning $\mathbb{R}^2$ under the operation of translations between these points. A given sublattice (the *mathematical* sublattice) of the parent lattice corresponds to a subgroup S of L. By translating the sublattice by *lattice* translations, without reconstructing the starting sublattice, equivalent sublattices can be constructed. This idea is illustrated in Figure 5.3. Note that the size of the set of such sublattices is finite and is equal to the index n of the superstructure associated with the sublattice. In group theory nomenclature, these sublattices are cosets of S. Together, they form a group called the quotient group $G = L/S$ (different to the general notation used for a group above), which is a finite group with n elements. In the example in Figure 5.3, there are four

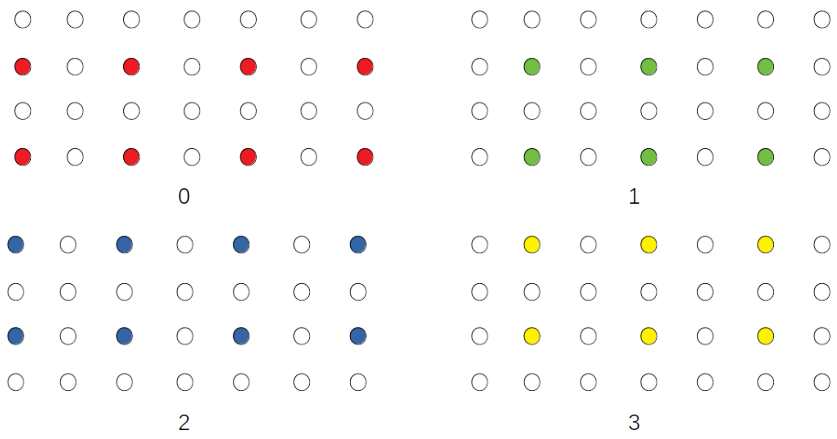


Figure 5.3: Illustration of the concept of the cosets of a subgroup S. The white dots represent the parent lattice. Four equivalent sublattices of order n=4 are shown (labelled 0,1,2,3), and are decorated with different colours. They differ only by a translation of the parent lattice, so correspond to a different choice of the origin. Their union is the parent lattice, decorated with the respective colours.

elements, which could each be labelled, such that the group G is labelled as (0,1,2,3). Note that this could also be viewed as decorating the different 4 sites of the superlattice with different atomic species, as illustrated by the different colours.

Table 5.4: Multiplication table of $\mathbb{Z}_2 \oplus \mathbb{Z}_2$.

$\mathbb{Z}_2$	(0,0)	(1,0)	(0,1)	(1,1)
(0,0)	(0,0)	(1,0)	(0,1)	(1,1)
(1,0)	(1,0)	(0,0)	(1,1)	(0,1)
(0,1)	(0,1)	(1,1)	(0,0)	(1,0)
(1,1)	(1,1)	(0,1)	(1,0)	(0,0)

If one were to shift all the sublattices by one "right translation of the parent lattice", the corresponding superlattice would still be the same, however, the labelling of the group G would change. In the illustration from Figure 5.3, sublattice 1↔2 and 3↔4, G would then be labelled as (1,0,3,2). However, this still corresponds to the same superlattice, there has just been a shift in origin. This equivalence between labellings [here (0,1,2,3) and (1,0,3,2)] can be rationalised through group theory. For this, the Smith Normal Form (SNF) matrix of the HNF of the lattice considered must be used.

Note, however, that many HNFs share the same SNF, leading to a low number of SNFs for each index n, as shown for the first values of n in Table 5.2. The SNF is a diagonal matrix. In the case it is a 3×3 matrix, as for a 3D lattice, the entries on the diagonal are denoted s_i, s_j and s_k. It turns out that the quotient group G described thus far corresponds to the direct sum of the cyclic groups $\mathbb{Z}_{s_i} \oplus \mathbb{Z}_{s_j} \oplus \mathbb{Z}_{s_k}$, where $\oplus$ is a direct sum. As a result, equivalent labellings correspond to the different rows or columns of a multiplication table of this group. In our example, G would be $\mathbb{Z}_2 \oplus \mathbb{Z}_2$. The multiplication table of this group is given in Table 5.4. In comparison to our example, by changing the labels such that $0 \to (0,0)$, $1 \to (1,0)$, $2 \to (0,1)$, $3 \to (1,1)$, then the first two rows show that (0,1,2,3) and (1,0,3,2) are equivalent and therefore correspond to the same superlattices. This approach allows us to identify equivalent labellings and hence eliminate translation duplicates.

To illustrate this process, consider all the different possible decorations for our binary system, given in Table 5.3. If these were used to decorate the 2D simple cubic lattice used as an example in this section, it can be seen how many of these are in fact translational duplicates of one another. This is illustrated in Table 5.5. Of the 14 possible decorations (excluding pure a or b), this shows there are in fact only 5 translationally distinct superlattices.

Table 5.5: Different decorations of the superlattice related to equivalent labellings.

(0,1,2,3)	(1,0,3,2)	(2,3,1,0)	(3,2,0,1)
aaab	aaba	abaa	baaa
aabb	aabb	bbaa	bbaa
abab	baba	abab	baba
abba	baab	baab	abba
abbb	babb	bbab	bbba

Since the number of SNFs is low for each n, compared to the number of HNFs, this implies that many redundant superlattices can be sorted through simultaneously. Overall, this step reduces by a factor of $\sim n$ the number of superlattices. This group theory approach alleviates the need to perform one-to-one comparisons of all such superlattices through lattice translations.

"Label-exchange" Duplicates

In the first piece of work published, G. Hart *et al.* [325] indicate that since the aim of the algorithm is to create unique structural prototypes, rather than unique materials, "label-exchange" superlattices are considered to be duplicates and are hence removed. In the binary case discussed previously, the labellings abbb and baaa would be considered to be such duplicates, since the underlying structures and decoration of sites is the same for each material, but the type of labellings are different (a exchanged for b). These are not relevant to this study, as the algorithm is applied at a fixed stoichiometry.

Superperiodic Labellings

The next set of equivalent labellings that are removed concern the superperiodic ones. These are labellings of a sublattice that make the period of the repeating unit shorter because of the specific ordering of decorations. Figure 5.2, in fact, depicts a superperiodic structure, as a smaller unit cell could be chosen to describe this superlattice (one consisting of only two sites rather than four). Such labellings correspond to ones that do not change upon certain permutations (parent lattice translation). This is the case for the ones on lines 2, 3 and 4 in Table 5.5, including the one from Figure 5.2 (the exact one would depend on the choice of site indexing).

"Label-Rotation" Duplicates

The last type of duplicates to consider are the equivalent superlattices that differ by a lattice rotation of the parent lattice. This depends on the symmetry of the parent lattice, so its associated rotation matrices are used. The left transformation matrix $\mathbf{L}$, used to construct the SNF from the HNF ($\mathbf{S} = \mathbf{LHR}$) is also required, as are the parent lattice vectors $\mathbf{A}$. The n members of the quotient group are represented with three components and together form the $3\times n$ matrix $\mathbf{G}$. These are transformed under a rotation to give new labellings $\mathbf{G}'$ as,

$$\mathbf{G}' = \mathbf{L}\mathbf{A}^{-1}\mathbf{R}\left(\mathbf{L}\mathbf{A}^{-1}\right)^{-1}\mathbf{G}. \tag{5.3}$$

As these checks are performed within the quotient group, duplicates labellings are eliminated in a time proportional to the number of labellings in the list.

Summary

The algorithm presented in this section is based on a group theoretical approach to generate all unique generative structures of a parent structure. Unique unit cells up to a pre-defined size are identified and generated through the HNF notation. All possible decorations of the superlattices are carefully labelled and group theory provides tools to eliminate the different duplicates mentioned, namely translational, "label-exchange", superperiodic and "label-rotation" ones.

From an initial structure, a maximum cell size and a number of species, a list of unique derivative structures is provided. An open-source FORTRAN code, ENUMLIB, is available from the authors of the work and is used here. Note that extensions of the methodology that allow the use of multi-site prototypes [326] and a fixed stoichiometry [287] generation are implemented in the computer program and are used for this work.

5.2.2 M3GNet

In the second phase of this work, the screening tool M3GNet (from graph neural networks with 3-body interactions) [163] is inserted into the workflow. This is a pre-trained machine-learning model that relies on pre-existing DFT data contained in the Materials Project [20] to be trained. It is a universal interatomic potential, since it can be used on combinations of 89 different atomic species. It was introduced to improve both on typical message-passing and graph neural networks [152, 157, 153, 327, 158] and conventional MLIAPs [28, 126, 112, 140, 148, 125] for high-throughput materials screening. The former can be used to make accurate property predictions [328] but can not perform atomic relaxations, while the latter is limited to the number of different chemical species present in a material, due to the explosion of the number of N-body interactions with increasing number of species and, hence, the number of basis functions required. M3GNet is capable of performing accurate relaxation and, due to the nature of crystal graph neural networks, is not limited by any number of species. It is also freely available and ready to use, as it interfaces with the PYTHON Materials Genomics (PYMATGEN) library [329]. In this subsection, a description of the

architecture of the M3GNet model is given.

MEGNet

The model relies on the MatErials Graph Network (MEGNet), introduced by the same authors in 2019 [328]. This machine-learning model differs significantly from the MLIAPs described previously. The latter rely on local atomic fingerprints combined with a machine-learning model and are used to predict energies, forces, and stress tensors. The representation of MEGNet is more global and encodes information of the full system, rather than taking the Behler-Parinello [112] approach of describing the system by a combination of local atomic descriptors.

MEGNet is a type of graph network [330], a deep-learning architecture. A system (molecule or crystal) of N atoms is described in terms of a mathematical graph, in which atoms are represented by nodes in the graph and bonds between neighbouring atoms are represented by edges. Each node i and edge k have an associated attribute vector, denoted $\vec{v}_i$, the atomic attribute, and $\vec{e}_k$, the bond attribute. $V = \{\vec{v}_i\}$ constitutes the set of all N atomic attributes, while $E = \{(\vec{e}_k, r_k, s_k)\}$ is the set of all edge attributes $\vec{e}_k$ with r_k and s_k the indices of the atoms delimiting each bond k. There is one last attribute $\vec{u}$ that makes up the graph. This is a global attribute of the full system that contains state attributes, such as temperature. The graph, defined as $G = (E, V, \vec{u})$, represents the system. Crucially, it is this representation that is updated during the "learning" process of the ML model. A visual aid to illustrate the graph is given in Figure 5.4.

A graph network (GN) framework makes use of "update" functions for relational reasoning over graphs [330]. The central unit in such a framework is the GN block, which maps a graph G to another graph G'. This is done by performing successive updates of the bond attributes, then of the atom attributes, and finally of the global state vector. These are respectively performed with the update functions ϕ_e, ϕ_v and ϕ_u. The bond attribute vectors $\vec{e}_k$ are updated from themselves, their connecting atoms with indices s_k and r_k and the global state vector $\vec{u}$, as,

$$\vec{e}_k{}' = \phi_e \left(\vec{v}_{s_k} \oplus \vec{v}_{r_k} \oplus \vec{e}_k \oplus \vec{u} \right), \tag{5.4}$$

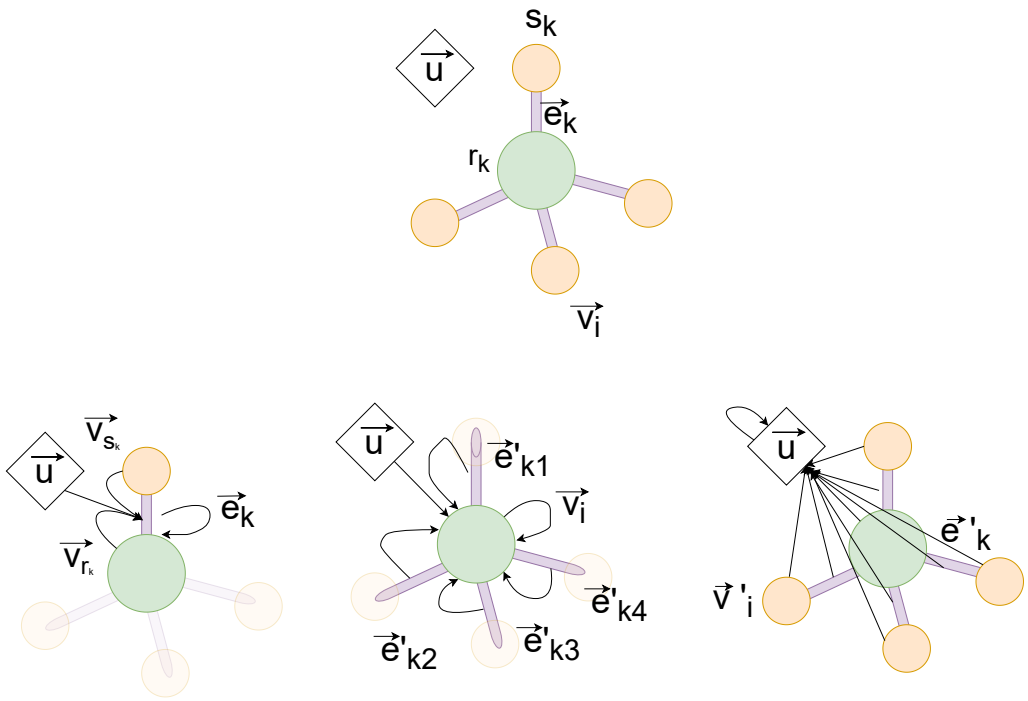


Figure 5.4: Illustration of the crystal graph and update steps in the MEGNet model. The top image shows a molecule as a graph, with atoms represented by circular nodes and edges by cylinders. $\vec{v}_i$ is the atom attribute for node i, $\vec{e}_k$ the bond attribute for edge k, and $\vec{u}$ is the global attribute. Below, the successive update steps are illustrated. First, for each edge, the new edge attribute $\vec{e}_k{}'$ is computed (left). The atom attribute $\vec{v}_i{}'$ is then calculated for each atom (centre). Finally, the new global feature $\vec{u}'$ is calculated (right) from all bond and atom attributes.

where $\oplus$ is the concatenation operator and ϕ_e is the update function for edges. The node attributes $\vec{v}_i$ are updated from themselves and their connecting bonds, along with the global state vector $\vec{u}$, as follows

$$\vec{v}_i^{\,e} = \frac{1}{N_i^e} \sum_{k=1}^{N_i^e} \{\vec{e}_k{}'\}_{r_k=i} \,, \tag{5.5a}$$

$$\vec{v}_i{}' = \phi_v \left(\vec{v}_i^{\,e} \oplus \vec{v}_i \oplus \vec{u}\right) . \tag{5.5b}$$

Here, ϕ_v is the update function for atoms and N_i^e is the number of bonds connected to i. The first sum is over all bonds surrounding atom i. Thus, $\vec{v}_i^{\,e}$ is the average edge vector from all neighbouring bonds of i. These first two steps enable the communication of

edge and atom attributes locally. Finally, the global state attribute vector $\vec{u}$ is updated according to the following equations

$$\vec{u}^{\,e} = \frac{1}{N^e} \sum_{k=1}^{N^e} \{\vec{e}_k{}'\}\,, \tag{5.6a}$$

$$\vec{u}^{\,v} = \frac{1}{N} \sum_{k=1}^{N} \{\vec{v}_i{}'\}\,, \tag{5.6b}$$

$$\vec{u}' = \phi_u \left(\vec{u}^{\,e} \oplus \vec{u}^{\,v} \oplus \vec{u}\right). \tag{5.6c}$$

ϕ_u is the update function for the global state vector and N^e is the total number of bonds in the system. The global state attribute can thus be used to communicate information on the scale of the whole system. The equations underlie one update of the graph and are associated with the MEGNet layer. Several such update rounds are incorporated in the full model. The model is a deep-learning one since the three update functions are neural networks with two hidden layers. For each one, there are three sets of weights and biases that are learned. The modified Softplus function is used as a nonlinear activation function [85, 331]. Two fully connected layers are added before the GN unit described to increase model accuracy. Note that a skip connection is also included to avoid over-fitting. These steps make up one MEGNet Block. The MEGNet block is illustrated in Figure 5.5.

Such a block is repeated N_{blocks} times to increase the expressiveness of the model and the connectivity between bonds and edges. For the full crystal MEGNet model, 3 blocks are used. Several other blocks are added to the full architecture. At the start, an embedding layer is added for V, as will be described. Following the MEGNet blocks, set2set models [332] are used to embed the set of all attribute vectors in E and V into one vector each. The two resulting vectors of this readout set are concatenated along with $\vec{u}$ into one vector, that is fed into a final neural network with two hidden layers. The output is a single floating number. The overall architecture is illustrated in Figure 5.5.

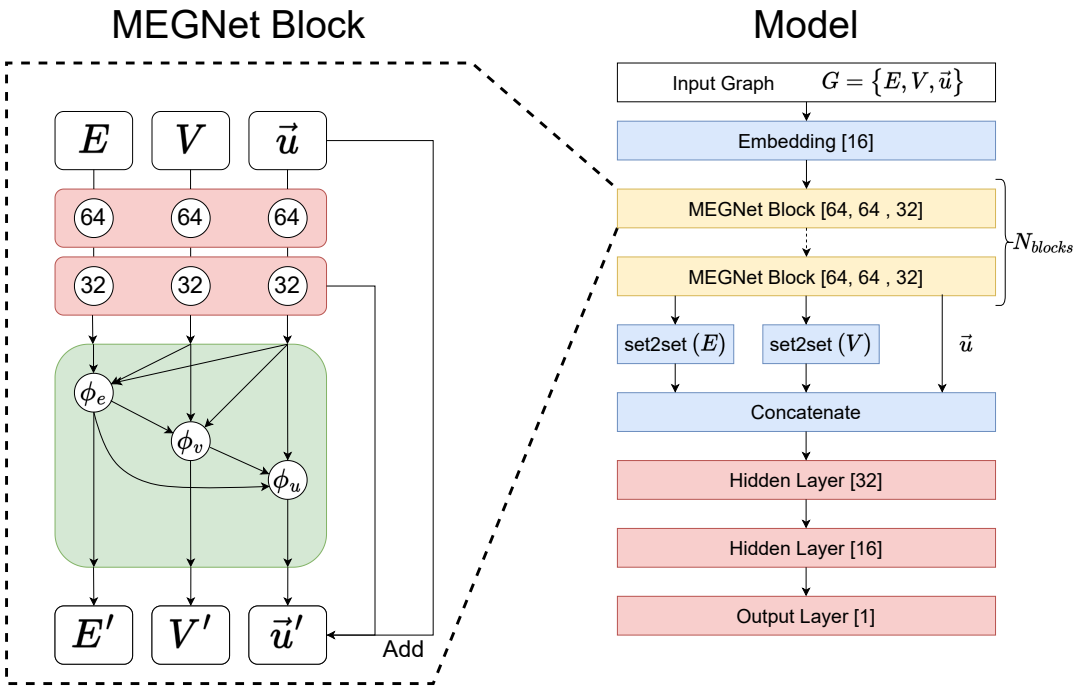


Figure 5.5: Architecture of the MEGNet model. The green block is the MEGNet layer. The yellow blocks are MEGNet blocks, repeated N_{blocks} times. The blue blocks are for embedding or concatenating, and the red ones are fully connected neural network layers. Input blocks are in white. The add arrows are skip connections. The square brackets indicate the number of nodes in each neural network layer. The schematic is based on that in Reference [328].

Vector Attributes

Some attention is now given to the content of the three vector attributes that make up the graph sets E, V and the global state vector $\vec{u}$. The initial atomic attribute for each atom, in the case where the system is a crystal structure, is a single number, the atomic number of the chemical species. This is embedded in a 16-feature vector in the first step before the MEGNet block. The process of embedding consists of turning a discrete variable (integer, word, etc.) into a vector of continuous numbers. The weights used to perform the elemental embeddings are learned. The bond attribute vector $\vec{e}_k$ of bond k, sample 100 points of a Gaussian distribution between 0 and 6 Å, centred on the edge's bond length d_k and with fixed width (0.5 Å), provided the bond length is less than a cut-off value R_{cut}. In the first MEGNet model, the global state vector $\vec{u}$ is used as a placeholder for global information exchange and

is a vector with two zero entries. This model has proved to beat state-of-the art algorithms such as SchNet [152], message-passing neural network enn-s2s [151], crystal graph convolutional neural networks [157], automatminer based on matminer [333] and random-forest-based models for the predictions on both molecular and crystal structure properties [328, 334]. The elemental embedding equally proved to be transferable for different properties, indicating that chemical identity had been learned through the embeddings. Through the exploitation of the global state, it has equally been used effectively to incorporate multi-fidelity data, which notably increased the predictions of high-fidelity data [335].

While MEGNet provides an ML architecture for training a model capable of making predictions of system properties, it is not an MLIAP. When making predictions on novel materials, the structures of the candidate compounds must be estimated, and they are unlikely to be in their equilibrium configuration. To drive them there, *ab-initio* relaxation is required. If such calculations were to be performed on all candidate structures, the gain from the ML model used to make the final property predictions would be dwarfed by the time taken to perform the DFT relaxation. To circumvent this bottleneck and increase the applicability of MEGNet-type models, M3GNet was developed.

M3GNet: Architecture

M3GNet builds upon MEGNet but includes additional features such that it can be used as an MLIAP, not just an ML property predictor. Here, the architecture of M3GNet is summarised and the differences with MEGNet are highlighted. The training and applications of M3GNet are presented in the following section.

The graph representing the system now includes two new features. A set X that includes the coordinates of all atoms is added, along with a 3×3 lattice matrix $\mathbf{M}$. The graph G is thus the set $(E, V, X, \vec{u}, \mathbf{M})$. In practice, the two additions are used to compute the forces and stress tensors at the final stage via auto-differentiation, and they are not used in the main deep-learning architecture. Figure 5.6 shows an illustration of the M3GNet architecture. A first difference with the previous model is that two pre-processing units, the graph featuriser, and the many-body computation unit, are present. The former creates an embedding for both the atomic and bond

attributes. The atomic embedding is identical to that in MEGNet, bar the fact that the final feature length is 64. The bond attributes are now required to be invariant upon translation and rotation of the system, as well as upon permutation of identical atoms for the model to be used as a MLIAP. The bond attributes are now denoted $\vec{e}_{ij}$, where, i and j label the atoms surrounding the bond. The initial bond attributes, $\vec{e}_{ij}^{0}$ are expanded in terms of the first three radial basis functions defined in Reference [336] for the bond length R_{ij}. The embedded attributes are then fed into the main message-passing block. In parallel, many-body terms, up to 3-body in the case of M3GNet, are computed. These correspond to the angles between bonds $\vec{e}_{ij}$ and $\vec{e}_{ik}$, denoted θ_{jik}.

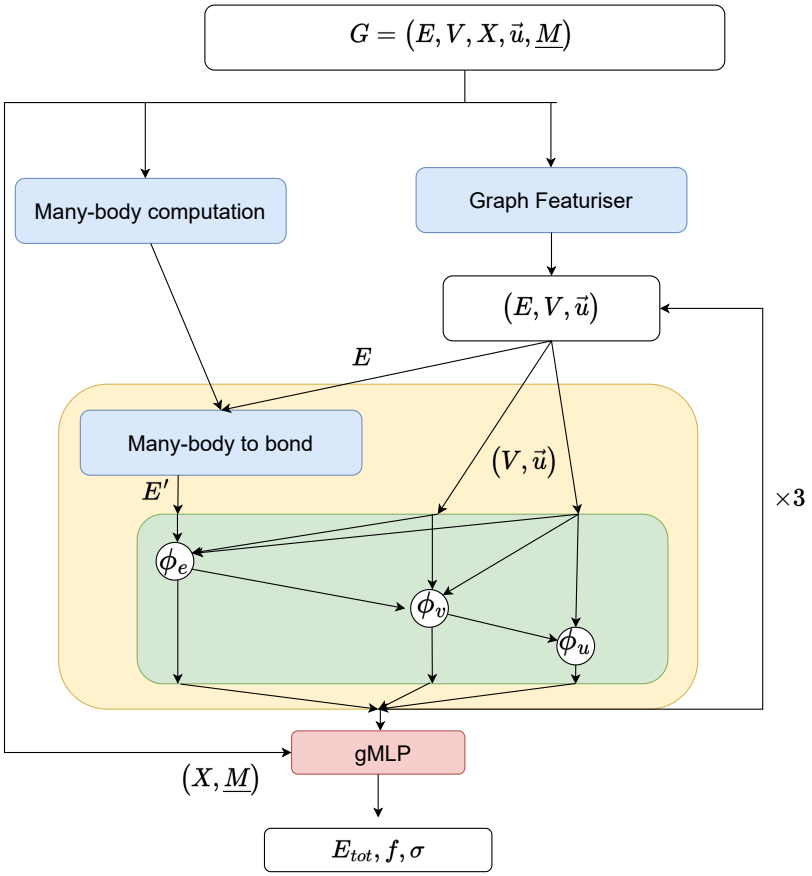


Figure 5.6: Architecture of the M3GNet model. The schematic is based on that in Reference [163].

In the main block, there are two main steps. The first is the many-bond to bond unit, which enriches E with 3-body terms. The second is the message-passing step, which plays the same role as the MEGNet unit in this architecture. This main block is repeated 3 times, as for MEGNet, to increase the expressivity of the model. A first new, important aspect of this model is the inclusion of gated neural networks instead of con-

ventional ones [337]. Gated neural networks, also called gated multi-layer perceptrons (gMLP), have a different structure than conventional neural networks. Their aim is to emulate the increased complexity of Transformer models [70] over neural networks, which are heavily used in the fields of natural language processing and computer vision [338]. A crucial innovation for these models is the use of the attention mechanism [70], which cannot be replicated if model parameters are fixed. Instead, gMLPs were suggested as an alternative to replicate the attention mechanism in a neural network with fixed weights. In practice, this takes the form of including a learnable spatial projection step, which aims to identify important interactions between the features. For M3GNet, this enables further model complexity. Most neural networks are replaced by gMLPs.

The other significant difference between M3GNet and its predecessor is the inclusion of invariant three-body terms. These are incorporated within the bond attributes E during the many-body to bond unit, shown explicitly in Figure 5.6. The mathematical grounding of this block is as follows. For a bond $\vec{e}_{ij}$, all neighbours k of atom i are considered by calculating their interatomic distance R_{ik} with i and the angle between atoms i, j, and k, θ_{jik}. This leads to a first bond update in the many-body to bond module, defined as,

$$\vec{\tilde{e}}_{i,j} = \sum_k j_l \left(z_{ln} \frac{R_{ik}}{r_c} \right) Y_l^0 \left(\theta_{jik} \right) \odot \sigma \left(\mathbf{W}_v \ \ \vec{v}_k + \vec{b}_v \right) f_c \left(r_{ij} \right) f_c \left(R_{ik} \right), \tag{5.7a}$$

$$\vec{e}_{i,j}{}' = \vec{e}_{i,j} + g \left(\mathbf{W}_2 \, \vec{\tilde{e}}_{i,j} + \vec{b}_2 \right) \odot \sigma \left(\mathbf{W}_1 \, \vec{\tilde{e}}_{i,j} + \vec{b}_1 \right). \tag{5.7b}$$

Here $\odot$ is the element-wise product, $\mathbf{W}$and $\vec{b}$ are weights, j_l are the spherical Bessel functions with roots z_{ln}, and Y_l^0 are the spherical harmonics. $\vec{\tilde{e}}_{i,j}$ are vectors with each component corresponding to expansions on different basis functions (with parameters l and n). They go up to cut-off values $l_{\max}$ and $N_{\max}$, respectively. f_c is a specifically defined cut-off function defined in Reference [339], σ is the sigmoid activation function and g is the non-linear activation function [340]. The two latter are the activation functions used for the gMLPs. These are then used along with the atom attributes and global state attribute in the message-passing block of the model. As before, these

require new update functions ϕ_e, ϕ_v and ϕ_u. The notation is kept from the previous section for simplicity, but the functions are now different. ϕ_e and ϕ_v are gMLPs while ϕ_u is a neural network with two hidden layers using the non-linear activation function [340]. The update steps follow the order of the equations below.

$$\vec{e}_{ij}{}'' = \vec{e}_{ij}{}' + \phi_e\left(\vec{v}_i \oplus \vec{v}_j \oplus \vec{e}_{ij}{}' \oplus \vec{u}\right) \mathbf{W}_e^0 \vec{e}_{ij}^0, \tag{5.8a}$$

$$\vec{v}_i{}' = \vec{v}_i + \sum_j \phi_v\left(\vec{v}_i \oplus \vec{v}_j \oplus \vec{e}_{ij}{}'' \oplus \vec{u}\right) \mathbf{W}_e^{0'} \vec{e}_{ij}^0, \tag{5.8b}$$

$$\vec{u}\,' = \phi_u\left(\vec{\bar{u}}^{\,v} \oplus \vec{u}\right). \tag{5.8c}$$

Here, $\mathbf{W}_e^0$ and $\mathbf{W}_e^{0'}$ are the weights of the model.

M3GNet: Training and Applications

From this model architecture, the universal MLIAP was trained on 187,687 structures, taken from the relaxation steps of 62,783 compounds present in the Materials Project [20]. The dataset included 187,687 energies, 16,875,138 force components and 1,689,183 stress components. In the full database, 89 different species are present, although some elements are more represented (oxygen) than others (noble gas species). An important aspect of the database was the inclusion of structures with a low interatomic distance, notably for compounds containing H, since this enabled the model to learn repulsive behaviour at such distances. The final MAE of the model was of 35 meV/atom on energies and 7.2×10^{-2} eV/Å for forces. The model was tested for its intended purpose of performing accurate relaxations. This was initially performed on the test set of more than 3,000 structures and proved to be very successful. Before relaxation, the M3GNet predicted energy error of the 80th percentile of the error distribution was of 71 meV/atom, whereas this decreased to 28 meV/atom for full MEGNet relaxation. This error was of 26 meV/atom when the relaxation was performed with DFT, proving the relaxation ability of MEGNet at a fraction of the *ab-initio* cost. The model was further applied for materials discovery on over 31 million novel structures. Of the top 1000 of these (predicted to be closest to the convex hull), 999 were found to have a DFT hull distance of less than 1 meV/atom. The DFT relaxations performed were

also accelerated by an order of magnitude in energy after the M3GNet relaxation. It is concluded that M3GNet can be used as an effective tool to perform accurate relaxation through machine learning, without resorting to DFT.

In the next section, the phase diagrams of the ternary systems studied in this chapter are presented.

5.3 Ternary Phase Diagrams

Four ternary systems are studied in this chapter. The first is the Ag-Au-Cu system, which is again used as a benchmark for the workflow development. Once developed, the workflow is applied to the Mo-Ta-W system. The magnetic systems Al-Fe-Ni and Bi-Fe-X (X = Zr,Ta) are further studied with the M3GNet workflow. Since the Ag-Au-Cu system has already been presented, the experimental phase diagrams of the other three systems are described here.

5.3.1 Mo-Ta-W

The Mo-Ta-W is descriptively the simplest system, since it displays full mixing across the whole compositional space for all three binaries and the ternaries [341, 342, 343, 344]. At lower temperatures, ordered phases are expected to form [342, 345]. All three metals possess the *bcc* structure in their elemental phase. The three elements are refractory metals with melting points of 2896 K for Mo, 3290 K for Ta and 3695 K for W [215]. They form *bcc* solid-state solutions across the full compositional range for binaries and ternaries. These refractory metals can notably form high entropy alloys [186].

This system was chosen as a test system, since four ternary compounds were present on AFLOWlib's database, more than for any other ternary transition metal alloy. These are MoTaW, Mo_2TaW, $MoTa_2W$ and $MoTaW_2$. The convex hull computed from the AFLOWlib DFT data is presented in Figure 5.7, to also highlight the stable binary phases. Note that the Mo-W edge of the convex hull is very shallow, with a minimum of only -6.1 meV/atom at the MoW stoichiometry. The Mo-Ta edge is lowest, at -194.0 meV/atom at the MoTa stoichiometry, and the Ta-W edge has a minimum of -115.0 meV/atom at the TaW_2 stoichiometry.

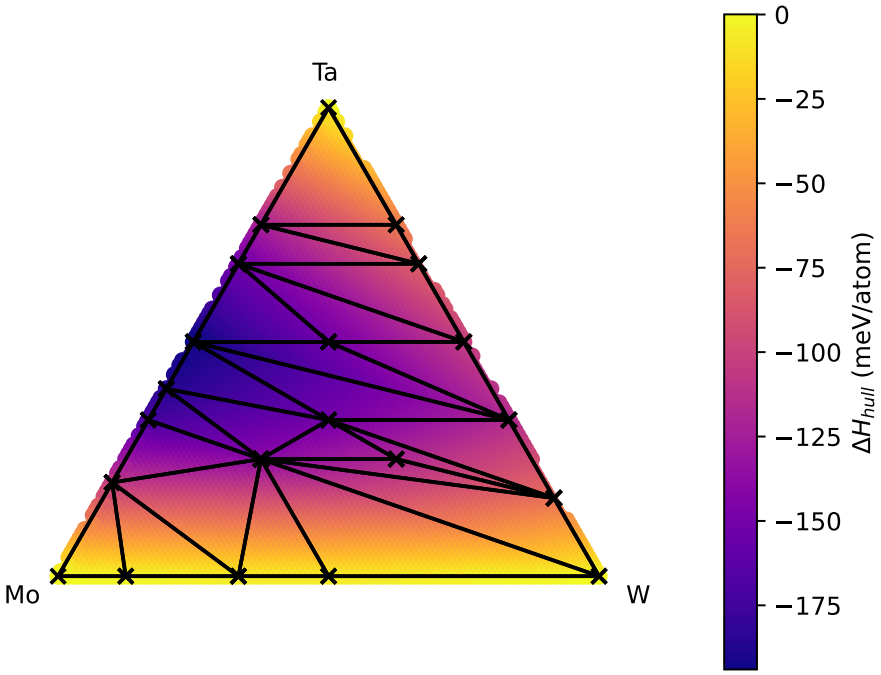


Figure 5.7: The AFLOWlib convex hull for the ternary Mo-Ta-W system. The crosses mark the convex hull points, the black lines are the convex hull tie lines, and the colour bar indicates the depth of the hull ΔH_{hull}.

A recent study of this system by Sotskov *et al.* [316] used a newly developed MLIAP aimed at predicting configurational energy and to be used in a crystal structure prediction scheme. In this computational work, they identified three new stable binary compounds (TaW_8, $TaMo_5$ and $TaMo_6$) and two new stable ternary intermetallics at $MoTa_2W_2$ and Mo_3Ta_3W. They also identified a near stable compound (within 1 meV/atom from the hull) at the $MoTaW_5$ stoichiometry. Since their work was directly compared to the convex hull from AFLOWlib, it can be compared to the results in this study. The work dislodged a number of AFLOWlib compounds from the convex hull, most notably the ternary $MoTa_2W$ compound. Note that the new ternary compounds they found were closer to the Ta-W edge.

5.3.2 Al-Fe-Ni

While the previous ternary system was chosen due to the richness of AFLOWlib's convex hull, the Al-Fe-Ni ternary convex hull is chosen because of the presence of

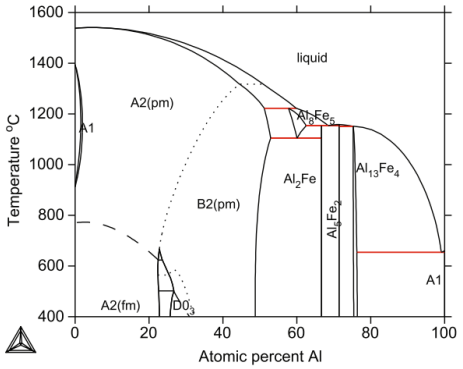
magnetic elements. The experimental phase diagram for this system is also very rich, for all three binary subsystems, as well as for the ternary region. Along with NiTi, these ternary alloys are notably used in industry as prominent shape-memory alloys [346]. Regarding the constituent elements, Al and Ni form an *fcc* phase, while Fe adopts the *bcc* lattice as a ground state at low temperatures, although its structural and magnetic ground states vary at higher temperatures [347]. Fe and Ni have a ferromagnetic ground state, while Al is paramagnetic. The melting points are 993 K, 1811 K and 1728 K for Al, Fe and Ni, respectively [215]. The experimental phase diagrams of the three binaries Al-Fe, Al-Ni and Fe-Ni are described in the following subsection and the corresponding AFLOWlib convex hulls are summarised.

Binary subsystems

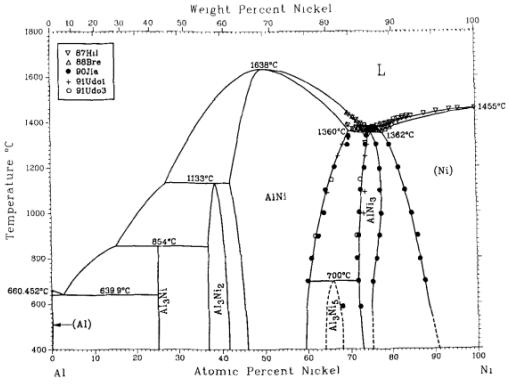
The experimental phase diagrams of the three binaries are shown in Figure 5.8. For the Al-Fe binary system, there are five stable phases, which are the $Al_{13}Fe_4$, Al_5Fe_2, Al_2Fe, AlFe and $AlFe_3$ intermetallics [348]. The Al-rich phases display a very small range of stability, while AlFe (ordered B2 phase) and $AlFe_3$ have larger stability ranges. There is a very large alloying ferromagnetic A2 phase on the Fe-rich side of the phase diagram, whereas no such alloying phase is close to the pure Al phase. Note that a high temperature intermetallic phase exists, with stoichiometry Al_8Fe_5.

A recent DFT study of this system [351] revealed that the Al_5Fe_2 phase and the $D0_3$ Al_2Fe one were not on the convex hull, while two Al-rich intermetallics, Al_6Fe and Al_9Fe_2 (mechanically unstable), were very close to the convex hull. Table 5.6 gives a summary of which phases are present on the AFLOWlib and the Materials Project (MP) convex hulls. Note that the $Al_{13}Fe_4$ phase is modelled with a stoichiometry of Al_3Fe. In the reference AFLOWlib database, only the Al_5Fe_2 phase is not present. Note that the $Al_{13}Fe_4$, Al_5Fe_2 and Al_2Fe phases form ternary alloys with Ni substitutions.

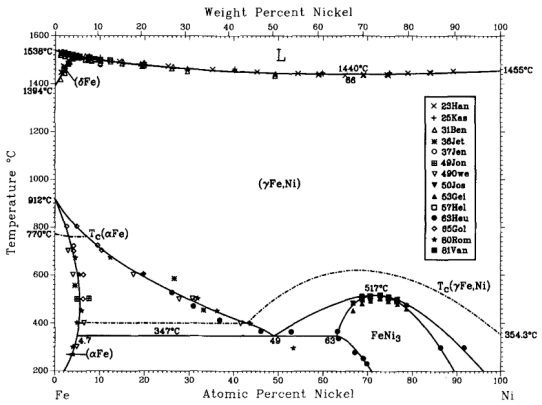
For the Al-Ni system, the phase diagram is similar, in that it presents Al-rich intermetallics with narrow stability ranges with respect to composition, the Al_3Ni and Al_3Ni_2 phases and an ordered B2 phase for AlNi with an extended stability range [349, 352, 353]. There are two Ni-rich intermetallics at Al_3Ni_5 and $AlNi_3$ and there is a significant alloying range with pure Ni, but not with Al. Note that the Al_4Ni_3 intermetallic phase (which is within the stability range of the ordered AlNi phase) has



(a) Phase diagram of Al-Fe.



(b) Phase diagram of Al-Ni.



(c) Phase diagram of Fe-Ni.

Figure 5.8: Binary phase diagrams of Al-Fe (left), Al-Ni (centre), and Fe-Ni (right). These phase diagrams were taken from references [348], [349], and [350], respectively.

also been synthesized, but is not always present in computed phase diagrams [354]. A first-principles study of this phase diagram shows good agreement with experiments, with a notable exception for the extended stability of the AlFe B2 phase [355]. On the computational databases, the convex hulls present all experimentally identified stable phases. The Al_3Ni and Al_3Ni_2 phases alloy with Fe to form ternary phases. The Ni alloy and $AlNi_3$ phases are known to be magnetic.

The FeNi system displays a rather different phase diagram to that of the other two systems [350, 356]. At high temperatures, it is entirely dominated by an A1 solid-state solution across the full compositional range. At lower temperatures, relevant for this study and for convex hulls, there is only one stable intermetallic phase, $FeNi_3$, which has an extended range of stability. The main portion of the phase diagram at low temperatures is dominated by a region in which there is coexistence of two phases, namely the $FeNi_3$ and Fe A2 alloy phases. Experimentally, ordered compounds for

Table 5.6: Table presenting the phases available present in different computational databases, AFLOWlib and Materials Project (MP), and ones in the experimentally computed phase diagrams (Exp.).

Database	$Al_{13}Fe_4$	Al_5Fe_2	Al_2Fe	AlFe	$AlFe_2$	$AlFe_3$
AFLOWlib	✓(Al_3Fe)	✗	✓	✓	✓	✓
MP	✓	✗	✗	✓	✗	✓
Exp.	✓	✓	✓	✓	✗	✓

FeNi and Fe_3Ni have been observed, although it is suspected that these are metastable or unstable. First principles studies have shown that FeNi and $FeNi_3$ phases lie on the convex hull [357]. The $FeNi_8$ phase is also present on the AFLOWlib convex hull, but has not been reported experimentally. All other relevant phases are present on AFLOWlib.

Ternary Phase Diagram

Phase diagrams for the Al-Fe-Ni system are only available in the literature for temperatures above 900K. The most up-to-date study of this phase diagram [358] is used as the reference for the present discussion. It is a CALPHAD study mostly based on experimental data. However, some first principles data for ternary compounds, notably "virtual" ones, is included to enrich the modeling. The computed phase diagrams at 1123K, as well as at the Al-rich corner at 900K, are shown in Figure 5.9. The latter corresponds to the most diverse out of all the phase diagrams and is hence highlighted here. The different ternary phases present can essentially be placed into three categories: ternary solid-state solutions, binary-based alloys and pure ternary intermetallics. The region of the phase diagram close to equimolarity is dominated by the β phase (notation from Figure 5.9), a solid-state solution with a *bcc* B2 structure, based on the B2 AlFe and AlNi phases. At low Al concentrations, the γ phase from the FeNi system is present and alloys with small concentrations of Al with the *fcc* A1 structure. At high Ni concentrations, there is a $L1_2$ phase, based on an alloying of $AlNi_3$ and $FeNi_3$. In the Fe-rich corner, at high temperatures, there is alloying with the pure Fe phase with the *bcc* A2 structure. Substitutional alloys based on binaries are present for many Al-rich binaries, which have narrow ranges of stability. These are

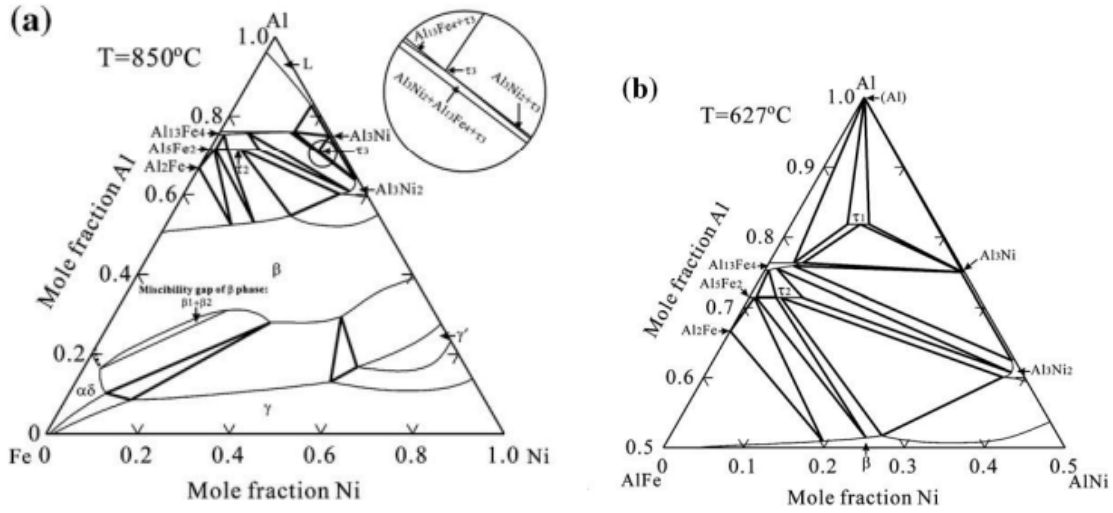
the $Al_{13}Fe_4$, Al_5Fe_2, Al_2Fe, Al_3Ni and Al_3Ni_2, as indicated in the previous subsection.

There are three known Al-Fe-Ni intermetallics, which occur in the region where there is no widespread presence of a solid-state solution, the Al-rich corner. This is notably due to the lack of solubility of Fe and Ni in Al. The three phases are labelled τ_1, τ_2 and τ_3. The most well-studied τ_2 phase is $Al_{10}Fe_{3-x}Ni_{1+x}$ (x between 0 and 2), typically in the $Al_{10}Fe_3Ni$ stoichiometry, but that extends to $Al_{10}FeNi_3$ [359] and includes the Al_5FeNi stoichiometry. The τ_1 phase has a stoichiometry $Al_9Fe_{2-x}Ni_x$ (x between 1 and 1.6) [360], typically referred to as Al_9FeNi, and has the Al_9Co_2 monoclinic crystal structure. Finally, the τ_3 intermetallic is at a very narrow compositional range at the specific stoichiometry of $Al_{71}Fe_5Ni_{24}$ and is only stable between 874K and 1203K [361]. The τ_1 and τ_2 phases are expected to affect the ternary phase diagram, and they are not present on AFLOWlib, which contains one ternary, the Al_2FeNi compound, on the convex hull. The *ab-initio* calculations provided by Reference [358] performed with VASP serve as a useful reference for the intermetallic ternaries, as well as the additional Al_2FeNi, $AlFe_2Ni$ and $AlFeNi_2$ compounds that were studied.

5.3.3 Bi-Fe-X

The two final phase diagrams investigated are that of the Bi-Fe-X systems, where X is one of the two refractory elements Ta or Zr. These systems were chosen as a space to deploy the workflows developed for magnetic material discovery. The inclusion of Fe is to provide a large exchange interaction and hence a high T_c, while Bi is selected for its high atomic number and hence provides strong spin-orbit interactions, which may give rise to a high magnetic anisotropy. It is also chosen as it is rarely used in ferromagnetic materials. Finally, Ta and Zr are chosen based on the fact that the Bi-Fe-(Ta,Zr) systems have not been extensively studied. Regarding the unaries that have not yet been presented, Bi forms a rhombohedral structure with a characteristically low melting temperature of 544K and Zr is another refractory metal with a *hcp* structure and a high melting temperature of 2125K.

There are two ternary convex hulls and five binary convex hulls related to these systems. For the two ternaries as well as the Bi-Fe and Bi-Ta subsystems, no compounds are present on the convex hull or the high-temperature phase diagrams. For the binaries, this implies that the two species do not mix [362]. Therefore, the three



(a) (b)

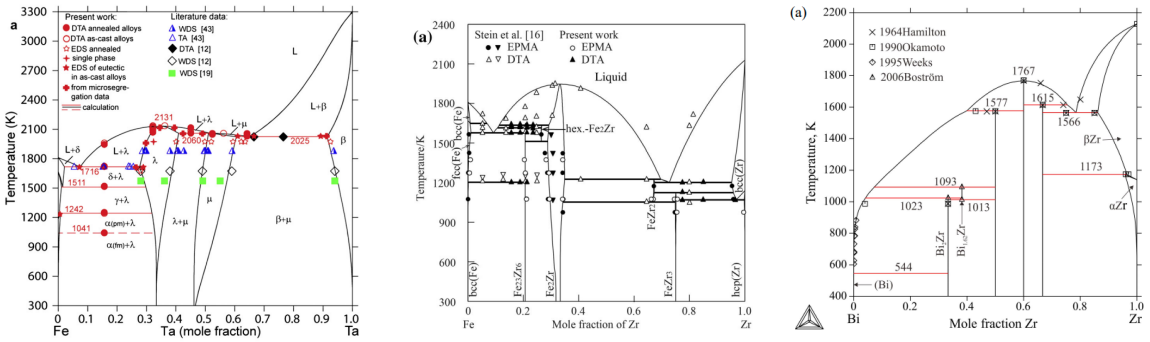
Figure 5.9: Phase diagram of Al-Fe-Ni at 1123K (left) and the Al-rich region of the phase diagram of Al-Fe-Ni at 900K (right) taken from Ref. [358]. The high-temperature phase diagram is computed with *ab-initio* data only, while the right one is with the full data, including experimental data.

binary subsystems with stable phases, Fe-Ta, Fe-Zr and Bi-Zr are presented briefly. Their phase diagrams are shown in Fig. 5.10.

For the Fe-Ta system, there are two experimentally observed stable phases that have an extended stability range [363]. These are Fe_2Ta and Fe_7Ta_6. A metastable Fe_2Ta_3 has also been reported. *Ab-initio* studies are consistent with the experimental phase diagram [364]. The Fe-Ta AFLOWlib convex hull presents three stable phases for Fe_2Ta, Fe_7Ta_6 and $FeTa_2$. The presence of the latter is inconsistent with experimental observations and it is in disagreement with the convex hull from MP.

The Fe-Zr system and its intermetallics are of notable interest for the disposal of nuclear waste [365]. As a result, there has been a very extensive investigation of its phase diagram, both experimentally and theoretically. There have notably been controversies on the stability and temperature ranges of various stable structures. There is an extensive literature review in Reference [365] and the phase diagram is taken from the same reference. Two phases have been accepted to be stable even at low temperatures, namely Fe_2Zr (cF24) and $FeZr_3$ (oC16). Another phase, $FeZr_2$ (tI12) has been stabilised at low temperatures, but it has been concluded that it is metastable at such temperatures and is only stable between 1053-1224K [366]. At high temperatures, a hexagonal Fe_2Zr (hP24) phase and the $Fe_{23}Zr_6$ phases have also been observed. The

former has been accepted to be a high-temperature phase, while for the latter, it has not been established whether this is a stable phase of the binary system or not. It is possible that the phase is stabilised by oxygen, rather than being a pure binary [366]. The most recent study suggests it is stable even at low temperatures [365], which differs with conclusions drawn from *ab-initio* studies [367, 368, 369]. The Fe-Zr AFLOWlib convex hull presents three stable phases for Fe_5Zr, Fe_2Zr and $FeZr_3$. The presence of Fe_5Zr is not in line with other studies and the MP. However, it lies only just "on" the convex hull (3 meV/atom below if it were removed), and the other two phases are the more important ones. The absence of $Fe_{23}Zr_6$ is in line with other DFT studies.



(a) Phase diagram of Fe-Ta. (b) Phase diagram of Fe-Zr. (c) Phase diagram of Bi-Zr.

Figure 5.10: Binary phase diagrams of Fe-Ta (left), Fe-Zr (centre), and Bi-Zr (right). These phase diagrams were taken from references [363], [365] and [370], respectively.

Finally, the Bi-Zr system possesses four intermetallics with no stability range at Bi_2Zr, BiZr, Bi_2Zr_3 and $BiZr_2$ [371, 370]. At high temperatures, the $Bi_{1.62}Zr$ phase has also been synthesized [372]. The Bi-Zr AFLOWlib convex hull presents four stable phases for Bi_2Zr, BiZr, $BiZr_2$ and $FeZr_8$. The latter is inconsistent with experiment, while there is a missing point for Bi_2Zr_3. The relevant points are present on the MP.

Following the presentation of the novel methods employed in this study and the phases that will be investigated, the workflow constructed is now presented. The first version of the workflow is centered around SNAP as a screening tool, and it is used to make predictions regarding the stability of ternary compounds of the Ag-Au-Cu and Mo-Ta-W systems.

5.4 SNAP Workflow Development

In this section, the main workflow is presented. The aim is to make use of the SNAP screening model developed in the previous chapter to accelerate the construction of DFT ternary convex hulls. As explained in the introduction, this is typically done in two steps, the construction of the database of candidate compounds that could lie on the hull, and the evaluation of their energy with DFT. An additional efficient screening step is inserted between these two in the workflow, which filters only the lowest energy compounds to be passed to the more costly but higher fidelity *ab-initio* energy evaluation. A key component of the workflow is the choice of candidate compounds used to construct the hull. The pool of screened intermetallics must be near equilibrium alloys, since this is a requirement for optimal performance of the SNAP model. In an attempt to achieve this, the structures of the binary compounds that lie close to the convex hull are chosen as prototypes. These are then decorated with three species following the ENUMLIB algorithm to form the database of candidate compounds. It is postulated that, since similar chemical environments are encountered in ternary and binary compounds, and that the structures of the binary compounds are already relaxed, these could provide close to equilibrium structures for ternary phases. SNAP is then used to drive a screening process with additional steps than just a straight energy prediction to improve the robustness and reliability of the screening outcome. In Figure 5.11, a diagram illustrating the overall architecture of the workflow is given.

The philosophy of this section is to give a description of the investigation followed during the workflow development. To a great extent, the order of the results shown here are in line with the chronological timeline of the progress made. As a result, the results presented may not follow one another logically or smoothly, as is the nature of the investigative process. The Ag-Au-Cu system is again used as a test system. Comparisons are made to the AFLOWlib convex hulls. The methodology used for AFLOW's hull constructions is termed the dictionary method [235].

5.4.1 Binaries as Prototypes

The construction of the library of candidate compounds (ternary prototypes in Figure 5.11) remained unchanged during the development phase of the workflow, notably due

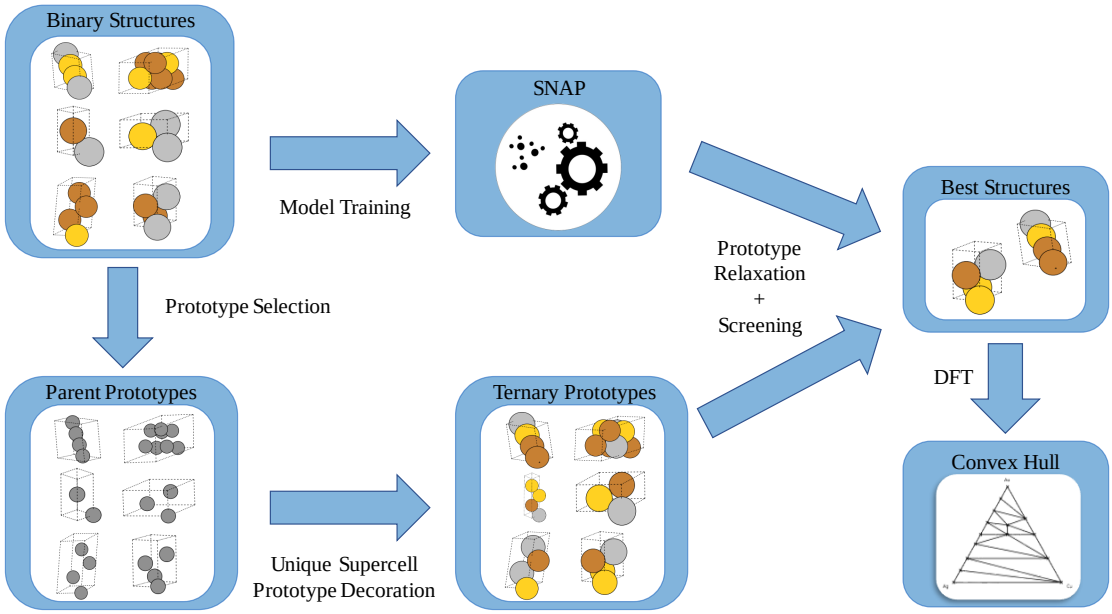


Figure 5.11: Schematic diagram of the workflow for the construction of ternary convex hulls.

to the early success with the databases generated, as well as the relatively low number of parameters to tune in the process. Firstly, the relevant binaries are selected, namely their structures are chosen as parent prototypes to be decorated. However, not all compounds from the database of binary subsystems are chosen. Instead, only those with enthalpies of formation lying on, or close to, the convex hull are taken. For this, a fixed energy window above the hull at all stoichiometries is created, and any compounds lying within this bracket are added to the database. This is performed for the three binary subsystems. The choice of the size of the enthalpy interval was initially set to $k_{\mathrm{B}}T$, with T set to 273 K. However, this led to a different number of selected compounds for each binary and too high a number of total structures. The Ag-Au-Cu system serves as a good example to see this effect. For Ag-Au, there are many low-lying structures relative to the hull, whereas for Ag-Cu, notably in the region of equimolarity, the lowest enthalpy systems lie more than 50 meV/atom above the hull. A fixed energy window would thus lead to the selection of more structures originating from the Ag-Au subsystem. Instead, the energy window is system-dependent and adapted such that the number of selected compounds per binary is constant and fixed at 25. These energy windows are shown in Figure 5.12 for the Ag-Au and Au-Cu systems. Note that for the latter, since the convex hull is a tie line joining the two elemental points,

the energy window is constant across the full compositional range and is represented as a rectangle.

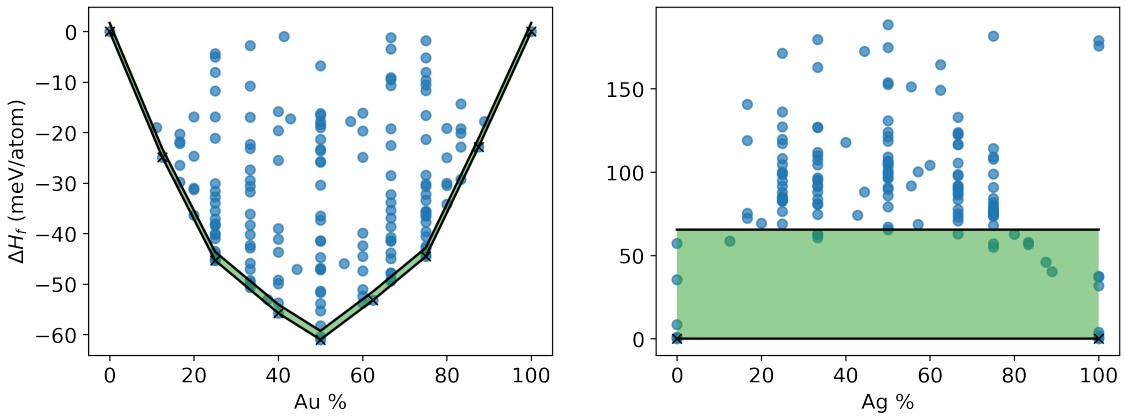


Figure 5.12: Binary convex hulls of Ag-Au (left) and Ag-Cu (right). The blue points represent different compounds from the AFLOWlib repository, while the black crosses indicate the position of the hull points. The bottom black lines are the tie lines of the convex hulls, and the top ones delimit the upper end of the energy window. The green-shaded area is the energy window.

The energy windows for the three binary subsystems are 1.7, 65.4 and 6.2 meV/atom for Ag-Au, Ag-Cu and Au-Cu, respectively. Following this selection, the bare structures are obtained following the removal of the chemical information from all compounds, and these are compared using the AFLOW symmetry tool to obtain the unique structures. Only the structures are of interest since they are later decorated with three species. This leads to a reduction of the size of the database, which for Ag-Au-Cu system is 45 compounds, down from the initial 75. This library of prototypes is fixed and is the starting point for the rest of the workflow.

The workflow is run for each stoichiometry independently. The first step consists in generating, from the full set of undecorated parent prototypes, the ternary prototypes used as candidates for the construction of the ternary convex hull. A maximum size of the unit cells generated (in terms of a maximum number of atoms, $N_{\max}$) is first chosen. Note that this is stoichiometry-dependent. Following this choice, the compatible prototypes are chosen. Only certain prototypes can be used for each composition, since the number of atoms of the prototype unit cell, or of a corresponding supercell with $N \leq N_{\max}$, must be a multiple of the sum of the stoichiometry. For example, for the 1-2-1 ($AgAu_2Cu$) stoichiometry and an $N_{\max}$ of 12, only prototypes which have cells/supercells with 4, 8 or 12 atoms are compatible. Thus, $N_{\max}$

is composition-dependent, since depending on the stoichiometry, a different number of parent prototypes are compatible. Moreover, as $N_{\max}$ is increased, the number of candidate structures increases drastically. For a parent *fcc* prototype (one atom per unit cell and hence compatible with all stoichiometries up to $N_{\max}$), the number of unique derivative structures increase from 1299 at $N_{\max}$=10 to 867,893 at $N_{\max}$=20 [325]. In an attempt to homogenise the number of generated candidates for each stoichiometry, $N_{\max}$ is adjusted manually for each stoichiometry. Following this selection of compatible prototypes, the ternary structures are generated with ENUMLIB at a fixed composition. For the Ag-Au-Cu system, the number of compatible parent prototypes, as well as the total number of ternary prototypes, are given in Table 5.7 for the different stoichiometries studied. In this case, a fixed value of $N_{\max}$=10 is chosen.

Table 5.7: Number of compatible parent prototypes and total number of ternary prototypes for each stoichiometry tested for the Ag-Au-Cu system. Note that the numbers are identical for permutable stoichiometries (e.g., 2-1-1 and 1-2-1).

Stoichiometry	1-1-1	2-1-1	3-1-1	2-2-1
# Parent Prototypes	18	19	7	7
# Ternary Prototypes	13,288	6,162	9,506	22,890

5.4.2 First Attempt at Screening

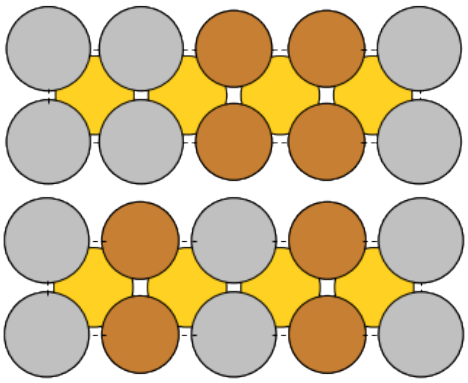
A first test run is performed to assess the credentials of binary structures for ternary prototypes. This is done for the AgAuCu stoichiometry, by computing the SNAP energies of all the novel compounds generated, as described in the previous subsection. The DFT energies are then computed for a subset of these and the results are compared. There are some important notes regarding the SNAP model employed here. Firstly, a single SNAP model, rather than an ensemble, is used for simplicity. Secondly, the hyperparameters of the model are different to those used in the previous chapter. This is for chronological reasons, as the Bayesian hyperparameter optimisation methodology followed formerly was only adopted prior to the workflow development. Instead, a simpler grid search is used here. The hyperparameters used here are R_{cut}=3.5Å, $J_{\max}$=4, λ=10^{-3}, p_{tr}=0.8 and N_{models}=1. The training/CV sets are the same as previously. A 10-fold Monte Carlo cross-validation scheme was followed for the optimisation process,

and the parameters were optimised independently following a grid search.

Finally, some consideration must be given regarding the choice of atomic weights. In the grid search, integer values between -5 and 5 (excluding 0) are considered for all 3 species. This leads to 1,000 different possible permutations of all weights. Crucially, some sets of weights are unsuitable for this application. Indeed, because of the high symmetry of some of the parent structures (*fcc*, *bcc*, etc.) the chemical environments of atoms in the ternary prototypes, which are simply derivative structures of the parent ones, will be structurally identical. In other words, if all chemical identities were removed, they would be identical chemical environments. Only the chemical identity breaks this symmetry and it is required since all the derivative structures are unique. This is illustrated in Figure 5.13, which shows that depending on the choice of atomic weights, two different compounds may have the same SNAP predicted energies. In the example shown, this happens when the weights of Ag and Cu are the same. This could occur with the other pairs of weights for equivalently decorated compounds. Given this constraint, the best-performing set of weights on the cross-validation error was given by w^{Ag}=1, w^{Au}=-2 and w^{Cu}=-1.

This model is deployed for the prediction of the energies of the ternary prototypes. Ultimately, the aim is to identify the ones lying closest to the convex hull, so that the energy ordering of the compounds is of particular interest. This is used as the main test of the workflow's performance. The ternaries generated are ordered according to their SNAP energies, which are used to filter the compounds passed on to DFT. Since the aim at this stage is to assess the workflow, not only are the top 10 lowest enthalpy compounds selected, but so are the 9 compounds ranked 20^{th}, 30^{th}, etc. and the 9 compounds ranked 200^{th}, 300^{th}, 400^{th}, etc., along with the 3 highest ranked compounds, which are outliers. These are relaxed with VASP, following the same conventions presented in Chapter 4.

Figure 5.14 shows two plots that are typical of those used as tests. Firstly, the *relaxed* energy or enthalpy of formation (equivalent at a fixed stoichiometry) is plotted as a function of the SNAP-predicted energy ordering. Note that the relaxed DFT energy is chosen here, as opposed to the unrelaxed one, despite the fact that the SNAP predictions are made on the unrelaxed compounds by construction. This is because of the fact that the former remains the quantity of interest when constructing a convex

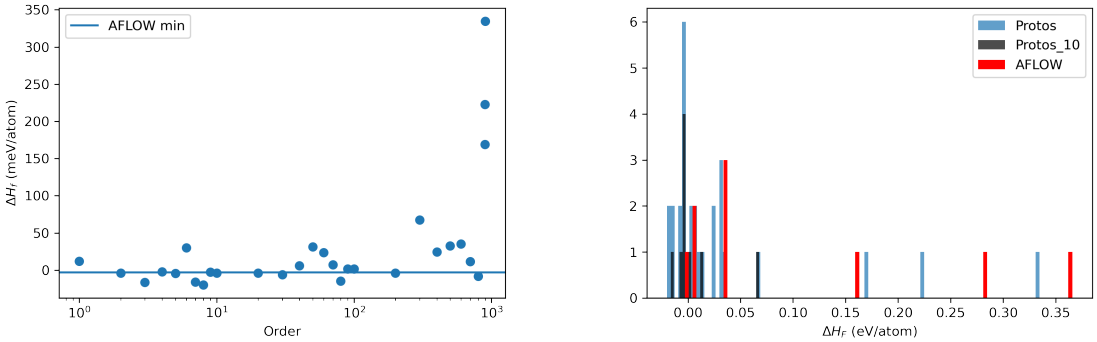


w^{Ag},w^{Au},w^{Cu}	E^{A}_{SNAP} (eV)	E^{B}_{SNAP} (eV)
1, 1, 1	-25.640	-25.640
1, 2, 1	-25.879	-25.879
2, 1, 1	-25.717	-25.676
1, 1, 2	-26.162	-26.047
1, 2, 3	-25.640	-25.742

Figure 5.13: Performance of SNAP for two structurally identical prototypes. The upper two panels show two different possible site occupations for a $3 \times 1 \times 1$ *bcc* derivative structure with $AgAu_2Cu$ stoichiometry (the top compound is A and the bottom one is B). Here we show the z-axis view, with gray, yellow, and brown spheres representing, respectively, Ag, Au and Cu. The table shows the SNAP-predicted energies for the two prototypes, when the SNAP is trained with different atomic weights, w^{α}, as indicated in the first column. Note that when the Cu and Ag weights are identical, the two energies coincide.

hull. Therefore, it is ultimately desirable that SNAP is able to accurately predict what this energy will be. Adding a relaxation step, or at least driving the structures closer to equilibrium before the SNAP predictions, would increase the accuracy of this prediction, both because the structure would be closer to the *ab-initio* relaxed configuration and because SNAP performs better in such cases. Regardless of what may be added, the relaxed energy remains the most important quantity to predict. The second plot in Figure 5.14 shows the enthalpies of formation of the ternary prototypes selected in the workflow, alongside the AFLOWlib ones. This serves to compare how the initial workflow performs against the dictionary method.

There are two main outcomes to be taken from these results. The first is that the choice of binary compounds as prototypes is a judicious one, since many of the novel ternaries have enthalpies of formation close to the ones chosen by AFLOWlib



(a) (b)

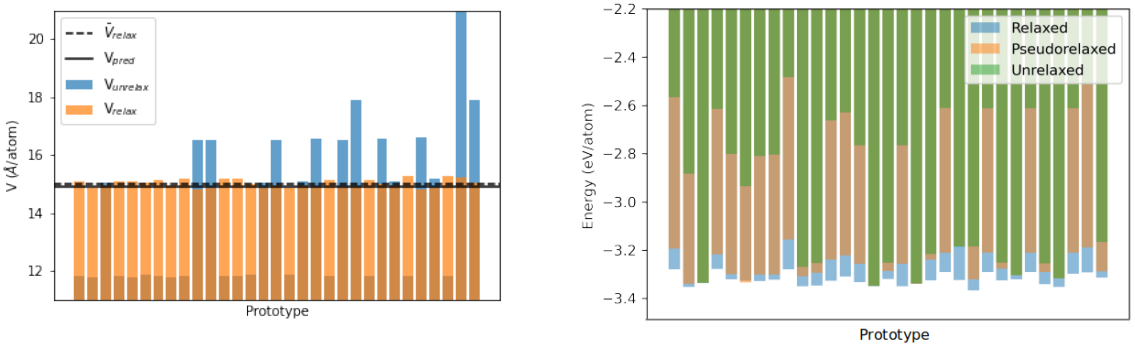
Figure 5.14: Panel (a) displays the relaxed enthalpy of formation as a function of the SNAP predicted energy ordering for the initial AgAuCu compounds. The full blue line indicates the enthalpy of formation of the lowest energy AFLOWlib compound. Panel (b) shows a bar plot of the enthalpies of formation of the ternary prototypes (Protos), the top ten prototypes (Protos_10) and the AFLOWlib compounds (AFLOW).

and, in fact, lie lower in energy than them. The second is that SNAP can serve to filter some of the compounds generated, as the lowest energy compounds figure in the SNAP top ten predictions. Nevertheless, the ordering is far from perfect, and it is difficult to draw definite conclusions on the small subset of compounds studied. On average, the change in DFT energy upon relaxation is of the order of 100 meV/atom, indicating that the compounds are still far from equilibrium. The SNAP-predicted energy ordering is significantly better if the DFT unrelaxed enthalpy of formation is plotted as a function of SNAP ordering (what is actually predicted). These encouraging first results stimulated further development of the workflow and notably to drive the candidates close to equilibrium before the screening step.

5.4.3 Volume Scaling

The first step taken towards improving the initial structure of the candidates focuses on an adequate guess of the cell volume. In the previous section, the volume of the parent structure is maintained, despite the difference in the atomic basis decoration. Subfigure 5.15a shows how the volume of the ternary candidates for the 2-1-1 stoichiometry changes upon DFT relaxation. The mean relaxed volume $\overline{V}_{\text{relax}}$ is also indicated. The plot has several notable features. The first is that for most prototypes, there is a significant change in the equilibrium volume upon relaxation. This could

be expected, considering the difference in chemical nature and atomic volumes of the constituent atoms in the binaries and ternaries. The second is that many of the ternary compounds have similar initial volumes. This comes from the fact that they are from prototype structures belonging to the same binary subsystem. Finally, the most important finding is that these candidates all have a similar relaxed volume and that there is little deviation from the mean value. This value is almost identical to V_{pred}, indicated by the full line in Subfigure 5.15a. This predicted volume is the suggested starting volume for the ternary compounds and corresponds to the stoichiometrically weighted sum of the atomic volumes of the constituent elements. In Figure 5.15b, the energies of the unrelaxed (without any volume adjustment), pseudo-relaxed (volume scaled), and relaxed compounds are displayed for the same ternary candidates. In all cases, the energies of the pseudo-relaxed compounds are lower or equal to that of the associated unrelaxed ones and, in many cases, are more than 0.5 eV/atom lower. It is also always less than 100 meV/atom away from the relaxed configuration. Based on these findings, a volume scaling step is added at the beginning of the workflow when the prototypes are generated.



(a) (b)

Figure 5.15: Diagnosis plots for 2-1-1 ternary candidates. Each bar is associated with a different candidate structure. Subplot (a) shows the initial and final volumes of the compounds (initially unscaled) following DFT-driven relaxation. The dashed (full) black line corresponds to the mean (weighted sum) sum volume denoted $\overline{V}_{\text{relax}}$ (V_{pred}). Subplot (b) shows a bar plot of the DFT energies of the unrelaxed, pseudo-relaxed (scaled volume), and the *ab-initio* relaxed ternaries.

5.4.4 SNAP Ensemble

Another improvement implemented relates to the screening model employed. In the previous chapter, an ensemble SNAP model was introduced, which had fixed hyperparameters and different subsets of the training set used to distinguish the individual models. This notably led to the use of the standard deviation as a performant error metric. For this work, this concept is deployed again to increase the reliability of the predictions. In this case, however, 5 models are used in the ensemble and these are trained on the same subset of data but each with different atomic weights. This is enforced in order to distinguish similar derivative structures of high symmetry compounds, as already detailed in Section 5.4.2. This strategy has the unfortunate disadvantage of decreasing the computational speed of the predictions made by the ensemble by a factor proportional to the number of models. Indeed, the difference in hyperparameters between models enforces a re-computation of the bispectrum components for each one, unlike the previous ensemble model, where the hyperparameters were fixed. This justifies the decrease in the number of models of the ensemble.

Table 5.8: Training and cross-validation (CV) mean absolute errors (MAE) for the 5 models with differing sets of weights, w^{α}, of the ensemble. All values are given in meV/atom.

w^{Ag},w^{Au},w^{Cu}	Training MAE	CV MAE
1, 2, 1	8.0	27.1
2, 2, 1	8.7	24.8
-2, -1, -1	9.7	30.6
-2, -2, -1	8.5	23.5
-1, -2, -1	7.7	25.6

The weights used for each model and their associated training/cross-validation MAEs are given in Table 5.8. All other hyperparameters are unchanged from the previous subsection. The average CV MAE of the ensemble model is of 22.1 meV/atom. This increase in error as compared to the previous model is attributed to two main factors. The first is the restriction of the weights, which implies that the models employed are not the most accurate ones, as the sets of weights used must distinguish

all pairs of weights at least once. Secondly, the optimisation process is based on a grid-space search, an optimisation methodology that is not as effective as the Bayesian optimisation algorithm previously used. Nevertheless, the ensemble model developed improves in performance relative to the single model and is used as the screening tool for the SNAP-based workflow. This can be seen in Figure 5.16. The same analysis as that carried out in Section 5.4.2 to assess the workflow for the 1-1-1 stoichiometry is carried out for the 2-1-1 stoichiometry, with the single model (left) and with the ensemble model (right) as screening tools. Note that the compounds were selected among the full dataset by the single SNAP model. It is only the *ranking* of this subset of compounds that is shown for both models. This energy ordering of ternaries at the 2-1-1 stoichiometry was initially very poorly predicted by the single model, whereas the ensemble model performs much better, notably for the lowest energy compounds, which it identifies. The lowest enthalpy compound is ranked first (ranked 700$^{\text{th}}$ by the single model), while the other two that are close in energy to the AFLOWlib minimum-energy compound rank in the top ten. Note that what is important is to sample all low-energy structures at the lowest ranking. Similar results are observed for other stoichiometries, so the ensemble is adopted as a screening tool.

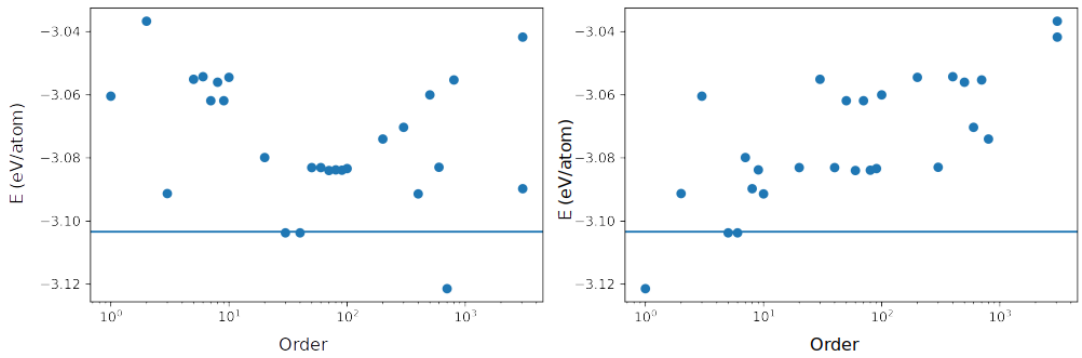


Figure 5.16: Plots of the DFT relaxed energy of a selection of 2-1-1 candidate ternary compounds against SNAP ordering. In the left plot, the single SNAP model is used for the ordering, while in the right plot the ensemble is used for ordering. Note that the full blue line displays the AFLOWlib minimum energy at that stoichiometry.

5.4.5 Relaxation

The improvements presented thus far aid the reliability of the workflow predictions, but are not enough to bring the ternary structures closer to equilibrium. For this,

ionic relaxation is required. In the last chapter, it was demonstrated that SNAP could not be used to drive reliable relaxation, as many structures collapsed or ended up in very high-energy configurations. This is, however, for structures that are from equilibrium. The candidate binaries used possess more appropriate structures with respect to compound stability, i.e., are closer to equilibrium, as the positions are pre-optimised from the binary compound relaxation, and the volumes are judiciously scaled. As a result, it is postulated that SNAP-driven relaxations may be more reliable, since the starting chemical environments, as well as those encountered during the relaxation path, may be a lot closer to the training set. It was quickly established that cell parameter optimisation was not possible with SNAP, notably due to a bias toward high compression of cells. As such, the tests concern ionic relaxation, where only the atomic positions are updated. As will be described, while SNAP can drive relaxation to a certain extent, it is not always reliable. Several steps are therefore introduced to increase the robustness of the relaxations. These are explained in the following subsections.

SNAP-driven relaxation

The need to include ionic relaxation was motivated following the running of the workflow for the 1-3-1 stoichiometry. The 16 lowest energy compounds selected were relaxed with DFT. Figure 5.17 shows the plot of the prototype order as predicted by SNAP against the unrelaxed energy. The configuration ranked 9^{th} had an energy significantly higher than the other compounds, notably in the top 10. However, upon DFT relaxation, despite almost no change in volume and cell parameters, the energy decreases by 191 meV/atom. This makes the SNAP predictions poor since they are on the out-of-equilibrium configurations, which have a much higher energy. The energy prediction is repeated following a SNAP relaxation, driven by a single model of the ensemble. This induces a large drop in the DFT unrelaxed energy, as SNAP successfully drives the structure close to equilibrium. The DFT relaxation only reduced the energy by a further 12.5 meV/atom. The SNAP-driven relaxation, therefore, enables better screening, as the compound is close to equilibrium, and leads to a reduction in the number of *ab-initio* ionic relaxation steps.

A larger benchmark test is performed for the 1-2-2 stoichiometry and the results

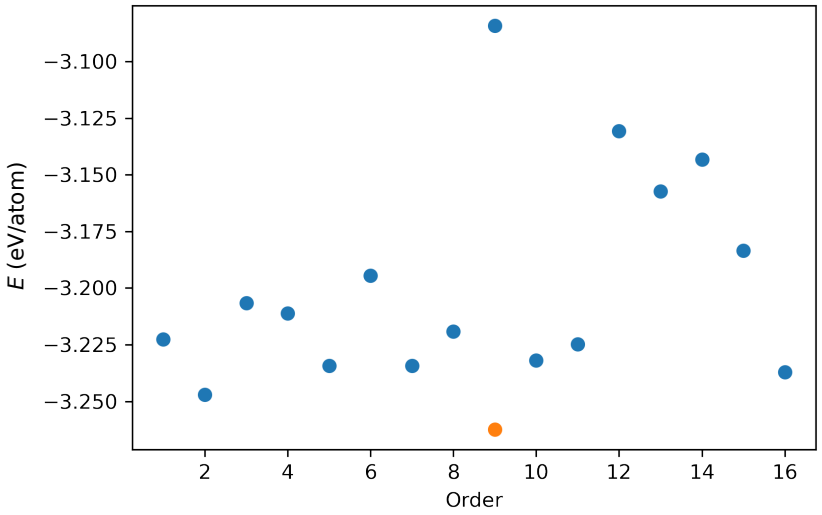
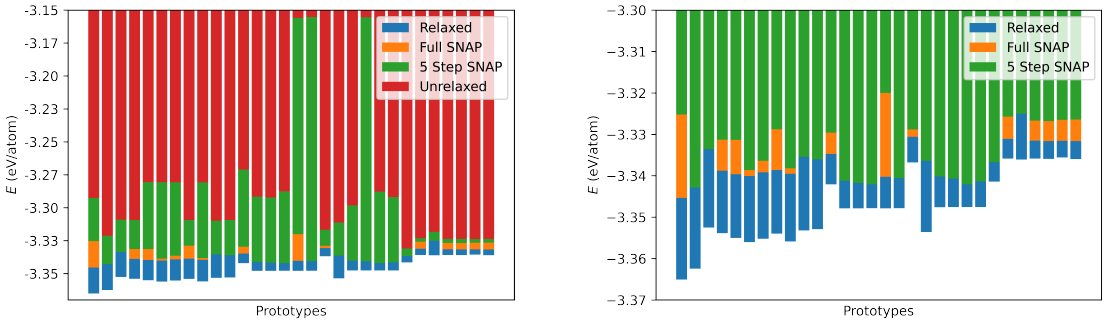


Figure 5.17: Unrelaxed energy of the top 16, 1-3-1 ternary candidates as a function of the SNAP predicted order. The blue points are for compounds that are not pre-relaxed with SNAP. The orange point shows the result for one point (ranked 9^{th}) that is pre-relaxed.

can be seen in Figure 5.18. Note that these tests are performed after the implementation of two steps in the workflow described in the next two subsections. This shows different DFT energies for the top 30 compounds selected by the workflow. These energies correspond to different structures per compound. These structures are: the "completely" unrelaxed structure, the fully DFT-relaxed structure, the structure pre-relaxed by SNAP (Full SNAP), but not DFT, and the equivalent structure, but for a 5-step SNAP ionic relaxation (5 Step SNAP). For the latter, the SNAP relaxation is cut short at 5 steps. The results show the significant benefit brought by pre-relaxing the compounds with SNAP. There is a large energy gain between the unrelaxed and the 5-step SNAP relaxed compounds, which on average have energies 45.1 meV/atom lower. Crucially, the difference to the DFT relaxed structure is only 14.0 meV/atom, as opposed to 59.1 meV/atom, when not pre-relaxed. The difference is also sustained across the full range of prototypes, as even those that are further from equilibrium are well relaxed with SNAP. There is only a very marginal gain from doing the full SNAP relaxation prior to DFT since the difference between the full SNAP-relaxed and the DFT-relaxed systems is of 11.8 meV/atom.

These results reveal two important insights. The first is that, in this case, SNAP can in fact be used to relax compounds toward their equilibrium. The second, is that



(a) (b)

Figure 5.18: Bar plots, full (left) and zoom (right), of the DFT energies of different structures associated with candidates of the 1-2-2 ternary compounds. These are the Unrelaxed (no SNAP relaxation), Full SNAP (fully SNAP relaxed), 5 Step SNAP relaxed (relaxed by SNAP with 5 ionic steps) and Relaxed (fully DFT relaxed) structures.

the relaxation only needs to be carried out for a limited number of steps (5 shown here), since it is in these steps that the largest change in energy occurs. This can be seen in Figure 5.19, which shows that in the first 5 steps the SNAP energy difference between unrelaxed and relaxed compounds changes from $>10^{-2}$ to 10^{-3} eV/atom and to 10^{-4} eV/atom in the subsequent 5 steps. Another important point is that while the results shown are from successful SNAP relaxations, some fail if fully driven by SNAP (i.e., not stopped at 5 steps). There is therefore some filtering performed before moving to DFT to ensure the predictions are reliable. In this effort to increase the workflow robustness, it is important to keep the number of ionic relaxation steps fixed and set to 5. This number strikes a good balance between achieving enough relaxation for many compounds and not driving others, poorly described by SNAP, too far from equilibrium.

Cross-Validated Relaxation

In the last part of the previous section, it is not described how the relaxation was performed with an ensemble of SNAP models. From the 5 models trained, each is used to drive a 5 step ionic relaxation. Since the parameters and hyperparameters of each MLIAP are different, the 5 structures obtained differ. While all compounds could be retained for screening, many are very similar to one another, and only one is selected. In order to decide which one to retain, a "cross-validated" approach is taken. For

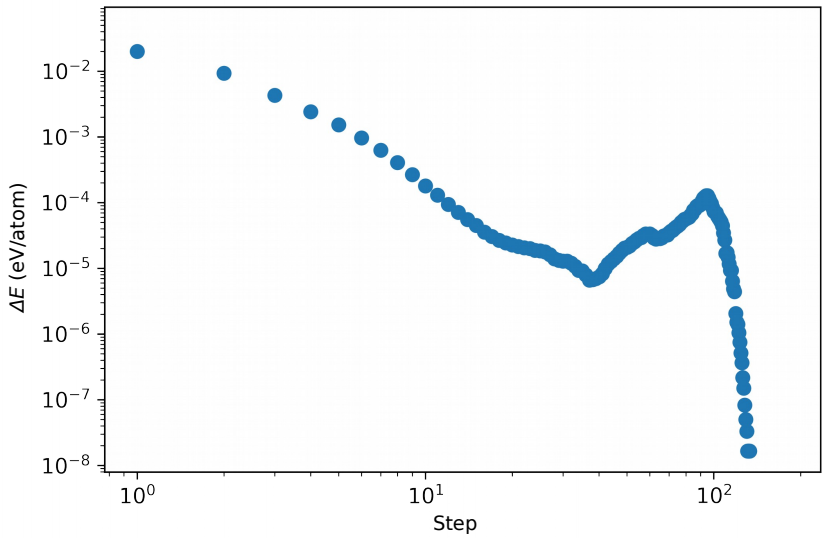
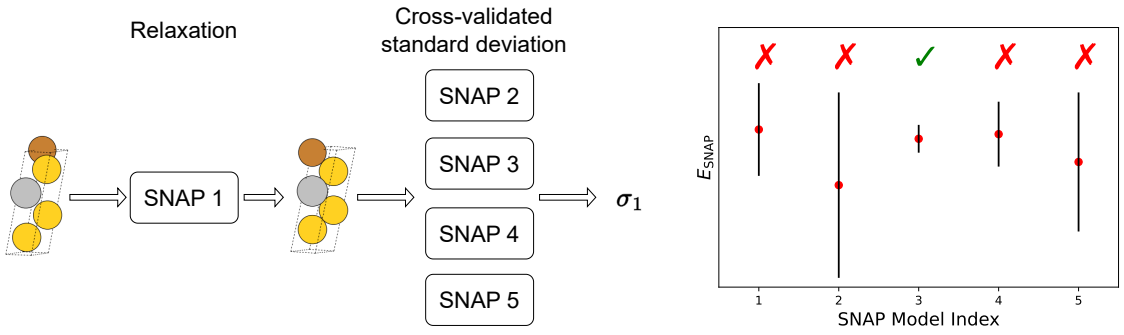


Figure 5.19: Average change in the SNAP energy with respect to the energy of the relaxed configuration as a function of step index for a subset of the 1-2-2 candidates.

each structure, the energy prediction is performed by the 4 models of the ensemble *not* used for the relaxation of the associated structure. Hence the name "cross-validated". The standard deviation, σ, of these predictions is retained and the structure with the lowest cross-validated standard deviation is kept. This sets the deciding criterion for structure selection. Of the 5 structures generated, the one with the lowest value of σ is chosen as the representative SNAP-relaxed structure for that prototype. The concept of cross-validated relaxation is illustrated in Figure 5.20.

The rationale for this approach is underpinned by a drive towards increased robustness of the workflow predictions. The relaxations are cross-validated by other models because of the bias of individual SNAP models. In the case of a failure during the relaxation process that leads to an unphysical or high-energy structure, the model driving the relaxation will predict a low energy for that compound, since it drove the energy minimisation process. This bias is removed for the structure selection process by preventing the SNAP-driving model from making an energy prediction. Furthermore, other SNAP models of the ensemble may also have a similar bias and predict a low energy for the poor structure generated. To circumvent the effect of this further bias, the standard deviation is used as a selection criterion rather than the mean energy. Indeed, the latter will be dragged down by very negative and poorly biased predictions. The standard deviation on the other hand will be large for such compounds as the different models will highly disagree on the energy predictions (even if they are all "too low").



(a) (b)

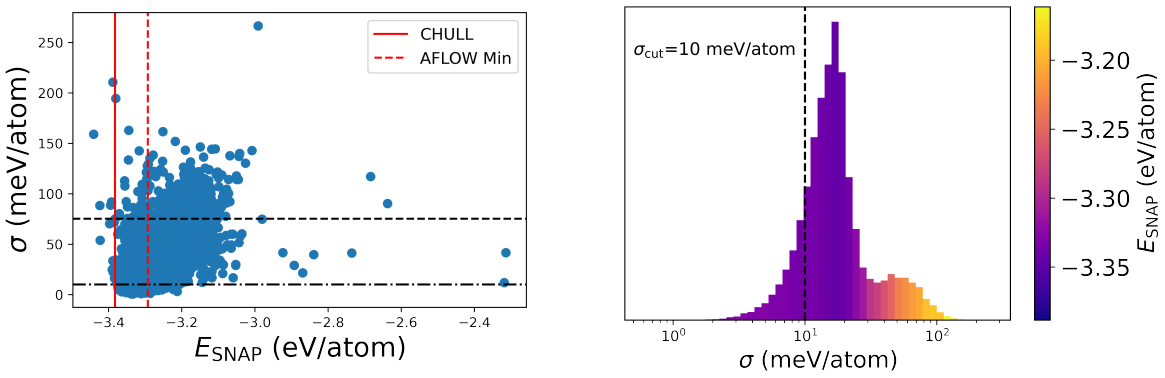
Figure 5.20: Illustrative figures of the cross-validated standard deviation selection methodology. Panel (a) shows a diagram illustrating the definition of the cross-validated standard deviation. The five SNAP models of the ensemble are labelled by integers and one relaxation out of five (performed by each model) is illustrated for one prototype. Panel (b) is a plot illustrating the selection process of the 5 relaxed structures for each prototype. The y-axis corresponds to the relaxing mean cross-validated SNAP energy and the error bars to the corresponding standard deviation. The structures resulting from the relaxation of the third model is selected since it has the lowest value of σ.

The standard deviation of the predictions has also been shown to correlate with DFT error very well.

Filtering with σ

Each prototype now has a SNAP-relaxed structure. As a last step, before completing the final selection of candidates, passed on to DFT, a filter on the cross-validated standard deviation is imposed. In practice, all compounds with $\sigma \geq \sigma_{\text{cut}}$ are removed from the selection process, where σ_{cut} is a cut-off value and a tunable parameter of the workflow. This implies that there is a removal of all compounds for which the disagreement between models is judged to be too large (large standard deviation). This step was found to be important in eliminating structures that SNAP predicted to have energies far below the convex hull, which turned out to contrast with the DFT results.

Subfigure 5.21a shows a plot of the standard deviation against the mean energy of the cross-validated predictions made by SNAP for the 1-2-2 candidates. The convex hull and AFLOWlib minimum energy at that stoichiometry are also displayed as a reference. Note that these are DFT values, whereas the data points are not. It can



(a) (b)

Figure 5.21: Panel (a) shows the SNAP cross-validated standard deviations against the mean energy energy for all candidates of 1-2-2 stoichiometry. The full (dashed) vertical red line is the convex hull (AFLOWlib minimum) energy as calculated by DFT. The horizontal dashed (dash-dotted) black line marks the 75 (10) meV/atom value of σ. Panel (b) shows the distribution of standard deviations of the compounds with the mean energy used as a colour scale. The cut-off value of σ_{cut}=10 meV/atom is indicated by the vertical dashed line.

be seen that for SNAP energies below that of the convex hull, the standard deviation is above 10 meV/atom and increases with increasing distance below the hull. There is a large cluster of points below 10 meV/atom that sits between the hull and the AFLOWlib minimum energies, which are the ideal candidates for the workflow. These are low-energy compounds for which predictions are made reliably. A higher line at 75 meV/atom is also displayed, which includes most compounds. Figure 5.21 shows how the mean energy varies as a function of standard deviation in the candidate standard deviation distribution, highlighting that on average, the lower energy compounds are more accurately described by SNAP. Finally, Figure 5.22 shows the selection of the top 15 candidates from the workflow at the 1-2-2 stoichiometry, when σ_{cut} is set to 75 meV/atom and 10 meV/atom. This shows that using a more conservative value for σ_{cut} gives better results.

5.4.6 Final Workflow

In the previous subsections, the finer details of the workflow were presented. Here, a more global summary of the workflow is given. It is supported by the illustration of the workflow given by the diagram in Figure 5.23. From the database of AFLOWlib

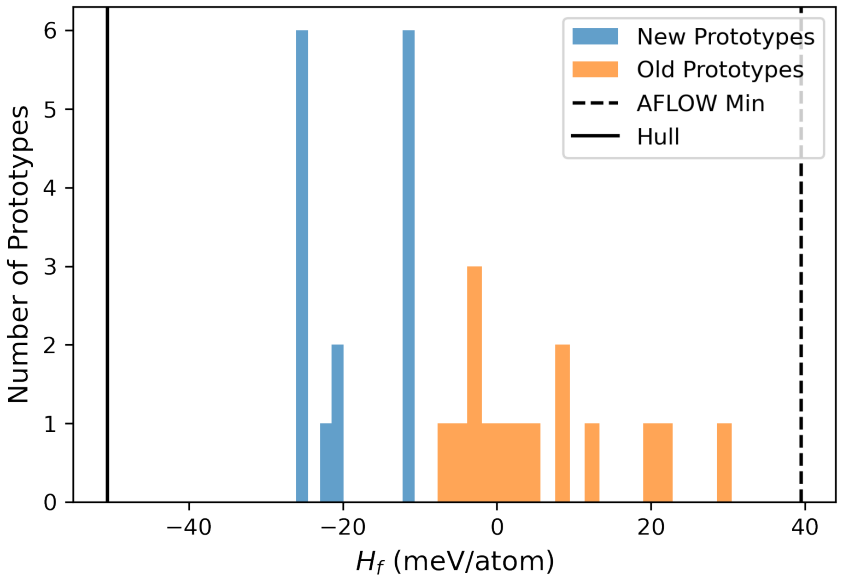


Figure 5.22: Distribution of the DFT enthalpies of formation of the top 15 selected prototypes for the 1-2-2 candidates with two different values of σ_{cut}. Protos is for 10 meV/atom while Old Protos is for 75 meV/atom. The full black line corresponds to the convex hull enthalpy, while the dashed line is the AFLOWlib minimum energy.

binaries, the ensemble SNAP model is trained and the binary prototypes are generated. The workflow is then run one stoichiometry at a time. Derivative ternary compounds of the parent prototypes form the pool of candidates. These undergo a cross-validated relaxation and screening with the SNAP ensemble model. For this step, each candidate is relaxed by the five different models and the structure, which yields the lowest cross-validated standard deviation, is selected. If this value is below σ_{cut}, the structure will move to the next stage. Following this filtering, the ternary compounds are ranked according to the SNAP energies and the top ones (typically 15) are selected for DFT relaxation. The lowest energy compound (as per DFT) is retained as the best attempt at reaching the convex hull for that stoichiometry. After this is repeated for all compositions of interest, the final convex hull is constructed.

There are several parameters that can be tuned in the workflow. Besides the obvious ones related to the SNAP models, the standard deviation cut-off value σ_{cut}, the number of steps performed in the SNAP-driven ionic relaxations and the number of ternary compounds retained for the DFT relaxations can all be changed. Here they are, respectively, set to 10 meV/atom, 5 steps and 15 compounds.

This concludes the section on the workflow development. The architecture remains

the same for the construction of the convex hulls of Ag-Au-Cu and Mo-Ta-W present below. Only the training of the SNAP models and the free parameters mentioned above differ.

5.5 SNAP Workflow Results

The results yielded by the SNAP workflow are now presented for the ternary convex hulls of Ag-Au-Cu and Mo-Ta-W systems. The workflow is run for the 1-1-1, 2-1-1, 3-1-1 and 2-2-1 stoichiometries and all associated permutations. Figure 5.24 shows the distance δ from the convex hull of the lowest enthalpy compounds discovered at each stoichiometry. Note that all results at this level are *ab-initio* ones, since the last step in the workflow involves a DFT relaxation. The convex hull is the recomputed one based on these results. These can be directly compared to the results from AFLOWlib. All best-performing compounds here are lower in energy than their AFLOWlib counterparts. Two points, at the $AgAu_2Cu$ and $AgAu_3Cu$ stoichiometries, are found to be below the convex hull. These are both in the Au-rich region of the phase diagram, which is consistent with the experimental phase diagram that presents stable ternary solid-state solutions only in the Au-rich region. The AFLOWlib convex hull presented no stable ternaries, although the $AgAu_2Cu$ composition presents a compound lying close to the convex hull, 3.13 meV/atom higher than the one revealed here. The family of 3-1-1 stoichiometries is not sampled in AFLOWlib. For the other part of the phase diagram, most points lie between 20 and 30 meV/atom away from the hull, aside from the 1-1-1 and 1-1-3 stoichiometries, for which the lowest enthalpy compounds are at 45.08 meV/atom and 47.12 meV/atom from the hull, respectively. The distance from the hull generally increases as the hull becomes shallower upon the approach of the immiscible binary Ag-Cu subsystem. In this respect, the 1-1-3 is not inconsistent with the trend. The 1-1-1 stoichiometry is, however, more of an outlier. The AFLOWlib lowest enthalpy compound lies at 57.09 meV/atom from the hull. It is unclear whether this result is an inherent feature of compounds at this stoichiometry (none are close to the convex hull) or if it is due to poor sampling by the workflow. The fact that this composition was the most studied during the development phase of the workflow suggests that it may be related to the composition.

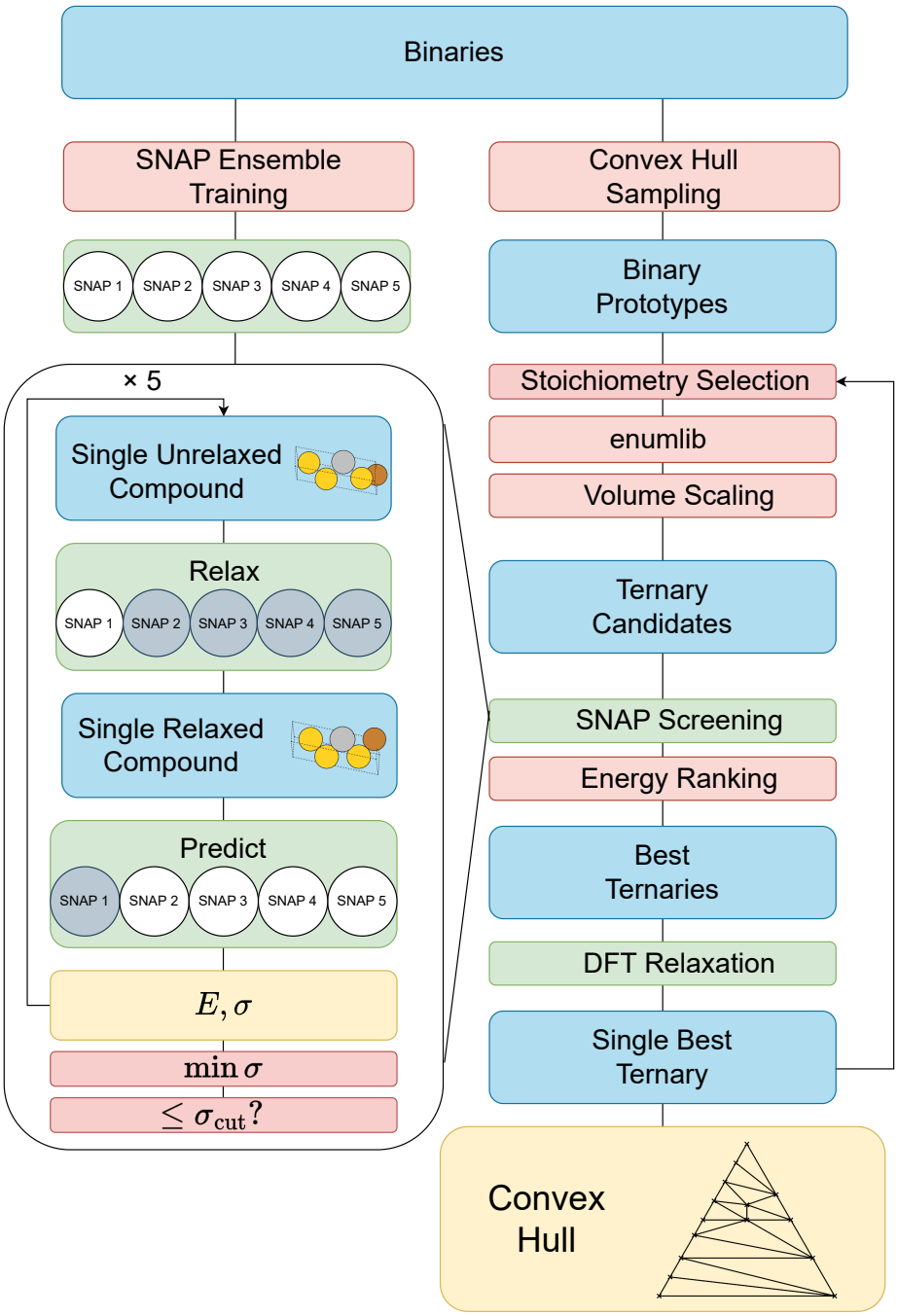


Figure 5.23: Diagram illustrating the structure of the workflow for the construction of ternary convex hulls. The left-hand panel shows the details of the SNAP relaxation and screening procedure, repeated 5 times, one for each model. The workflow is run for all desired stoichiometries from the Stoichiometry Selection step to the Single Best Ternary step (lowest DFT relaxed compound). The convex hull is constructed from these results. Here, E and σ are the cross-validated mean and standard deviations of the energy predictions.

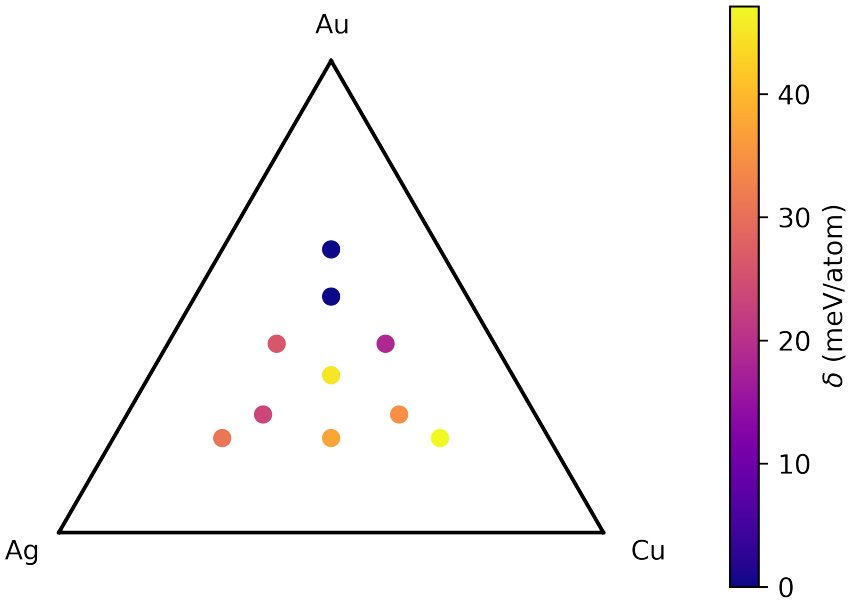
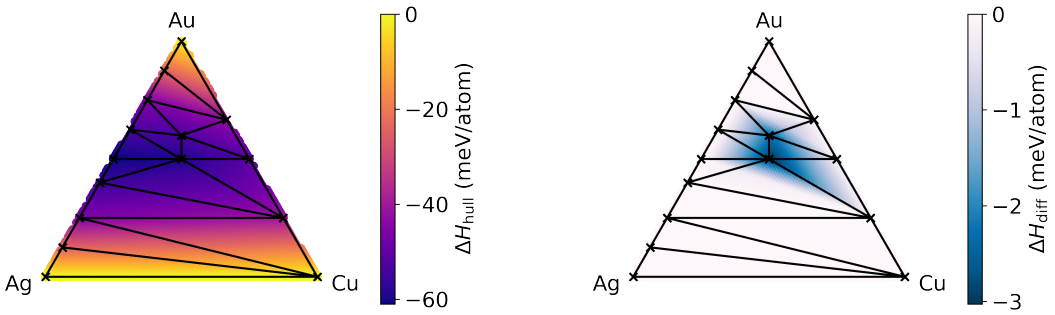


Figure 5.24: Plot of the ternary phase diagram of Ag-Au-Cu with the distance to the convex hull of the compounds given by the workflow. The position of the points indicates the compositions, and the colour scale gives the distance to the convex hull of the lowest energy compound of that stoichiometry.

The new convex hull is presented and compared to AFLOWlib's convex hull in the subfigures of Figure 5.25. The two new stable compounds identified at 1-2-1 and 1-3-1 are now present on the convex hull, and it is this region that is changed relative to AFLOWlib's convex hull. The compounds identified present the common structural feature of separating the Ag and Cu atoms by Au atoms in stacked structures. The 1-2-1 compound is a derivative structure (two cells) of a tetragonally distorted *bcc* cell, the cell being expanded along the direction with the largest elongation. The space group is 123, and it is from the Ag-Cu prototype 15 from AFLOWlib[1]. The 1-3-1 compound has a titled unit cell with space group 139, and its binary prototype is the Au-Cu f33 AFLOWlib binary[2].

Following the successful deployment of the workflow for scanning the Ag-Au-Cu system, the Mo-Ta-W one is chosen as a more challenging test. This system is the one with the most stable ternaries on AFLOWlib (4), with stable ternaries at the 1-

[1]auid: aflow:69a8361cc1cb1112
[2]auid: aflow:2c082f6dca0164b2

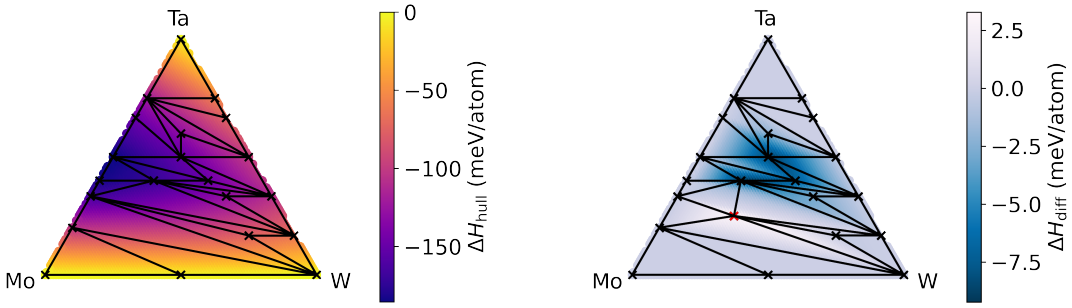


(a) (b)

Figure 5.25: Panel (a) shows the reconstructed Ag-Au-Cu convex hull from the AFLOWlib data and the aded ternary points given by the workflow. The colour bar gives the enthalpy of formation of the convex hull, the hull points are indicated by black crosses and the tie lines by black lines. Panel (b) shows the difference between the new convex hull and that computed from AFLOWlib data.

1-1 and all permutations of the 2-1-1 compositions. The aim is to determine whether the workflow can succeed in identifying stable compounds at these stoichiometries and even test other compositions. For this study, the same stoichiometries as well as the permutations of 4-1-1 are run with the workflow. Note that for the present phase diagram, the SNAP model is only trained on AFLOWlib data, so that the ∼1,600 DFT energies of the binary compounds are not performed. Only the energies of a small subset (around 10) are calculated, and used to construct the binary convex hulls, such that the energies of the novel ternaries suggested are consistent. This is an important point, since the model is deployed entirely on recycled data from AFLOWlib, from structures to energies.

The recalculated convex hull following the workflow run is presented in Figure 5.26a. Figure 5.26b shows its difference with respect to the AFLOWlib one. Note that, upon recalculating the convex hull from the energies of AFLOWlib's ternary compounds, two of the four ternaries are removed from the convex hull. Only the 1-1-1 and the 2-1-1 compounds remain. The former is removed from the hull when integrated with the new data provided by the workflow findings, as it is undercut by other points. The 2-1-1 would remain on the convex hull, and the minimum energy compound uncovered in this investigation did not beat AFLOWlib's one. In fact, AFLOWlib beats this point by 14.91 meV/atom and is 3.29 meV/atom below the workflow derived convex hull.



(a) (b)

Figure 5.26: Panel (a) shows the reconstructed Mo-Ta-W convex hull from the AFLOWlib binary data and the add added ternary points given by the workflow. The colour bar gives the enthalpy of formation of the convex hull, the hull points are indicated by black crosses and the tie lines by black lines. Panel (b) shows the difference between the new convex hull and that computed from AFLOWlib data. Note that the only AFLOWlib ternary still on the convex hull is marked in red.

Overall, the workflow outperforms AFLOWlib over most stoichiometries. Aside from the aforementioned 2-1-1 composition, AFLOWlib only has better-performing compounds at the 1-1-2 composition, where it is 2.23 meV/atom lower than the workflow results. For the 1-1-1 stoichiometry, the same energy is achieved in both cases. Nevertheless, for both these compositions, the compounds do not lie on the newly computed convex hull. For all other stoichiometries, the workflow outperforms the AFLOW dictionary method used to build AFLOWlib. At the 1-2-1 stoichiometry, it allows the discovery of a compound on the convex hull, 10.96 meV/atom lower than the one present on AFLOWlib. Note that on the initial AFLOW convex hull, this point is present. The distances from the hull for the other points are summarised in Table 5.9. Note that for all permutations of the 3-1-1 stoichiometry, no compounds are available on AFLOWlib. For the other compositions, the AFLOWlib compounds are beyond 300 meV/atom away from the hull, whereas they are all within 10 meV/atom for the workflow, aside from the 4-1-1 stoichiometry, which is still an order of magnitude closer to the convex hull than AFLOW's lowest energy compound at that stoichiometry. Importantly, these include four new hull points at 1-3-1, 1-1-4, 2-2-1 and 1-2-2. The sixth ternary hull point seen in Figure 5.25a is at the 1-2-3 composition. Note that this specific stoichiometry is probed, since it lay in a region that seemed prone to material stability. The workflow is thus tested here, resulting in the discovery of another convex

hull ternary.

Table 5.9: Table comparing the distance to the convex hull of compounds at all stoichiometries associated with the permutations of 2-2-1, 3-1-1 and 4-1-1 for the workflow (δ_w) and AFLOWlib (δ_{AFLOWlib}) compounds. The three integers give the relative concentration of Mo, Ta and W, respectively.

Stoichiometry	δ_{w} (meV/atom)	δ_{AFLOWlib} (meV/atom)
3-1-1	9.63	N/A
1-3-1	Hull	N/A
1-1-3	1.17	N/A
4-1-1	46.56	320.95
1-4-1	3.25	516.30
1-1-4	Hull	334.16
2-2-1	Hull	880.90
2-1-2	8.50	1032.50
1-2-2	Hull	962.84

These results show that the workflow is successful at increasing the throughput, speed and coverage of the convex hull construction. Many new ternaries are unraveled by the workflow and the lowest energy compounds are almost always lower in energy than those available on AFLOWlib. The only composition for which AFLOW outperforms the workflow is at Mo_2TaW. This compound is, in fact, an important feature of the final convex hull (hull point highlighted in Figure 5.26b). Investigating further, the AFLOWlib candidate is a Heusler compound, displayed in Figure 5.27, next to the minimum energy compound found with the workflow. There are notable structure differences, as there is closer proximity of the Mo atoms in the case of the workflow structure, whereas there are separate layers of each element in AFLOWlib's case. Crucially, the AFLOWlib compound is a *fcc* derivative structure and it is found that such a parent prototype is not obtained from the workflow. This is due to the fact that such a structure is unfavourable to form for all elemental and binary subsystems, which prefer the *bcc* structure. Two limitations are thus revealed. Firstly, that the screening model may be incorrectly biased towards certain configurations. This could be improved by using a better force field. The other is that the ternary candidates suggested are limited by the initial database of parent prototypes. If the latter does not include structures that stabilise ternary compounds, then certain structures are

missed. These two points are addressed in the construction of the M3GNet workflow.

(a) (b)

Figure 5.27: Subfigure (a) a 2×2×2 supercell of the primitive cell of the AFLOWlib minimum energy compound for Mo_2TaW. Subfigure (b) a 2×2×2 supercell of the primitive cell of the workflow minimum energy compound for Mo_2TaW. For both figures, Mo is green, Ta is light blue, and W dark blue.

The constructed convex hull suggests that at low temperatures, and the region with low Mo concentrations is particularly prone to forming stable compounds. At higher Mo concentrations, the 2-1-1 compound heavily impacts the convex hull and it is the only stable compound. A recent study by Sotskov *et al.* [316], investigating the same ternary phase diagram, used a novel crystal structure prediction algorithm, named CSP-on-lattice, that takes advantage of an MLIAP. Within the full ternary system and binary subsystems, three binaries and two novel ternaries were discovered, along with a third ternary close to stability (1 meV/atom away from the convex hull). The novel ternaries were found at 1-1-5 (near stable), 3-3-1 and 1-2-2. Note that the former matches the finding of this study and that the other two stoichiometries were not investigated here, although the 1-1-4 and 2-2-1 phases (closest compositions to the other two) were found to be stable. Interestingly, in the Mo-rich region, no novel ternaries were found, reinforcing the idea that the AFLOWlib 2-1-1 compound is a strong candidate for stability, as it is on the aforementioned authors' convex hull as well. In their study, it is suggested that the equimolar Mo-Ta compounds are good candidates for stability. This is notably supported by the presence of the AFLOWlib 1-1-1 compound on their hull, which is inconsistent with these results. Moreover, it is indicated that the 1-1-2 AFLOWlib compound would be on the hull had their compounds not been added, which contrasts with the initial findings here. Upon comparing enthalpies of formation, for

several AFLOWlib hull points, there are inconsistencies of the order of 2-3 meV/atom. This makes direct additions of the novel ternaries to the convex hull here impossible. It is likely, however, that the 3-3-1 compound is stable with respect to the SNAP convex hull since it lies more than 5.5 meV/atom below it. Overall, there is good agreement in the Mo-rich region and for the points lying close to the Ta-W binary subsystem.

In this section, the results obtained from the SNAP workflow were presented and were shown to be better than those obtained with AFLOW's dictionary method. This is particularly notable for the case of Mo-Ta-W system, which has the most stable ternaries on AFLOWlib. The SNAP model for that workflow was trained entirely on AFLOWlib data and predicted as many as six novel ternaries. It is apparent that this workflow is useful in identifying areas of the ternary phase diagram that are prone to form stable materials. The workflow is run for a total number of candidate compounds across all compositions, on the order of 10^5 in total, and the final convex hull is fully obtained from DFT. There are nevertheless some drawbacks in this methodology. Most notably, the workflow may yield poor predictions at certain compositions. This could be related to the poor modeling of certain compounds or to the lack of diversity in the database of parent prototypes. Such issues are addressed in the next section, where the M3GNet universal force-field is used in place of SNAP.

5.6 M3GNet Workflow: M3GW

5.6.1 Development

Following the publication and data availability of the M3GNet universal MLIAP [163], the workflow is adapted to incorporate the crystal graph neural network as the central screening tool. This is done to increase the accuracy and the reliability of the model. Indeed, M3GNet boasts a 35 meV/atom test MAE on the Materials Project compounds across the full periodic table. From its wide range of training data, including high energy, out-of-equilibrium compounds encountered throughout the relaxation paths that it was trained on, M3GNet is notably designed to perform robust ionic relaxation. It is then ideal to be employed in a high-throughput materials design framework. In this workflow, it is potentially a significant upgrade with respect to a SNAP ensemble, since it is more accurate and can already perform robust relaxation, without requiring

some of the steps used in the original workflow (e.g., cross-validated relaxation and standard deviation filtering). While it is a heavier model with respect to SNAP in terms of number of parameters and model architecture, the featurisation is faster and the fact that it does not rely on an ensemble makes the relaxation process faster than that employed for the SNAP workflow. It is equally a pre-trained model that can be used "out of the box". It therefore potentially offers a more accurate, efficient and robust screening tool.

The new workflow developed is coined M3GW (**M3G**Net **W**orkflow). It is very similar to the previous one, where instead of the SNAP ensemble, M3GNet is substituted in as the relaxation and screening MLIAP. This results in two main sets of changes, one related to the relaxation and screening process, which becomes simpler. The other change relates to the parent prototype selection, which becomes broader as the force-field is now more robust. Regarding the first set of changes, the training step is removed as the model is pre-trained. The ionic relaxation is performed by the single model used and it is not stopped after 5 steps but 50. A stop on the number of steps is still implemented, since it serves to limit the time spent on less important parts of the relaxation process and still drives the structures close to equilibrium. No standard deviation is generated by the model and, hence, the filter based on σ_{cut} is removed. Only the initial prototype volume re-scaling step takes place. The process simply becomes a fixed cell relaxation of each structure followed by an energy ranking of all candidate compounds. This allows for higher throughput since fewer compounds are removed from the screening process by the standard deviation filtering.

Regarding the prototype creation, a more diverse set of compounds is now considered. It is reminded that initially, the binary compounds close to the convex hull were considered as prototypes for the ternaries as these were assumed to possess structures resembling equilibrium ternary structures. This was required for SNAP to be effective as a screening model. This is no longer a requirement with M3GNet. The starting point remains the AFLOWlib binary subsystems database for each ternary. However, rather than only selecting compounds close to the convex hull, the selection is first performed across all different space groups within the binary database. The compounds are first "undecorated", namely, the backbone structures, with no chemical species associated with the constituent atoms, are considered. These are filtered to select only

the unique ones. They are then categorised in space groups. One compound from each space group is then selected. The specific choice is based on the distance from the convex hull, δ, of that structure's associated binary. The structure with the minimum δ is selected. If the number of prototype structures thus generated is above a desired number, the prototype database is considered complete. Otherwise, for each space group, if possible, the second closest structure to the convex hull at each space group is added. This process is repeated until the required number of prototypes is met. This number is set to 50. The final database is, therefore, structurally more diverse than the ones created with the previous workflow.

The M3GW architecture is illustrated in Figure 5.28. It reflects the changes mentioned above. It is also run separately for each individual stoichiometry. The methodology for generating derivative structures from the parent prototypes is the same and the volume scaling is retained. The results using this workflow are presented in the next sections.

5.6.2 Results for Mo-Ta-W

Before testing the workflow on more complex systems, it is deployed to construct the Mo-Ta-W ternary phase diagram. In order to compare directly to the previous results with the SNAP workflow, the same prototype database is used. The prototype generation step described previously is saved for future convex hull constructions. The results for the convex hull obtained is shown in Figure 5.29a, along with the difference between the M3GW deduced convex hull and the one obtained with SNAP (Figure 5.29b). In order to have a closer look at the differences between the two workflows, Figure 5.30 compares the hull values and minimum enthalpies obtained for each workflow. The M3GW-derived hull equally has 6 points at 1-1-1, 2-2-1, 1-2-2, 2-1-2 and the W-heavy 1-1-2 and 1-1-3 compositions. The 2-1-1 (that beats AFLOWlib's point) and the 1-1-4 stoichiometry are very close to hull at 0.64 and 0.14 meV/atom from the convex hull respectively. M3GW finds points closer to equimolarity that are more than 10 meV/atom lower than the SNAP-generated convex hull, bringing the hull down in the central region as well as the W-heavy region. For the two compositions AFLOWlib had lower enthalpy compounds compared to SNAP at 1-1-1 and 2-1-1, M3GW is able to identify novel lower-lying compounds. However, the 2-1-2 compound just undercuts

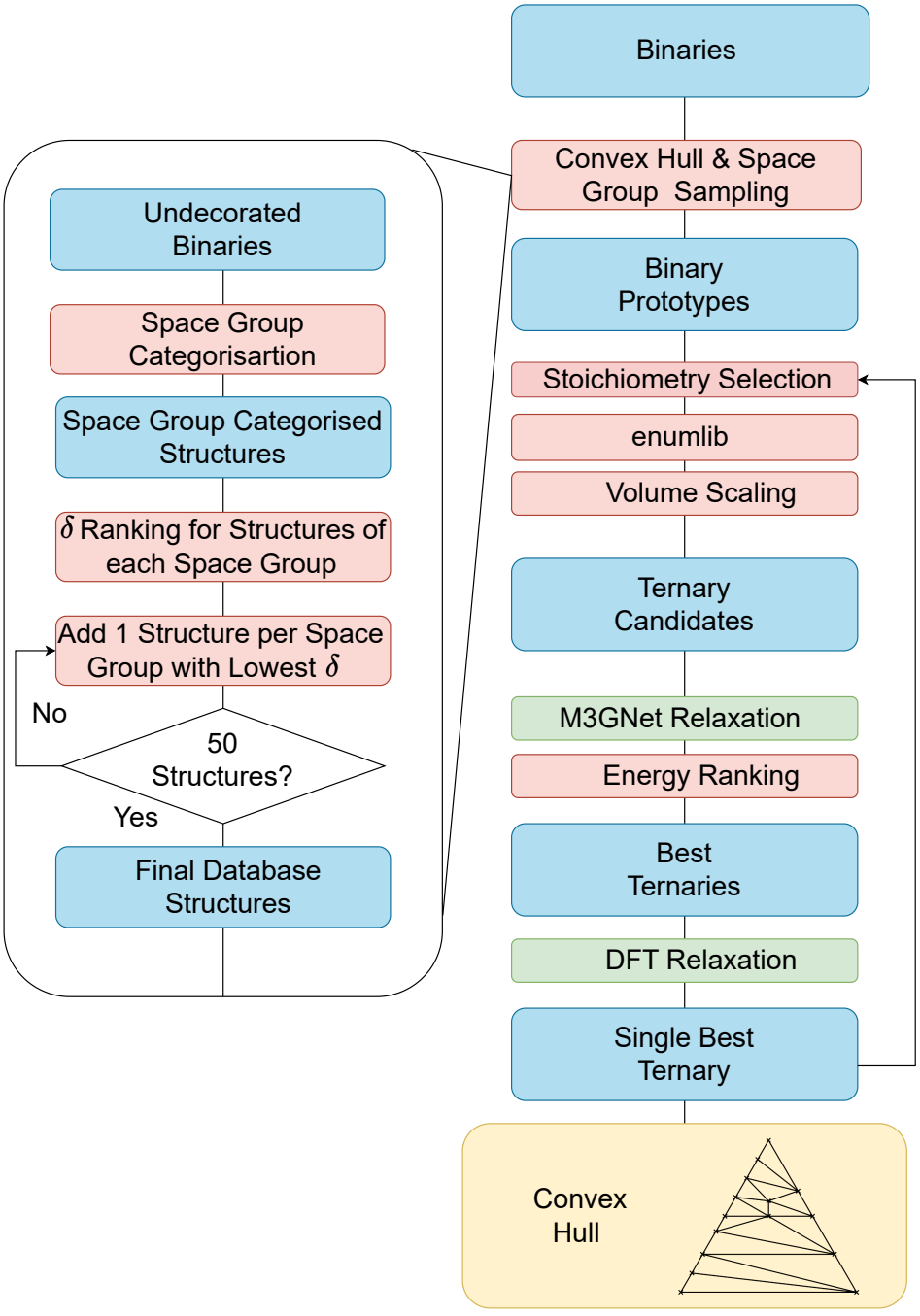
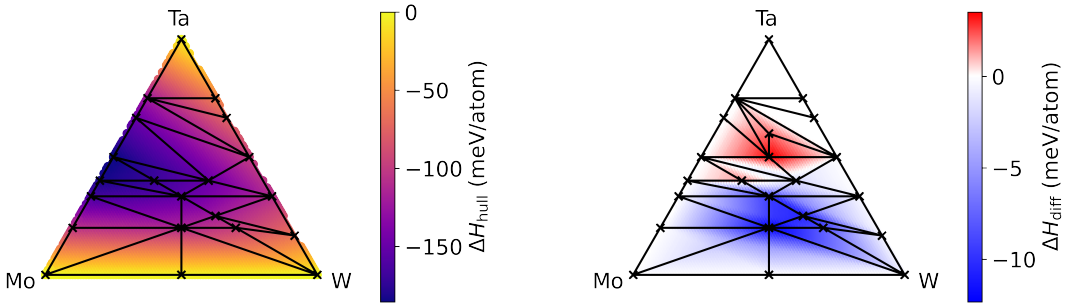


Figure 5.28: Diagram illustrating the structure of M3GW for the construction of ternary convex hulls. The left-hand panel shows the details of the parent prototype generation protocol. The workflow is run for all desired stoichiometries from the Stoichiometry Selection step to the Single Best Ternary step (lowest DFT relaxed compound). The convex hull is constructed from these results.



(a) (b)

Figure 5.29: Panel (a) shows the reconstructed Mo-Ta-W convex hull from the AFLOWlib binary data and the added ternary points given by M3GW. The colour bar gives the enthalpy of formation of the convex hull, the hull points are indicated by black crosses and the tie lines by black lines. Panel (b) shows the difference ΔH_{diff} between the new convex hull and that computed with the result of the previous workflow. Note that the underlying ternary convex hull was calculated by combining the results of both workflows.

the 2-1-1 composition. Nevertheless, like AFLOWlib, M3GW predicts the latter to be close to stability. The only part of the convex hull in which M3GW shows poorer performance and in fact misses two convex hull points is in the Ta-rich corner. At the 1-2-1 and 1-3-1 compositions, M3GW has no hull points and the hull is 3.52 meV/atom and 1.89 meV/atom above the enthalpies of the SNAP discovered compounds. While this energy difference is low, the lacking of any points in this region is problematic and M3GW predicts none of the Ta-rich compounds (1-2-1, 1-3-1 and 1-4-1) to be within less than 5 meV/atom of the hull. However, this still remains well below the 25 meV/atom of potential entropic contributions at room temperature, so that the compounds are still in contention for stability.

The diversity convex hulls uncovered by AFLOWlib, the two workflows implemented in this study, as well as that by Sotskov *et al.* [316], is a consequence of the ability of the Mo, Ta and W to mix across the full compositional range. It is therefore perhaps not surprising that large differences are observed between hulls and that even in regions where no points are on the convex hull (Ta-rich region in Figure 5.29a), the lowest enthalpy compounds are well within $k_{\text{B}}T$ of the convex hull. Based on all the results collected, no specific composition of the phase diagram appears more prone to forming stable phases at low temperatures. Only the Mo-rich corner beyond the 2-1-1

composition appears to have fewer convex hull points across studies. M3GW remains the best methodology out of those tested here, as quantitatively, it drives the biggest changes. In the region where SNAP performs better, the lowest enthalpy compounds are within 9 meV/atom of the SNAP convex hull, while the M3GW hull is never more than 4 meV/atom away. This increased performance, along with the other advantages presented in Subsection 5.6, makes M3GW the superior methodology and is, hence, used for further investigations.

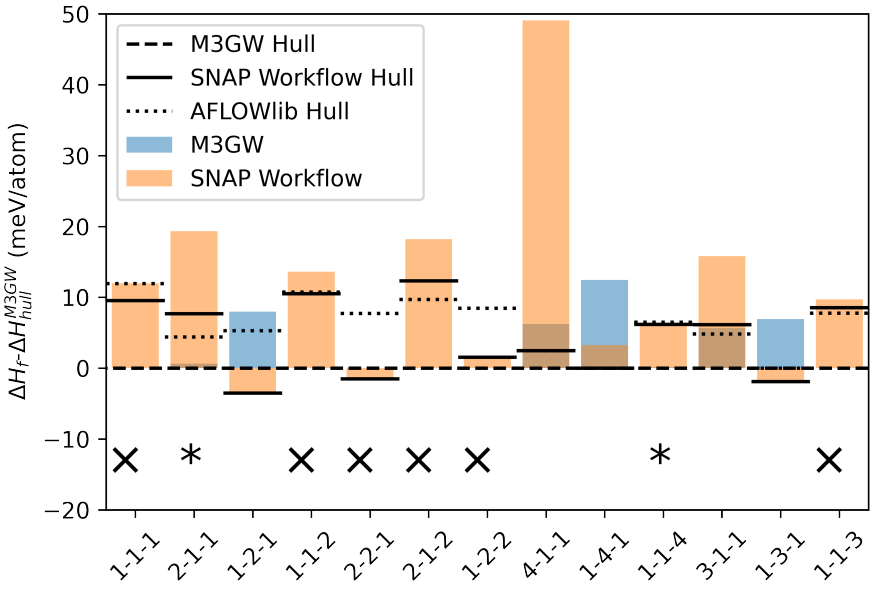


Figure 5.30: Bar plot showing the convex hull and compound minimum enthalpies of formation for different methods with respect to the M3GW convex hull level $\Delta H_{\text{hull}}^{\text{M3GW}}$. This is shown for a range of different stoichiometries. The full dashed line is $\Delta H_{\text{hull}}^{\text{M3GW}}$, set to 0 meV/atom, while the hull levels for AFLOWlib (SNAP workflow) is shown as a dotted (full) line. The minimum enthalpies of formation of compounds at each stoichiometries are shown in blue (orange) for the M3GW (SNAP workflow). Crosses indicate M3GW-found convex hull points and stars indicate points very close to the convex hull ($\delta \leq 1$ meV/atom).

5.7 M3GW Results

5.7.1 Al-Fe-Ni

For the final two sections, ternary systems with potential magnetic properties are investigated. The first one, used as a benchmark, is the Al-Fe-Ni ternary, already presented in Section 5.3.2. The DFT results are first briefly presented here. Regarding the binary subsystems, the experimentally stable Al_5Fe_2 phase should, in principle,

be added to the database retrieved from AFLOWlib, since it is missing. However, it has an unusual structure and it has previously been reported not to be on the DFT convex hull [373]. As a result, it is discarded here. The ternaries, the τ_1 and τ_2 phases, with respective compositions $Al_{10}Fe_{3-x}Ni_{1+x}$ (x between 0 and 2) and $Al_9Fe_{2-x}Ni_x$ (x between 1 and 1.6), are not present on AFLOWlib and are added to the database. AFLOWlib possesses one stable ternary in the Al-rich region, Al_2FeNi. The τ_3 phase is not considered as it is a high-temperature phase with a large unit cell.

Regarding the DFT calculations the same standards as before are used. A plane-wave energy cut-off of 400eV is kept for relaxations, performed for the experimental compounds added and the workflow compounds, and a 600eV cut-off is used for static calculations performed on AFLOWlib compounds and after relaxations. Values for N_{KPPRA} of 6,000 and 12,000 are used for relaxation and static calculations respectively. All calculations performed for this system are spin-polarised, with initial magnetic moments set to 5, 1 and 1 μ_B for Fe, Al, and Ni, respectively.

For the experimental compounds, the structures are taken from Reference [358]. Two backbone structures are used for the τ_1 and τ_2 phases. For the former, calculations are performed for the Al_9FeNi (9-1-1) composition in a 22-atom cell. In this, 4 positions are available for the Fe and Ni atoms. Four structures, associated with the unique permutations of Fe and Ni on those sites, are used for calculations. For the τ_2 phase, the structure is decorated to obtain the $Al_{10}Fe_3Ni$ (10-3-1) and $Al_{10}FeNi_3$ (10-1-3) compositions. The Fe and Ni atoms are swapped in each structure. The cell had a total of 28 atoms. Neither of these are found to be magnetic. The DFT convex hull obtained with AFLOWlib, including the ternary experimental data, is shown in Figure 5.31. The 10-3-1 and 9-1-1 compounds are present on the convex hull, while the 10-1-3 is not.

The workflow is then run for a wider range of compositions than previously. Table 5.10 gives the different stoichiometries sampled, the number of stoichiometries associated by permutation, the maximum cell size per stoichiometry and the total number of candidates obtained for each composition. There is a total of 34 different compositions. In total, 51 parent prototypes are used and 30 structures are sent for DFT relaxation per stoichiometry. This last increase is a reflection of the increase in parent prototypes. The total number of candidate compounds considered is 11,371,718.

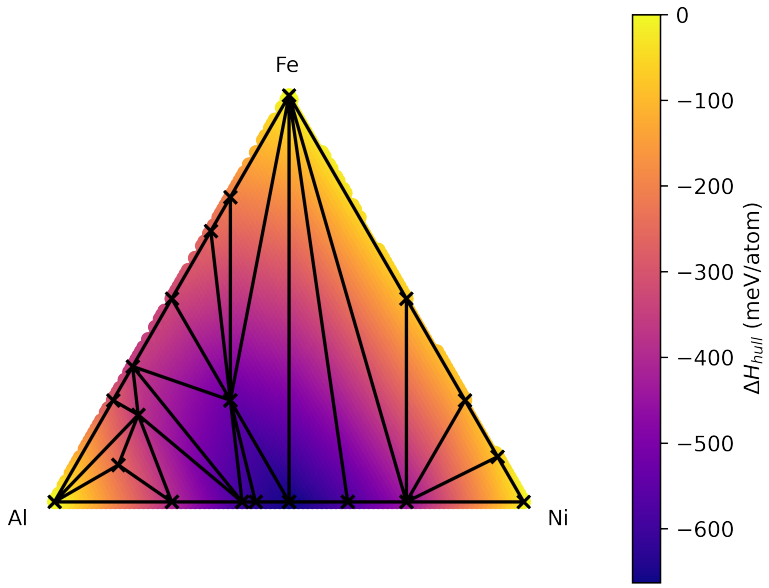


Figure 5.31: DFT Al-Fe-Ni convex hull from the AFLOWlib data and the added ternary points given by the experimental phases τ_1 and τ_2. The colour bar gives the enthalpy of formation of the convex hull, the hull points are indicated by black crosses and the tie lines by black lines.

The convex hull uncovered for Al-Fe-Ni is shown in Figure 5.32. Figure 5.33a shows the difference between the M3GW and the AFLOWlib/experimental hull, along with a plot displaying δ for the lowest energy compounds found at different compositions in Figure 5.33b. Note that for the latter, the 5-1-1 stoichiometry result is removed since it is more than 117.47 meV/atom away from the hull, and its inclusion diminishes the clarity of the plot. M3GW appears to enrich the convex hull through the inclusion of four stoichiometries at 2-1-1, 4-2-1, 3-2-1 and 2-4-1. The first match the AFLOWlib data, but the compound obtained is 8.28 meV/atom above AFLOWlib's. The 4-2-1 and 2-4-1 compounds are undercut by the AFLOWlib convex hull. As a result, the sole finding from the workflow is the compound at 3-2-1, which lowers the hull by 4.91 meV/atom (highlighted by the blue region in Figure 5.33a). The experimental compounds are below the M3GW convex hull by 21.19 meV/atom at 10-3-1 and 6.53 meV/atom at 9-1-1. These points are associated with the most pronounced features of the difference in convex hulls. It is important to point out that the workflow is carried out for magnetic materials, despite the spins not being explicitly taken into account in

Table 5.10: Number of permutations, maximum atoms and ternary prototypes for each stoichiometry.

Stoichiometry	1-1-1	2-1-1	3-1-1	2-2-1	2-2-3
# Permutations	1	3	3	3	3
# Maximum Atoms	9	12	10	10	14
# Ternary Prototypes	13,358	530,064	14,258	33,782	534,890
Stoichiometry	3-2-1	3-3-1	4-1-1	4-2-1	5-1-1
# Permutations	6	3	3	6	3
# Maximum Atoms	12	14	12	14	14
# Ternary Prototypes	1,009	935,620	119,736	505,567	71,746

the M3GNet.

Regarding the performance of M3GW, fewer points are found compared to the previous phase diagram. At 2-1-1, the minimum enthalpy compound is missed. Nevertheless, it is present on the convex hull, but enables other points to be on it as well. It also does not find any low enthalpy compounds for the very richest Al compounds, at 3-1-1, 4-1-1, and 5-1-1. The last stoichiometry is notably very far away. This area is closest to the experimentally known ternary intermetallic compounds and it would be desired that such stoichiometries be highlighted by the workflow. It would not have been possible to find the exact ternaries considering the choice of prototype compounds and the sizes of the cells used compared to the experimental structures. The method is affected by the high dependence of the number of candidates on the composition. There is a large disparity (order of 10^3) between stoichiometries, such as the 3-2-1 and 2-1-1 ones, as indicated in Table 5.10. This is due to the number of available parent prototypes and their symmetries. This sheds light on an important feature of the workflow. It appears that this method is not suitable for finding specific intermetallic compounds, since these may have exotic structures, different to those of the binaries. A global structure search method may then be more suitable. Instead, the workflow seems better at highlighting areas of fertility: regions of compositional space, where compounds are found to lie at or close to the convex hull.

The hull points discovered in Figure 5.32, as well as the areas for which compounds lie close to the convex hull in Figure 5.33, indicate regions of stability. These correspond to the areas in the elemental-rich corners, most notably the Al-rich one, which possess

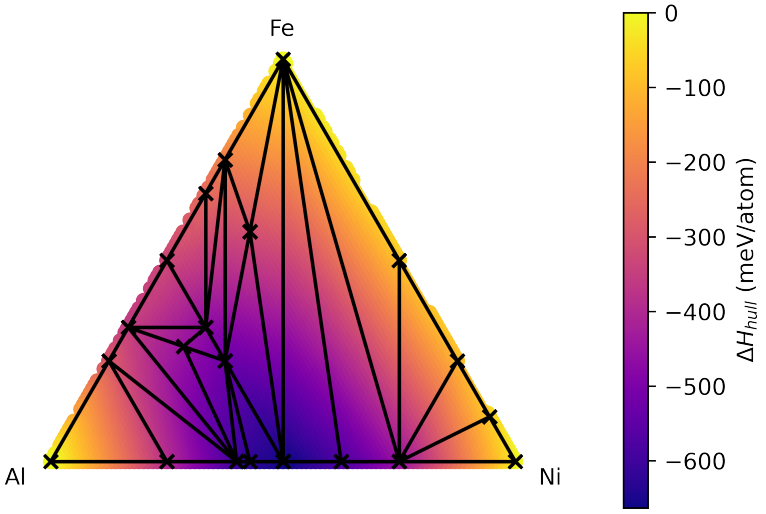
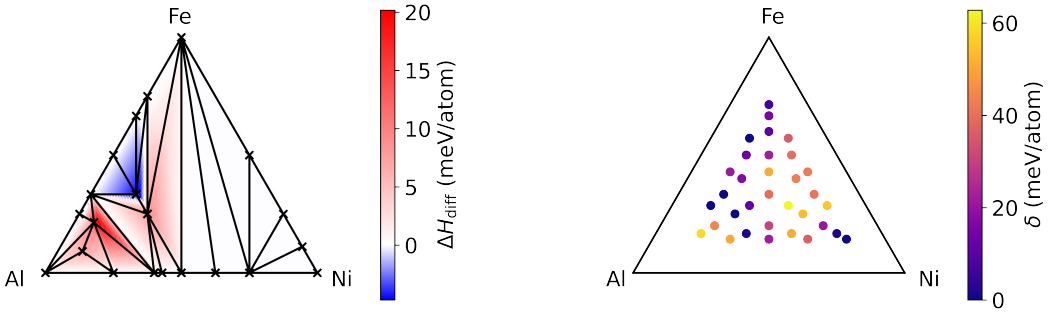


Figure 5.32: DFT Al-Fe-Ni convex hull obtained from M3GW. The colour bar gives the enthalpy of formation of the convex hull, the hull points are indicated by black crosses and the tie lines by black lines.

a range of stability across the 3-2-1, 2-1-1 and 3-1-2 compositions. In the central region, as well as the area with low Al concentrations and high Ni concentrations, δ is higher than 25 meV/atom across all stoichiometries. This is broadly in line with the experimental phase diagram that displays an immiscibility region at the same area, between the β and γ phases. Aside from the intermetallics, the Al-rich region shows no solid-state solution behaviour. In conclusion, M3GW appears to highlight regions in which solid-state solutions may form.

5.7.2 Bi-Fe-X

The final systems for which M3GW was tested are Bi-Fe-X, with X = Ta, Zr. Note that the Bi-Fe-Ta results were entirely run by myself and the Bi-Fe-Zr ones by Michail Minotakis. These systems are chosen since they have previously known little investigation. They are also composed of Fe and Bi, which can provide potential stable compounds with high magnetisation, T_C, and magnetic crystalline anisotropy. The experimental phase diagrams and convex hulls from online repositories presented no stable ternary phases. The Bi-Ta and the Bi-Fe binary subsystem, common to both ternary systems, form no known stable phases. The other three binary subsystems



(a) (b)

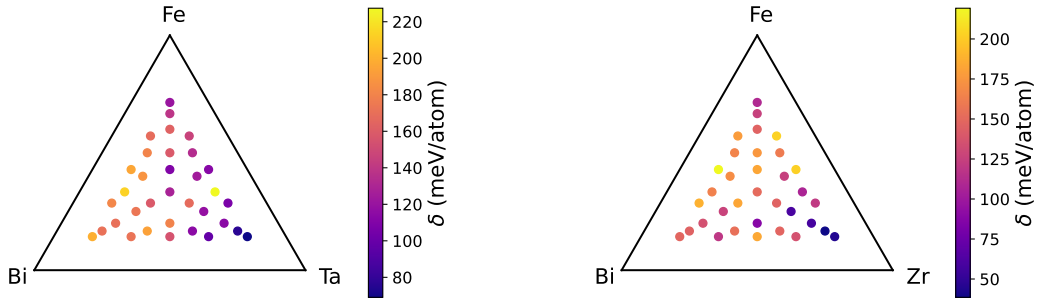
Figure 5.33: The left plot displays the difference between the AFLOWlib+experimentally determined convex hull and that obtained with M3GW for Al-Fe-Ni. ΔH_{diff} is the difference between the hulls. Positive red regions indicate that the AFLOWlib+experimental convex hull regions are lower, whereas the negative blue regions show the regions lower for M3GW. The right plot shows δ, the distance from the M3GW-determined convex hull of the lowest enthalpy compounds found with M3GW at different ternary compositions.

form stable binary phases.

The M3GW workflow is run for both systems. The final number of structures screened for *ab-initio* calculations is capped at 30 per stoichiometry. However, to increase the diversity of compounds passed to DFT, the lowest M3GNet-energy compounds for *every parent prototype* are forced to be selected. Thus, all the parent prototypes[3] (binary structures) are represented by at least one candidate in the final pool of ternary compounds. If this pool contained less than 30 structures, the second lowest energy compounds for each prototype are added. This is repeated one last time if the number of candidates does not reach 30. The DFT calculations are spin polarised and were initialised with magnetisations of 1.0, 5.0 and 1.0 μ_B for Bi, Fe and Ta/Zr respectively. The same 34 stoichiometries as for Al-Fe-Ni are probed. The number of candidate structures in M3GW is of the order of 10^7 and of the order of 10^3 ($\sim 30\times34$) for the DFT calculations.

After running the workflow, no stable ternaries are discovered for either ternary system. The distance from the convex hull δ of the lowest enthalpy compounds for each stoichiometry are shown in Figure 5.34. All values are positive, the lowest distance is 69.05 meV/atom for $BiFeTa_5$ (1-1-5) for Bi-Fe-Ta and 38.56 meV/atom for $BiFeZr_4$

[3]All the ones that are compatible with a given stoichiometry.



(a) (b)

Figure 5.34: Distance from the convex hull δ of the lowest enthalpy compounds found at the different compositions sampled by M3GW for Bi-Fe-Ta (left) and Bi-Fe-Zr (right). The colour scale marks the distance from the hull, with dark blue being closest and yellow being furthest.

(1-1-4) for Bi-Fe-Zr. For the former, the average distance from the convex hull of the minimum energy compound, over all compositions probed, is of 150.80 meV/atom. For three Ta-rich compositions, 1-1-5, 1-1-4 and 1-2-4, values of δ below 100 meV/atom are found. These lie close to or above $3k_{\mathrm{B}}T$ ($k_{\mathrm{B}}T \sim 25$ meV/atom) from the convex hull and are very unlikely to form stable phases. The compositions closest to the Fe-Ta subsystem are also lower than other regions, consistent with this binary being the only miscible system. For Bi-Fe-Zr, a similar trend is observed, with the average distance to the hull being 141.32 meV/atom, but the Zr-rich corner presents several compounds closer to the hull. Indeed, the 1-1-2, 1-1-3, 1-1-4 and 1-1-5 have values of δ of -60.09, -58.45, -38.56 and -44.42 meV/atom respectively. The latter two are therefore within $2k_{\mathrm{B}}T$. It is perhaps not surprising that this region presents compounds closer to the convex hull, since Zr forms binaries with both other elements. Only one other compound at 2-1-2 is less than 100 meV/atom away from the hull, at 87.98 meV/atom.

The results here are consistent with previous findings of no stable ternary phases. Some DFT relaxations are notably difficult and costly to perform, indicating that certain structures are, in fact, far from equilibrium. This can be in part attributed to the large discrepancy in ionic radius between Bi and the other elements. The results are consistent with M3GW identifying areas of the phase diagram susceptible to phase formation and none have been found here. Final remarks are presented in the conclusions of the chapter.

5.8 Conclusions

In this chapter, it was explained how two MLIAPs, SNAP and M3GNet, were deployed in a high-throughput workflow aimed at screening stable materials. The process of enumerating all unique derivative structures from a parent prototype was reviewed and was employed to generate candidate ternary compounds. The crystal graph neural network M3GNet was also presented as a universal MLIAP. The phase diagrams of four relevant systems, Mo-Ta-W, Al-Fe-Ni, Bi-Fe-X (X=Ta,Zr) were also presented.

The workflow developed revolved around using available data on binaries to construct ternary convex hulls. Binaries were used first as a pool of prototypes, whose structures were decorated to form derivative structures with three elements. The binary compounds, along with their energies, also form the training database of the ensemble SNAP model, presented in the previous chapter. This was used as a screening tool for the binary prototypes. The so-constructed workflow only makes use of binary data to build ternaries.

Several steps were added to the workflow in order to increase its robustness. These included scaling the volume, performing a SNAP-driven relaxation, cross-validating the energy predictions, and screening the standard deviation of the ensemble predictions to select only high-fidelity compounds. The workflow was then run on Ag-Au-Cu to identify the lowest energy compounds at several ternary stoichiometries. It was demonstrated that the workflow can adequately rank the compounds with respect to energy and identify the low enthalpy compounds. A set of the most promising compounds were then relaxed with DFT to construct a ternary convex hull. The workflow was run on Ag-Au-Cu and two stable compounds, $AgAu_2Cu$ and $AgAu_3Cu$, were identified. The process was repeated for Mo-Ta-W. In this case, the SNAP ensemble model was entirely trained on AFLOWlib data. The workflow identified 6 stable phases and only missed the Mo_2TaW stable phase present on AFLOWlib. The convex hull was consistent with the finite temperature experimental phase diagrams. Overall, the workflow was successful at discovering novel compounds and enhancing the ternary convex hull.

The workflow is enhanced in robustness, accuracy, and throughput by integrating M3GNet as a screening tool instead of SNAP. The new workflow, M3GW, was able to deal with a more diverse set of structures and performed reliable relaxations, without

the requirement of any additional steps. M3GW is first tested on Mo-Ta-W and found further stable phases, beating all the AFLOWlib compounds. Only, the Ta-rich corner is better described with the previous workflow. The potentially magnetic ternary phases of Al-Fe-Ni and Bi-Fe-X (X = Ta,Zr) were then studied. For the former, similar results to AFLOWlib were found, with the Al-rich corner providing several stable phases. It is established that the workflow is able to determine regions of the convex hull fertile for the formation of stable compounds, rather than reliably identifying novel intermetallics with a specific structure. Finally, for the other two phase diagrams, no new discoveries were made. This is in agreement with the experimental phase diagrams that contain no ternary phases.

6

Conclusions and Future Work

This work inscribes itself in the field of inverse materials design, in which novel compounds are conceived virtually, with the aim of obtaining tailored properties. More specifically, the goal is to use high-throughput, first principles calculations to generate novel stable materials. In this traditional approach, it is necessary to calculate the DFT energies of many candidate compounds in order to construct the convex hull and identify which contestants lie on the T=0K phase diagram and are thus stable. The work presented in this thesis establishes to what extent the process may be accelerated using machine learning, specifically in the form of MLIAPs, for ternary systems. The two potentials tested, SNAP and M3GNet, are in fact all trained on pre-existing data, present on large online materials repositories.

In Chapter 2, DFT was presented as the high-fidelity model to accurately describe the electronic structure of materials and compute their energies (along with atomic forces and stress tensor elements), which are the central quantities required to obtain enthalpies of formation and hence predict compound stability. The work of Hohenberg, Kohn and Sham was summarised to show how the many-body interacting electron system problem could be recast in terms of the electron density. This resulted in the formulation of the self-consistent field Kohn-Sham scheme. Since the VASP package is used for this work, the associated methodology used to solve the KS equations was presented, with particular focus on the projector augmented wave pseudopotential formalism. Afterward, the conjugate-gradient algorithm, used to perform structural and ionic relaxation, was described, as it is pivotal in finding the equilibrium configuration of a compound.

In Chapter 3, machine learning and MLIAPs were extensively introduced. The latter was defined as the set of methods employed to automatically (without explicit human intervention) unravel and learn patterns from data. The emphasis of the section was on supervised learning and involved the description of the importance of a judicious selection of data, feature vectors, and algorithms in machine learning pipelines. Ridge regression and neural networks were presented as relevant examples of two models that can be used to map the feature vectors to the target property. This led to the presentation of the recently developed class of interatomic potentials called MLIAPs. These models aim to represent the PES of a system at the level of DFT, based on local, symmetry-invariant atomic fingerprints and a machine learning algorithm. It is stressed that these potentials have parameters that are tuned through a learning process using a database of compounds and corresponding energies (forces, stress tensors), which must effectively sample the region of the PES on which the MLIAP is intended to be used. Two specific models, namely NNP and SNAP, are described in further detail, with a particular focus on the descriptors used. Finally, the main successful applications of MLIAPs are presented, and are centered around performing simulations over large lengthscales and timescales (MD) or on a large number of structures (high-throughput materials screening).

Chapter 4 presented the first set of results related to this work. The goal was to train a SNAP model on binary compounds in order to make energy predictions and hence screen ternary intermetallics. The relevant theory on thermodynamic stability was first expounded, followed by the AFLOW environment and standards for DFT high-throughput calculations. The Ag-Au-Cu system served as a test alloy for the study and the subsystem binary and full ternary experimental phase diagrams were given. The first set of results focused on the training methodology of the SNAP ensemble, for which the training data points were all binary compounds of the Ag-Au, Ag-Cu, and Ag-Cu systems present on AFLOWlib (the DFT energies were re-computed), amounting to more than 600 structures. It was shown that the so-trained interatomic potentials could be used to predict the energies of *fully relaxed* ternary compounds with errors below 30 meV/atom. Predictions of the bulk moduli could also be made with relative errors of no more than 15%. The two main limitations of the devised model were its inability to accurately predict the energies of out-of-equilibrium compounds (albeit

with an adequate energy ranking), and to fail to perform full relaxations. This restricts the use of the model to equilibrium structures. It was argued that the reason for this is the lack of configurations with large forces and hence the absence of chemical environments associated with the high-energy regions of the PES. The results were shown to hold when switching ternary systems, as exemplified by the Mo-Pt-Ti and Cd-Hf-Rh systems, for which AFLOWlib energy data were used for the MLIAP training. In a final part, several error metrics were tested. These were $d_{\min}$, the Euclidian distance of a chemical environment of the test set to the training set ones, γ, the extrapolation grade, and σ the ensemble standard deviation on predictions. The first was shown to be inadequate for identifying high-error test compounds, whereas the other two, especially when being species-independent, performed well. σ was shown to be the most reliable one. In conclusion, it was shown how an ensemble of SNAP models trained on pre-computed binary data could be used to screen equilibrium ternary compounds and identify non-equilibrium ones.

In Chapter 5, workflows were devised to make use of MLIAPs to accelerate the construction of ternary convex hulls. Aside from a screening model, this required candidate compounds, susceptible of lying close to the hull. The structures of binary compounds were chosen to fill this role. The ENUMLIB algorithm was introduced as the algorithm of choice to generate the ternary derivative structures from the binary prototypes. Two screening tools were tested, the ensemble SNAP model previously developed and the M3GNet universal force-field. The mathematical and algorithmic architectures of the latter were presented. While Ag-Au-Cu was used as a test system for development, the ternary systems Mo-Ta-W, Al-Fe-Ni, and Bi-Fe-X (X = Ta, Zr) were used as a playground in an attempt to identify novel stable phases. The associated phase diagrams and available convex hulls for these and their binary systems were given.

The initial workflow developed used the ensemble SNAP model as a screening tool and the steps in the development process were detailed. Since the candidate compounds were based on the structures of binary alloys, the latter tended to be close to the equilibrium structures. The energy predictions of the MLIAPs were therefore adequate for screening the candidates most likely to be stable. Nevertheless, several steps were incorporated to increase the robustness of the predictions. These included volume scaling of the structures according to a Vegard's-like law, prematurely aborted ionic

relaxations driven by SNAP, cross-validated predictions from the SNAP ensemble, and a cut-off on the permitted standard deviation of the predictions. Following this pipeline, the compounds were ranked with increasing energies, and the first ones were selected as the most promising. Full DFT relaxation was performed on them. This workflow was run for fixed stoichiometries. Thus the construction of the final convex hull relied on runs at a range of different compositions. The final T=0K phase diagram is of first principles level and all the heavy DFT computational effort is focused on the best candidates. This was run for Ag-Au-Cu and Mo-Ta-W and each case revealed novel phases, not present on the AFLOWlib repository. For the latter, a single point was missed, whereas one was undercut as 6 new stable compounds were identified. This shows that a SNAP-centered workflow is capable of improving the completeness of ternary convex hulls.

The focus was then shifted to a different, recently published MLIAP, M3GNet. This crystal graph neural network was also trained on pre-existing data, this time on the Materials Project, but included data across all phases and systems available, including the relaxation path data. The inclusion of high-energy structures enables the model to perform accurate structural and ionic relaxations across a huge range of compounds and compositions. Using this model changed the workflow, as steps to increase robustness were no longer required and a larger range of parent binary prototypes could be used. The stoichiometry-dependent predictions therefore only required an ionic relaxation and an energy prediction on the relaxed structure for each candidate. The final convex hull is again constructed from the DFT energies of the best candidates selected by M3GNet. The first results on Mo-Ta-W showed better performance than AFLOW over the full compositional range. With respect to the SNAP workflow, a deeper convex hull is found in all regions aside from the Ta-heavy one, in which the lowest energy compounds discovered are systematically worse. This points to a potential systematic bias in the M3GNet model. However, overall the predictions are better and the workflow is more efficient. M3GW was deployed for three other systems, which included magnetic elements. In the case of Al-Fe-Ni, the convex hull obtained agreed with the AFLOWlib one and the experimental, finite temperature phase diagram, identifying the Al-rich corner as being fertile for material stability. Regarding the Bi-Fe-X systems, no novel compounds or areas likely to contain stable compounds were

indentified, in agreement with experiment. In conclusion, the workflows devised help identify novel compounds on the ternary phase diagrams by screening a number of candidates on the order of 10^6-10^7. They seem efficient at identifying regions in which material stability is the most likely and produce DFT-quality convex hulls.

Overall, work in this thesis displayed how it is possible to use machine learning to increase the throughput in the search for novel ternary compounds, fitting within the third modality of inverse materials design. This is possible by using recycled, already available data, and pre-trained models. The future direction of this work can be broken down into different time horizons. In the short term, M3GW can be deployed in its present form to probe the convex hulls of more ternary systems, with a more judicious choice of the three constituent elements. This would be for a system where there are known binaries and therefore for miscible species. It would also be for a poorly explored system, ideally made of three transition metals, with one magnetic and one heavy element, if magnetic properties are desired. The selection of such elements could even be done using data mining techniques, in full inverse design fashion.

Over a longer time horizon, the workflow would be adapted for higher robustness, such that a higher diversity of prototypes could be used and for increased accuracy, to provide better screening. To address the first matter, methods to evaluate uncertainty in the predictions could be incorporated, to first identify when predictions are likely to be less reliable. This could serve to find compounds that could help train a more accurate model, in an active learning scheme. The issue would be that M3GNet is too large a model to be trained or fine-tuned on the fly (several times during a single workflow). This could only be feasible once a large enough database was calculated (at least 10^4 compounds), after runs of M3GW across many ternary convex hulls. The problematic compounds could however be used as a training dataset for a specialised MLIAP, such as SNAP, that could work in conjunction with M3GNet. The latter would be a general relaxation and rough screening model, while the other could screen the finer details. Alternatively, M3GNet could be fine-tuned before a M3GW run for a given system, with specific data available for that system. What risks to limit the accuracy of universal screening models remains the lack of extensive, relevant, and good-quality data. M3GNet is already trained on one of the largest materials repositories (Materials Project with 10^5 compounds including relaxation path structures),

and it is unlikely enough data could be generated for orders of magnitude much higher than 10^6 (AFLOWlib is of this size currently). It seems difficult to envisage a single, general, DFT-accurate model being enough because of the lack of data.

Unrelated models that have been trained on much larger databases are Large Language Models (LLMs). Their predictive power has already been proved to be immense and it is still unknown to what extent they can be applied. A recent investigation has notably shown them to be useful in mathematical optimisation problems (e.g. linear regression) with the right prompting, despite not being trained specifically for this [374]. Following these lines, perhaps these models, likely to be the largest ML models in the coming decade, and that dwarf the size of any material-specific models, may be used in certain inverse materials design pipelines.

Appendix A

List of Publications

- Krenzer, G., Klarbring, J., Tolborg, K., Rossignol, H., McCluskey, A. R., Morgan, B. J. & Walsh, A. Nature of the superionic phase transition of lithium nitride from machine learning force fields. *Chemistry of Materials* **35,** 6133–6140 (2023).

- Minotakis, M., Rossignol, H., Cobelli, M. & Sanvito, S. Machine-learning surrogate model for accelerating the search of stable ternary alloys. *Physical Review Materials* **7,** 093802 (2023).

- Rossignol, H., Minotakis, M., Cobelli, M. & Sanvito, S. Machine-learning-assisted construction of ternary convex hull diagrams. *arXiv preprint arXiv:2308.15907 [cond-mat.mtrl-sci]* (2023).

Bibliography

1. Will, K. *The $150 Million Machine Keeping Moore's Law Alive* `https://www.wired.com/story/asml-extreme-ultraviolet-lithography-chips-moores-law/` (2023).
2. Chung, D. D. *Carbon Materials: Science and Applications* (World Scientific, Singapore, 2019).
3. Beaumont, P. W., Soutis, C. & Hodzic, A. *The Structural Integrity of Carbon Fiber Composites: Fifty Years of Progress and Achievement of the Science, Development, and Applications* (Springer, Cham, 2016).
4. Xie, Z. *Polyacrylonitrile Nanofibre Yarn; Electrospinning and their Post-Drawing Behaviour* PhD thesis (Deakin University, Melbourne, 2013).
5. Srinivasa, V. *et al.* Fracture morphology of carbon fiber reinforced plastic composite laminates. *Materials Research* **13,** 417–424 (2010).
6. De Goeij, W., Van Tooren, M. & Beukers, A. Implementation of bending-torsion coupling in the design of a wind-turbine rotor-blade. *Applied Energy* **63,** 191–207 (1999).
7. Babuska, P. F. *Bend-Twist Coupled Carbon-Fiber Laminate Beams: Fundamental Behavior and Applications* PhD thesis (University of Washington, Washington, 2017).
8. Falkowicz, K. Validation of extension-bending and extension-twisting coupled laminates in elastic element. *Advances in Science and Technology. Research Journal* **17,** 309–319 (2023).

9. Ashwill, T. Passive load control for large wind turbines in *51st AIAA/ASME/ ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference* (Orlando, 2010).

10. Fedorov, V. *Bend-Twist Coupling Effects in Wind Turbine Blades* PhD thesis (Technical University of Denmark, Lyngby, 2012).

11. De Frías López, R. *A 3D Finite Beam Element for the Modelling of Composite Wind Turbine Wings* (Royal Institute of Technology, Stockholm, 2013).

12. Brower, M. *Wind Resource Assessment: A Practical Guide to Developing a Wind Project* (John Wiley & Sons, Hoboken, 2012).

13. Zunger, A. Inverse design in search of materials with target functionalities. *Nature Reviews Chemistry* **2,** 0121 (2018).

14. Burke, K. & Wagner, L. O. DFT in a nutshell. *International Journal of Quantum Chemistry* **113,** 96–101 (2013).

15. Lejaeghere, K. *et al.* Reproducibility in density functional theory calculations of solids. *Science* **351,** aad3000 (2016).

16. Bacon, R. & Moses, C. T. Carbon fibers, from light bulbs to outer space in *High Performance Polymers: Their Origin and Development: Proceedings of the Symposium on the History of High Performance Polymers* (New York, 1986).

17. Gorss, J. *American Chemical Society National Historic Chemical Landmarks. High Performance Carbon* `http://www.acs.org/content/acs/en/education/whatischemistry/landmarks/carbonfibers.html` (2023).

18. Sanvito, S. *et al.* Accelerated discovery of new magnets in the Heusler alloy family. *Science Advances* **3,** e1602241 (2017).

19. Curtarolo, S. *et al.* AFLOWLIB.ORG: a distributed materials properties repository from high-throughput *ab initio* calculations. *Computational Materials Science* **58,** 227–235 (2012).

20. Jain, A. *et al.* Commentary: the materials project: a materials genome approach to accelerating materials innovation. *APL Materials* **1,** 011002 (2013).

21. Saal, J. E., Kirklin, S., Aykol, M., Meredig, B. & Wolverton, C. Materials design and discovery with high-throughput density functional theory: the open quantum materials database (OQMD). *JOM* **65,** 1501–1509 (2013).

22. Gražulis, S. *et al.* Crystallography open database – an open-access collection of crystal structures. *Journal of Applied Crystallography* **42,** 726–729 (2009).

23. Belsky, A., Hellenbrandt, M., Karen, V. L. & Luksch, P. New developments in the inorganic crystal structure database (ICSD): accessibility in support of materials research and design. *Acta Crystallographica Section B: Structural Science* **58,** 364–369 (2002).

24. Ashton, M., Paul, J., Sinnott, S. B. & Hennig, R. G. Topology-scaling identification of layered solids and stable exfoliated 2D materials. *Physical Review Letters* **118,** 106101 (2017).

25. Faber, F. A., Lindmaa, A., Von Lilienfeld, O. A. & Armiento, R. Machine learning energies of 2 million elpasolite (ABC_2D_6) crystals. *Physical Review Letters* **117,** 135502 (2016).

26. Bartel, C. J. *et al.* New tolerance factor to predict the stability of perovskite oxides and halides. *Science Advances* **5,** eaav0693 (2019).

27. Isayev, O. *et al.* Universal fragment descriptors for predicting properties of inorganic crystals. *Nature Communications* **8,** 15679 (2017).

28. Thompson, A. P., Swiler, L. P., Trott, C. R., Foiles, S. M. & Tucker, G. J. Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials. *Journal of Computational Physics* **285,** 316–330 (2015).

29. Krenzer, G. *et al.* Nature of the superionic phase transition of lithium nitride from machine learning force fields. *Chemistry of Materials* **35,** 6133–6140 (2023).

30. Kresse, G. & Furthmüller, J. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Computational Materials Science* **6,** 15–50 (1996).

31. Minotakis, M., Rossignol, H., Cobelli, M. & Sanvito, S. Machine-learning surrogate model for accelerating the search of stable ternary alloys. *Physical Review Materials* **7,** 093802 (2023).

32. Rossignol, H., Minotakis, M., Cobelli, M. & Sanvito, S. Machine-learning-assisted construction of ternary convex hull diagrams. *arXiv preprint arXiv:2308.15907 [cond-mat.mtrl-sci]* (2023).

33. Martin, R. M. *Electronic Structure: Basic Theory and Practical Methods* (Cambridge University Press, Cambridge, 2020).

34. Born, M. & Oppenheimer J, R. Zur quantentheorie de molekeln. *Annalen der Physik* **84,** 457 (1927).

35. Hohenberg, P. & Kohn, W. Inhomogeneous electron gas. *Physical Review* **136,** B864 (1964).

36. Levy, M. Universal variational functionals of electron densities, first-order density matrices, and natural spin-orbitals and solution of the v-representability problem. *Proceedings of the National Academy of Sciences* **76,** 6062–6065 (1979).

37. Levy, M. Electron densities in search of Hamiltonians. *Physical Review A* **26,** 1200 (1982).

38. Lieb, E. H. Density functionals for Coulomb systems. *International Journal of Quantum Chemistry* **24,** 243–277 (1983).

39. Kohn, W. & Sham, L. J. Self-consistent equations including exchange and correlation effects. *Physical Review* **140,** A1133–A1138 (1965).

40. Dirac, P. A. M. Note on exchange phenomena in the Thomas atom. *Mathematical Proceedings of the Cambridge Philosophical Society* **26,** 376–385 (1930).

41. Ortiz, G. & Ballone, P. Correlation energy, structure factor, radial distribution function, and momentum distribution of the spin-polarized uniform electron gas. *Physical Review B* **50,** 1391 (1994).

42. Perdew, J. P., Burke, K. & Ernzerhof, M. Generalized gradient approximation made simple. *Physical Review Letters* **77,** 3865 (1996).

43. Hellman, H. Einführung in die quantenchemie. *Franz Deuticke, Leipzig* **285** (1937).

44. Feynman, R. P. Forces in molecules. *Physical Review* **56,** 340–343 (1939).

45. Nielsen, O. H. & Martin, R. M. First-principles calculation of stress. *Physical Review Letters* **50,** 697–700 (1983).

46. Nielsen, O. H. & Martin, R. M. Quantum-mechanical theory of stress and force. *Physical Review B* **32,** 3780–3791 (1985).

47. Birch, F. Finite elastic strain of cubic crystals. *Physical Review* **71,** 809 (1947).

48. Murnaghan, F. D. The compressibility of media under extreme pressures. *Proceedings of the National Academy of Sciences* **30,** 244–247 (1944).

49. Kresse, G. & Hafner, J. *Ab initio* Molecular dynamics for liquid metals. *Physical Review B* **47,** 558 (1993).

50. Kresse, G. & Furthmüller, J. Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set. *Physical Review B* **54,** 11169 (1996).

51. Kresse, G. & Joubert, D. From ultrasoft pseudopotentials to the projector augmented-wave method. *Physical Review B* **59,** 1758 (1999).

52. Ashcroft, N. W. & Mermin, N. D. *Solid State Physics* (Thomson Learning Inc., London, 1976).

53. Monkhorst, H. J. & Pack, J. D. Special points for Brillouin-zone integrations. *Physical Review B* **13,** 5188–5192 (1976).

54. Hamann, D. R., Schlüter, M. & Chiang, C. Norm-conserving pseudopotentials. *Physical Review Letters* **43,** 1494–1497 (1979).

55. Vanderbilt, D. Soft self-consistent pseudopotentials in a generalized eigenvalue formalism. *Physical Review B* **41,** 7892 (1990).

56. Rostgaard, C. The projector augmented-wave method. *arXiv pre- print arXiv:0910.1921 [cond-mat.mtrl-sci]* (2009).

57. Wood, D. M. & Zunger, A. A new method for diagonalising large matrices. *Journal of Physics A: Mathematical and General* **18,** 1343 (1985).

58. Pulay, P. Convergence acceleration of iterative sequences. the case of scf iteration. *Chemical Physics Letters* **73,** 393–398 (1980).

59. Wilson, S. & Diercksen, G. H. *Methods in Computational Molecular Physics* (Springer Science & Business Media, Bad Windsheim, 1992).

60. Moler, C. & Shavitt, I. *Numerical Algorithms in Chemistry: Algebraic Methods: Report on the Workshop* tech. rep. (Lawrence Berkeley Lab., California University, Berkeley, 1978).

61. Broyden, C. G. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation* **19,** 577–593 (1965).

62. Jacob, C. R. & Reiher, M. Spin in density-functional theory. *International Journal of Quantum Chemistry* **112,** 3661–3684 (2012).

63. Press, W. H. *Numerical Recipes Third Edition: The Art of Scientific Computing* (Cambridge University Press, New York, 2007).

64. Nocedal, J. & Wright, S. J. *Numerical Optimization* (Springer, New York, 1999).

65. Barlow, H. B. Unsupervised learning. *Neural Computation* **1,** 295–311 (1989).

66. Sutton, R. S. & Barto, A. G. *Reinforcement Learning: An Introduction* (MIT press, Cambridge, USA, 2018).

67. Hastie, T., Tibshirani, R., Friedman, J. H. & Friedman, J. H. *The Elements of Statistical Learning: Data mining, Inference, and Prediction* (Springer, 2009).

68. LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **86,** 2278–2324 (1998).

69. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Communications of the ACM* **60,** 84–90 (2017).

70. Vaswani, A. *et al.* Attention is all you need in *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, 2017).

71. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N. & Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics in *Proceedings of the 32nd International Conference on Machine Learning* (Lille, 2015).

72. LeCun, Y. 1.1 Deep learning hardware: past, present, and future in *2019 IEEE International Solid-State Circuits Conference-(ISSCC)* (San Francisco, 2019).

73. Giattino, C., Mathieu, E., Samborska, V., Broden, J. & Roser, M. Artificial intelligence. *Our World in Data.* https://ourworldindata.org/artificial-intelligence (2022).

74. Mahadevkar, S. V. *et al.* A review on machine learning styles in computer vision-techniques and future directions. *IEEE Access* **10,** 107293–107329 (2022).

75. Gilligan, L. P., Cobelli, M., Taufour, V. & Sanvito, S. A rule-free workflow for the automated generation of databases from scientific literature. *arXiv preprint arXiv:2301.11689 [cond-mat.mtrl-sci]* (2023).

76. Steck, H. *et al.* Deep learning for recommender systems: a Netflix case study. *AI Magazine* **42,** 7–18 (2021).

77. OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774 [cs.CL]* (2023).

78. Touvron, H. *et al.* Llama 2: open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288 [cs.CL]* (2023).

79. Borji, A. Generated faces in the wild: quantitative comparison of Stable Diffusion, Midjourney and DALL-E 2. *arXiv preprint arXiv:2210.00586 [cs.CV]* (2023).

80. Pratap, V. *et al.* Scaling speech technology to 1,000+ languages. *arXiv preprint arXiv:2305.13516 [cs.CL]* (2023).

81. Sun, S., Cao, Z., Zhu, H. & Zhao, J. A survey of optimization methods from a machine learning perspective. *IEEE Transactions on Cybernetics* **50,** 3668–3681 (2019).

82. Géron, A. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow* (O'Reilly Media Inc., Sebastopol, 2022).

83. Nair, V. & Hinton, G. E. Rectified linear units improve restricted Boltzmann machines in *Proceedings of the 27th International Conference on International Conference on Machine Learning* (Haifa, 2010).

84. Clevert, D.-A., Unterthiner, T. & Hochreiter, S. Fast and accurate deep network learning by exponential linear units (ELUs). *arXiv preprint arXiv:1511.07289 [cs.LG]* (2016).

85. Dugas, C., Bengio, Y., Bélisle, F., Nadeau, C. & Garcia, R. Incorporating second-order functional knowledge for better option pricing in *Proceedings of the 13th International Conference on Neural Information Processing Systems* (Denver, 2000).

86. Hornik, K., Stinchcombe, M. & White, H. Multilayer feedforward networks are universal approximators. *Neural Networks* **2,** 359–366 (1989).

87. Kelley, H. J. Gradient theory of optimal flight paths. *ARS Journal* **30,** 947–954 (1960).

88. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980 [cs.LG]* (2014).

89. Wu, Y., Mansimov, E., Liao, S., Grosse, R. & Ba, J. Scalable trust-region method for deep reinforcement learning using Kronecker-factored approximation in *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, 2017).

90. Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models in *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Vancouver, 2020).

91. Karras, T. *et al.* Analyzing and improving the image quality of StyleGAN in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (Online, 2020).

92. Buduma, N., Buduma, N. & Papa, J. *Fundamentals of Deep Learning* (O'Reilly Media Inc., Sebastopol, 2022).

93. Müser, M. H., Sukhomlinov, S. V. & Pastewka, L. Interatomic potentials: achievements and challenges. *Advances in Physics: X* **8,** 2093129 (2023).

94. Carlsson, A. E. Beyond pair potentials in elemental transition metals and semiconductors in *Solid State Physics* (Academic Press Inc., San Diego, 1990), pp. 1–91.

95. Tadmor, E. B. & Miller, R. E. *Modeling Materials: Continuum, Atomistic and Multiscale Techniques* (Cambridge University Press, New York, 2011).

96. Morse, P. M. Diatomic molecules according to the wave mechanics. II. vibrational levels. *Physical Review* **34,** 57 (1929).

97. Buckingham, R. A. The classical equation of state of gaseous helium, neon and argon. *Proceedings of the Royal Society of London A* **168,** 264–283 (1938).

98. Lennard-Jones, J. E. Cohesion. *Proceedings of the Physical Society* **43,** 461 (1931).

99. Ducastelle, F. Modules élastiques des métaux de transition. *Journal de Physique* **31,** 1055–1062 (1970).

100. Foiles, S., Baskes, M. & Daw, M. S. Embedded-atom-method functions for the fcc metals Cu, Ag, Au, Ni, Pd, Pt, and their alloys. *Physical Review B* **33,** 7983 (1986).

101. Tersoff, J. New empirical model for the structural properties of silicon. *Physical Review Letters* **56,** 632 (1986).

102. Finnis, M. & Sinclair, J. A simple empirical N-body potential for transition metals. *Philosophical Magazine A* **50,** 45–55 (1984).

103. Kohn, W. Density functional and density matrix method scaling linearly with the number of atoms. *Physical Review Letters* **76,** 3168–3171 (1996).

104. Deringer, V. L., Caro, M. A. & Csányi, G. Machine learning interatomic potentials as emerging tools for materials science. *Advanced Materials* **31,** 1902765 (2019).

105. Morrow, J. D., Gardner, J. L. & Deringer, V. L. How to validate machine-learned interatomic potentials. *The Journal of Chemical Physics* **158,** 121501 (2023).

106. Podryabinkin, E. V. & Shapeev, A. V. Active learning of linearly parametrized interatomic potentials. *Computational Materials Science* **140,** 171–180 (2017).

107. Zhang, L., Lin, D.-Y., Wang, H., Car, R. & Weinan, E. Active learning of uniformly accurate interatomic potentials for materials simulation. *Physical Review Materials* **3,** 023804 (2019).

108. Vandermause, J. *et al.* On-the-fly active learning of interpretable Bayesian force fields for atomistic rare events. *npj Computational Materials* **6,** 20 (2020).

109. Karabin, M. & Perez, D. An entropy-maximization approach to automated training set generation for interatomic potentials. *The Journal of Chemical Physics* **153,** 094110 (2020).

110. Montes de Oca Zapiain, D. *et al.* Training data selection for accuracy and transferability of interatomic potentials. *npj Computational Materials* **8,** 189 (2022).

111. Briganti, V. & Lunghi, A. Efficient generation of stable linear machine-learning force fields with uncertainty-aware active learning. *Machine Learning: Science and Technology* **4,** 035005 (2023).

112. Behler, J. & Parrinello, M. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Physical Review Letters* **98,** 146401 (2007).

113. Behler, J. Constructing high-dimensional neural network potentials: a tutorial review. *International Journal of Quantum Chemistry* **115,** 1032–1050 (2015).

114. Behler, J. Four generations of high-dimensional neural network potentials. *Chemical Reviews* **121,** 10037–10072 (2021).

115. Gastegger, M., Schwiedrzik, L., Bittermann, M., Berzsenyi, F. & Marquetand, P. wACSF — weighted atom-centered symmetry functions as descriptors in machine learning potentials. *The Journal of Chemical Physics* **148,** 241709 (2018).

116. Artrith, N., Morawietz, T. & Behler, J. High-dimensional neural-network potentials for multicomponent systems: applications to zinc oxide. *Physical Review B* **83,** 153101 (2011).

117. Anstine, D. M. & Isayev, O. Machine learning interatomic potentials and long-range physics. *The Journal of Physical Chemistry A* **127,** 2417–2431 (2023).

118. Ko, T. W., Finkler, J. A., Goedecker, S. & Behler, J. A fourth-generation high-dimensional neural network potential with accurate electrostatics including non-local charge transfer. *Nature Communications* **12,** 398 (2021).

119. Eckhoff, M. *et al.* Closing the gap between theory and experiment for lithium manganese oxide spinels using a high-dimensional neural network potential. *Physical Review B* **102,** 174102 (2020).

120. Eckhoff, M. & Behler, J. Insights into lithium manganese oxide–water interfaces using machine learning potentials. *The Journal of Chemical Physics* **155,** 244703 (2021).

121. Weinreich, J., Paleico, M. L. & Behler, J. Properties of α-brass nanoparticles II: structure and composition. *The Journal of Physical Chemistry C* **125,** 14897–14909 (2021).

122. Mangold, C. *et al.* Transferability of neural network potentials for varying stoichiometry: phonons and thermal conductivity of Mn_xGe_y compounds. *Journal of Applied Physics* **127,** 244901 (2020).

123. Bartók-Pártay, A. *The Gaussian Approximation Potential: An Interatomic Potential Derived from First Principles Quantum Mechanics* (Springer Science & Business Media, Heidelberg, 2010).

124. Bartók, A. P., Kondor, R. & Csányi, G. On representing chemical environments. *Physical Review B* **87,** 184115 (2013).

125. Drautz, R. Atomic cluster expansion for accurate and transferable interatomic potentials. *Physical Review B* **99,** 014104 (2019).

126. Wood, M. A. & Thompson, A. P. Extending the accuracy of the SNAP interatomic potential form. *The Journal of Chemical Physics* **148,** 241721 (2018).

127. Chen, C. *et al.* Accurate force field for molybdenum by machine learning large materials data. *Physical Review Materials* **1,** 043603 (2017).

128. Li, X.-G., Chen, C., Zheng, H., Zuo, Y. & Ong, S. P. Complex strengthening mechanisms in the NbMoTaW multi-principal element alloy. *npj Computational Materials* **6,** 70 (2020).

129. Deng, Z., Chen, C., Li, X.-G. & Ong, S. P. An electrostatic spectral neighbor analysis potential for lithium nitride. *npj Computational Materials* **5,** 75 (2019).

130. Cusentino, M. A., Wood, M. A. & Thompson, A. P. Explicit multielement extension of the spectral neighbor analysis potential for chemically complex systems. *The Journal of Physical Chemistry A* **124,** 5456–5464 (2020).

131. Nguyen-Cong, K. *et al.* Billion atom molecular dynamics simulations of carbon at extreme conditions and experimental time and length scales in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (St. Louis, 2021).

132. Wang, H., Zhang, L., Han, J. & E, W. DeePMD-kit: a deep learning package for many-body potential energy representation and molecular dynamics. *Computer Physics Communications* **228,** 178–184 (2018).

133. Smith, J. S., Isayev, O. & Roitberg, A. E. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chemical Science* **8,** 3192–3203 (2017).

134. Artrith, N., Urban, A. & Ceder, G. Efficient and accurate machine-learning interpolation of atomic energies in compositions with many species. *Physical Review B* **96,** 014112 (2017).

135. Jiang, B. & Guo, H. Permutation invariant polynomial neural network approach to fitting potential energy surfaces. *The Journal of Chemical Physics* **139,** 054112 (2013).

136. Zhang, Y., Hu, C. & Jiang, B. Embedded atom neural network potentials: efficient and accurate machine learning with a physically inspired representation. *The Journal of Physical Chemistry Letters* **10,** 4962–4967 (2019).

137. Glick, Z. L. *et al.* AP-Net: an atomic-pairwise neural network for smooth and transferable interaction potentials. *The Journal of Chemical Physics* **153,** 044112 (2020).

138. Ghasemi, S. A., Hofstetter, A., Saha, S. & Goedecker, S. Interatomic potentials for ionic systems with density functional accuracy based on charge densities obtained by a neural network. *Physical Review B* **92,** 045131 (2015).

139. Xie, X., Persson, K. A. & Small, D. W. Incorporating electronic information into machine learning potential energy surfaces via approaching the ground-state electronic energy as a function of atom-based electronic populations. *Journal of Chemical Theory and Computation* **16,** 4256–4270 (2020).

140. Bartók, A. P., Payne, M. C., Kondor, R. & Csányi, G. Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons. *Physical Review Letters* **104,** 136403 (2010).

141. MacKay, D. J. *Information Theory, Inference and Learning Algorithms* (Cambridge University Press, Cambridge, 2003).

142. Deringer, V. L. *et al.* Gaussian process regression for materials and molecules. *Chemical Reviews* **121,** 10073–10141 (2021).

143. Li, Z., Kermode, J. R. & De Vita, A. Molecular dynamics with on-the-fly machine learning of quantum-mechanical forces. *Physical Review Letters* **114,** 096405 (2015).

144. Chmiela, S. *et al.* Machine learning of accurate energy-conserving molecular force fields. *Science Advances* **3,** e1603015 (2017).

145. Christensen, A. S., Bratholm, L. A., Faber, F. A. & Anatole von Lilienfeld, O. FCHL revisited: faster and more accurate quantum machine learning. *The Journal of Chemical Physics* **152,** 044107 (2020).

146. Rupp, M., Tkatchenko, A., Müller, K.-R. & Von Lilienfeld, O. A. Fast and accurate modeling of molecular atomization energies with machine learning. *Physical Review Letters* **108,** 058301 (2012).

147. Jinnouchi, R., Lahnsteiner, J., Karsai, F., Kresse, G. & Bokdam, M. Phase transitions of hybrid perovskites simulated by machine-learning force fields trained on the fly with Bayesian inference. *Physical Review Letters* **122,** 225701 (2019).

148. Shapeev, A. V. Moment tensor potentials: a class of systematically improvable interatomic potentials. *Multiscale Modeling & Simulation* **14,** 1153–1173 (2016).

149. Domina, M., Patil, U., Cobelli, M. & Sanvito, S. Cluster expansion constructed over jacobi-legendre polynomials for accurate force fields. *Physical Review B* **108,** 094102 (2023).

150. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521,** 436–444 (2015).

151. Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. Neural message passing for quantum chemistry in *Proceedings of the 34th International Conference on Machine Learning* (Sydney, 2017).

152. Schütt, K. T. *et al.* SchNet: A continuous-filter convolutional neural network for modeling quantum interactions in *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, 2017).

153. Gasteiger, J., Groß, J. & Günnemann, S. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123 [cs.LG]* (2020).

154. Batzner, S. *et al.* E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications* **13,** 2453 (2022).

155. Schütt, K., Unke, O. & Gastegger, M. Equivariant message passing for the prediction of tensorial properties and molecular spectra in *Proceedings of the 38th International Conference on Machine Learning* (Online, 2021).

156. Batatia, I., Kovacs, D. P., Simm, G., Ortner, C. & Csanyi, G. MACE: higher order equivariant message passing neural networks for fast and accurate force fields in *Proceedings of the 36th International Conference on Neural Information Processing Systems* (New Orleans, 2022).

157. Xie, T. & Grossman, J. C. Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties. *Physical Review Letters* **120,** 145301 (2018).

158. Schmidt, J., Pettersson, L., Verdozzi, C., Botti, S. & Marques, M. A. Crystal graph attention networks for the prediction of stable materials. *Science Advances* **7,** eabi7948 (2021).

159. Schmidt, J. *et al.* Machine-learning-assisted determination of the global zero-temperature phase diagram of materials. *Advanced Materials* **35,** 2210788 (2023).

160. Bartel, C. J. *et al.* A critical examination of compound stability predictions from machine-learned formation energies. *npj Computational Materials* **6,** 97 (2020).

161. Pandey, S., Qu, J., Stevanović, V., John, P. S. & Gorai, P. Predicting energy and stability of known and hypothetical crystals using graph neural network. *Patterns* **2,** 100361 (2021).

162. Park, C. W. & Wolverton, C. Developing an improved crystal graph convolutional neural network framework for accelerated materials discovery. *Physical Review Materials* **4,** 063801 (2020).

163. Chen, C. & Ong, S. P. A universal graph deep learning interatomic potential for the periodic table. *Nature Computational Science* **2,** 718–728 (2022).

164. Deng, B. *et al.* CHGNet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nature Machine Intelligence* **5,** 1031–1041 (2023).

165. Zuo, Y. *et al.* Performance and cost assessment of machine learning interatomic potentials. *Journal of Physical Chemistry A* **124,** 731–745 (2020).

166. Niu, H., Bonati, L., Piaggi, P. M. & Parrinello, M. Ab initio phase diagram and nucleation of gallium. *Nature Communications* **11,** 2654 (2020).

167. Deringer, V. L. & Csányi, G. Machine learning based interatomic potential for amorphous carbon. *Physical Review B* **95,** 094203 (2017).

168. Bartók, A. P., Kermode, J., Bernstein, N. & Csányi, G. Machine learning a general-purpose interatomic potential for silicon. *Physical Review X* **8,** 041048 (2018).

169. Pandey, A., Gigax, J. & Pokharel, R. Machine learning interatomic potential for high-throughput screening of high-entropy alloys. *JOM* **74,** 2908–2920 (2022).

170. Pickard, C. J. Ephemeral data derived potentials for random structure search. *Physical Review B* **106,** 014102 (2022).

171. Podryabinkin, E. V., Tikhonov, E. V., Shapeev, A. V. & Oganov, A. R. Accelerating crystal structure prediction by machine-learning interatomic potentials with active learning. *Physical Review B* **99,** 064114 (2019).

172. Tong, Q., Xue, L., Lv, J., Wang, Y. & Ma, Y. Accelerating CALYPSO structure prediction by data-driven learning of a potential energy surface. *Faraday Discussions* **211,** 31–43 (2018).

173. Bisbo, M. K. & Hammer, B. Global optimization of atomistic structure enhanced by machine learning. *Physical Review B* **105,** 245404 (2022).

174. Mortensen, H. L., Meldgaard, S. A., Bisbo, M. K., Christiansen, M.-P. V. & Hammer, B. Atomistic structure learning algorithm with surrogate energy model relaxation. *Physical Review B* **102,** 075427 (2020).

175. Gubaev, K., Podryabinkin, E. V., Hart, G. L. & Shapeev, A. V. Accelerating high-throughput searches for new alloys with active learning of interatomic potentials. *Computational Materials Science* **156,** 148–156 (2019).

176. Mortazavi, B. *et al.* Machine-learning interatomic potentials enable first-principles multiscale modeling of lattice thermal conductivity in graphene/borophene heterostructures. *Materials Horizons* **7,** 2359–2367 (2020).

177. Rosenbrock, C. W. *et al.* Machine-learned interatomic potentials for alloys and alloy phase diagrams. *npj Computational Materials* **7,** 24 (2021).

178. Srinivasan, P., Shapeev, A., Neugebauer, J., Körmann, F. & Grabowski, B. Anharmonicity in bcc refractory elements: a detailed *ab initio* analysis. *Physical Review B* **107,** 014301 (2023).

179. Eckhoff, M. & Behler, J. High-dimensional neural network potentials for magnetic systems using spin-dependent atom-centered symmetry functions. *npj Computational Materials* **7,** 170 (2021).

180. Novikov, I., Grabowski, B., Körmann, F. & Shapeev, A. Magnetic moment tensor potentials for collinear spin-polarized materials reproduce different magnetic states of bcc Fe. *npj Computational Materials* **8,** 13 (2022).

181. Nikolov, S. *et al.* Data-driven magneto-elastic predictions with scalable classical spin-lattice dynamics. *npj Computational Materials* **7,** 153 (2021).

182. Domina, M., Cobelli, M. & Sanvito, S. Spectral neighbor representation for vector fields: machine learning potentials including spin. *Physical Review B* **105,** 214439 (2022).

183. Giannozzi, P. *et al.* QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. *Journal of Physics: Condensensed Matter* **21,** 395502 (2009).

184. Gonze, X. *et al.* The ABINIT project: impact, environment and recent developments. *Computer Physics Communications* **248,** 107042 (2020).

185. Hautier, G. *et al.* Novel mixed polyanions lithium-ion battery cathode materials predicted by high-throughput *ab initio* computations. *Journal of Materials Chemistry* **21,** 17147–17153 (2011).

186. Lederer, Y., Toher, C., Vecchio, K. S. & Curtarolo, S. The search for high entropy alloys: a high-throughput *ab-initio* approach. *Acta Materialia* **159,** 364–383 (2018).

187. Curtarolo, S. *et al.* The high-throughput highway to computational materials design. *Nature Materials* **12,** 191–201 (2013).

188. Oganov, A. R., Saleh, G. & Kvashnin, A. G. *Computational Materials Discovery* (Royal Society of Chemistry, Croydon, 2018).

189. Hinuma, Y. *et al.* Discovery of earth-abundant nitride semiconductors by computational screening and high-pressure synthesis. *Nature Communications* **7,** 11962 (2016).

190. Pickard, C. J. & Needs, R. *Ab initio* random structure searching. *Journal of Physics: Condensensed Matter* **23,** 053201 (2011).

191. Lyakhov, A. O., Oganov, A. R., Stokes, H. T. & Zhu, Q. New developments in evolutionary structure prediction algorithm USPEX. *Computer Physics Communications* **184,** 1172–1182 (2013).

192. Chen, Z., Jia, W., Jiang, X., Li, S.-S. & Wang, L.-W. SGO: a fast engine for ab initio atomic structure global optimization by differential evolution. *Computer Physics Communications* **219,** 35–44 (2017).

193. Kaappa, S., Del Río, E. G. & Jacobsen, K. W. Global optimization of atomic structures with gradient-enhanced gaussian process regression. *Physical Review B* **103,** 174114 (2021).

194. Brockherde, F. *et al.* Bypassing the Kohn-Sham equations with machine learning. *Nature Communications* **8,** 872 (2017).

195. Grisafi, A. *et al.* Transferable machine-learning model of the electron density. *ACS Central Science* **5,** 57–64 (2018).

196. Focassio, B., Domina, M., Patil, U., Fazzio, A. & Sanvito, S. Linear Jacobi-Legendre expansion of the charge density for machine learning-accelerated electronic structure calculations. *npj Computational Materials* **9,** 87 (2023).

197. Yang, Y., Jiménez-Negrón, O. A. & Kitchin, J. R. Machine-learning accelerated geometry optimization in molecular simulation. *The Journal of Chemical Physics* **154,** 234704 (2021).

198. Jinnouchi, R., Karsai, F. & Kresse, G. On-the-fly machine learning force field generation: application to melting points. *Physical Review B* **100,** 014105 (2019).

199. Zhang, X. & Sluiter, M. H. Cluster expansions for thermodynamics and kinetics of multicomponent alloys. *Journal of Phase Equilibria and Diffusion* **37,** 44–52 (2016).

200. Wu, Q., He, B., Song, T., Gao, J. & Shi, S. Cluster expansion method and its application in computational materials science. *Computational Materials Science* **125,** 243–254 (2016).

201. Hautier, G., Fischer, C. C., Jain, A., Mueller, T. & Ceder, G. Finding nature's missing ternary oxide compounds using machine learning and density functional theory. *Chemistry of Materials* **22,** 3762–3767 (2010).

202. Oliynyk, A. O., Adutwum, L. A., Harynuk, J. J. & Mar, A. Classifying crystal structures of binary compounds ab through cluster resolution feature selection and support vector machine analysis. *Chemistry Materials* **28,** 6672–6681 (2016).

203. Kim, K. *et al.* Machine-learning-accelerated high-throughput materials screening: discovery of novel quaternary Heusler compounds. *Physical Review Materials* **2,** 123801 (2018).

204. Schmidt, J., Chen, L., Botti, S. & Marques, M. A. Predicting the stability of ternary intermetallics with density functional theory and machine learning. *The Journal of Chemical Physics* **148,** 241728 (2018).

205. Artrith, N. & Behler, J. High-dimensional neural network potentials for metal surfaces: a prototype study for copper. *Physical Review B* **85,** 045439 (2012).

206. Deringer, V. L. *et al.* Origins of structural and electronic transitions in disordered silicon. *Nature* **589,** 59–64 (2021).

207. Li, X.-G. *et al.* Quantum-accurate spectral neighbor analysis potential models for Ni-Mo binary alloys and fcc metals. *Physical Review B* **98,** 094104 (2018).

208. Shimizu, K. *et al.* Phase stability of Au-Li binary systems studied using neural network potential. *Physical Review B* **103,** 094112 (2021).

209. Ibarra-Hernández, W. *et al.* Structural search for stable Mg-Ca alloys accelerated with a neural network interatomic model. *Physical Chemistry Chemical Physics* **20,** 27545–27557 (2018).

210. Hajinazar, S., Shao, J. & Kolmogorov, A. N. Stratified construction of neural network based interatomic models for multicomponent materials. *Physical Review B* **95,** 014114 (2017).

211. Rowe, P., Deringer, V. L., Gasparotto, P., Csányi, G. & Michaelides, A. An accurate and transferable machine learning potential for carbon. *The Journal of Chemical Physics* **153,** 034702 (2020).

212. Abedsoltan, A., Belkin, M. & Pandit, P. Toward large kernel models. *arXiv preprint arXiv:2302.02605 [cs.LG]* (2023).

213. Artrith, N., Urban, A. & Ceder, G. Constructing first-principles phase diagrams of amorphous Li_xSi using machine-learning-assisted sampling with an evolutionary algorithm. *The Journal of Chemical Physics* **148,** 241711 (2018).

214. Toher, C., Oses, C., Hicks, D. & Curtarolo, S. Unavoidable disorder and entropy in multi-component systems. *npj Computational Materials* **5,** 69 (2019).

215. Baker, H. Introduction to alloy phase diagrams in *ASM Handbook Volume 3: Alloy Phase Diagrams* (ASM International, Novelty, 1992), pp. 1.1–1.29.

216. Porter, D. A., Easterling, K. E. & Sherif, M. Y. *Phase Transformations in Metals and Alloys* (CRC Press, Abingdon, 2021).

217. Callen, H. B. *Thermodynamics and an Introduction to Thermostatistics* (John Wiley & Sons Inc., Singapore, 1991).

218. Pártay, L. B., Bartók, A. P. & Csányi, G. Efficient sampling of atomic configurational spaces. *The Journal of Physical Chemistry B* **114,** 10502–10512 (2010).

219. Saunders, N. & Miodownik, A. P. *CALPHAD: Calculation of Phase Diagrams: A Comprehensive Guide* (Elsevier, Oxford, 1998).

220. Körmann, F. *et al.* Free energy of bcc iron: integrated *ab initio* derivation of vibrational, electronic, and magnetic contributions. *Physical Review B* **78,** 033102 (2008).

221. Grabowski, B., Hickel, T. & Neugebauer, J. Formation energies of point defects at finite temperatures. *Physica Status Solidi B* **248,** 1295–1308 (2011).

222. Cao, J. & Berne, B. A Born–Oppenheimer approximation for path integrals with an application to electron solvation in polarizable fluids. *The Journal of Chemical Physics* **99,** 2902–2916 (1993).

223. Ikeda, Y., Grabowski, B. & Körmann, F. *Ab initio* Phase stabilities and mechanical properties of multicomponent alloys: a comprehensive review for high entropy alloys and compositionally complex alloys. *Materials Characterization* **147,** 464–511 (2019).

224. Zunger, A., Wei, S.-H., Ferreira, L. & Bernard, J. E. Special quasirandom structures. *Physical Review Letters* **65,** 353 (1990).

225. Vitos, L. Total-energy method based on the exact muffin-tin orbitals theory. *Physical Review B* **64,** 014107 (2001).

226. Vitos, L., Abrikosov, I. & Johansson, B. Anisotropic lattice distortions in random alloys from first-principles theory. *Physical Review Letters* **87,** 156401 (2001).

227. Kresse, G., Furthmüller, J. & Hafner, J. *Ab initio* Force constant approach to phonon dispersion relations of diamond and graphite. *Europhysics Letters* **32,** 729 (1995).

228. Parlinski, K., Li, Z. & Kawazoe, Y. First-principles determination of the soft mode in cubic ZrO_2. *Physical Review Letters* **78,** 4063 (1997).

229. Körbel, S., Marques, M. A. & Botti, S. Stability and electronic properties of new inorganic perovskites from high-throughput *ab initio* calculations. *Journal of Materials Chemistry C* **4,** 3157–3167 (2016).

230. Schmidt, J. *et al.* Predicting the thermodynamic stability of solids combining density functional theory and machine learning. *Chemistry of Materials* **29,** 5090–5103 (2017).

231. Carrete, J., Li, W., Mingo, N., Wang, S. & Curtarolo, S. Finding unprecedentedly low-thermal-conductivity half-Heusler semiconductors via high-throughput materials modeling. *Physical Review X* **4,** 011019 (2014).

232. Armiento, R., Kozinsky, B., Hautier, G., Fornari, M. & Ceder, G. High-throughput screening of perovskite alloys for piezoelectric performance and thermodynamic stability. *Physical Review B* **89,** 134103 (2014).

233. Jain, A. *et al.* A high-throughput infrastructure for density functional theory calculations. *Computational Materials Science* **50,** 2295–2310 (2011).

234. Pallikara, I., Kayastha, P., Skelton, J. M. & Whalley, L. D. The physical significance of imaginary phonon modes in crystals. *Electronic Structure* **4,** 033002 (2022).

235. Oses, C. *et al.* AFLOW-CHULL: cloud-oriented platform for autonomous phase stability analysis. *Journal of Chemical Information and Modeling* **58,** 2477–2490 (2018).

236. Barber, C. B., Dobkin, D. P. & Huhdanpaa, H. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software* **22,** 469–483 (1996).

237. Ong S. P., Richards W. D. *pyhull* version 2.1. Sept. 25, 2018. `https://pypi.org/project/pyhull/`.

238. Curtarolo, S. *et al.* AFLOW: an automatic framework for high-throughput materials discovery. *Computational Materials Science* **58,** 218–226 (2012).

239. Villars, P. *et al.* The Pauling file, binaries edition. *Journal of Alloys and Compounds* **367,** 293–297 (2004).

240. Setyawan, W. & Curtarolo, S. High-throughput electronic band structure calculations: challenges and tools. *Computational Materials Science* **49,** 299–312 (2010).

241. Monkhorst, H. J. & Pack, J. D. Special points for Brillouin-zone integrations. *Physical Review B* **13,** 5188 (1976).

242. Methfessel, M. & Paxton, A. High-precision sampling for Brillouin-zone integration in metals. *Physical Review B* **40,** 3616 (1989).

243. Blöchl, P. E., Jepsen, O. & Andersen, O. K. Improved tetrahedron method for Brillouin-zone integrations. *Physical Review B* **49,** 16223 (1994).

244. Taylor, R. H. *et al.* A RESTful API for exchanging materials data in the AFLOW-LIB.org consortium. *Computational Materials Science* **93,** 178–192 (2014).

245. Rose, F. *et al.* AFLUX: the LUX materials search API for the AFLOW data repositories. *Computational Materials Science* **137,** 362–370 (2017).

246. Masse, M. *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces* (O'Reilly Media, Inc., Sebastopol, 2011).

247. Mehl, M. J. *et al.* The AFLOW library of crystallographic prototypes: part 1. *Computational Materials Science* **136,** S1–S828 (2017).

248. Hicks, D. *et al.* The AFLOW library of crystallographic prototypes: part 2. *Computational Materials Science* **161,** S1–S1011 (2019).

249. Hicks, D. *et al.* The AFLOW library of crystallographic prototypes: part 3. *Computational Materials Science* **199,** 110450 (2021).

250. Wyckoff, R. W. G. *The Analytical Expression of the Results of the Theory of Space-groups* **318** (Carnegie Institution of Washington, 1922).

251. *International Tables for Crystallography. Volume A: Space-Group Symmetry* (ed Hahn, T.) (Springer Netherlands, Dordrecht, 2002).

252. Aroyo, M. I. *et al.* Crystallography online: Bilbao crystallographic server. *Bulgarian Chemcal Communications* **43,** 183–197 (2011).

253. Hicks, D. *et al.* AFLOW-Xtalfinder: a reliable choice to identify crystalline prototypes. *npj Computational Materials* **7,** 30 (2021).

254. Hicks, D. *et al.* AFLOW-SYM: platform for the complete, automatic and self-consistent symmetry analysis of crystals. *Acta Crystallographica Section A: Foundations and Advances* **74,** 184–203 (2018).

255. Kim, H.-I. *et al.* Age-hardenability and related microstructural changes during and after phase transformation in an Au-Ag-Cu-based dental alloy. *Materials Research* **16,** 71–87 (2013).

256. Ellis, T. W. The future of gold in electronics. *Gold Bulletin* **37,** 66–71 (2004).

257. Yang, Y. *et al.* The wetting characteristics of molten Ag–Cu–Au on Cu substrates: a molecular dynamics study. *Physical Chemistry Chemical Physics* **22,** 25904–25917 (2020).

258. Gong, C. & Leite, M. S. Noble metal alloys for plasmonics. *ACS Photonics* **3,** 507–513 (2016).

259. Suh, I.-K., Ohta, H. & Waseda, Y. High-temperature thermal expansion of six metallic elements measured by dilatation method and X-ray diffraction. *Journal of Materials Science* **23,** 757–760 (1988).

260. Kusoffsky, A. Thermodynamic evaluation of the ternary Ag–Au–Cu system—including a short range order description. *Acta Materialia* **50,** 5139–5145 (2002).

261. Cao, W., Chang, Y., Zhu, J., Chen, S. & Oates, W. Thermodynamic modeling of the Cu–Ag–Au system using the cluster/site approximation. *Intermetallics* **15,** 1438–1446 (2007).

262. Wagner, C. Thermodynamics of the liquidus and the solidus of binary alloys. *Acta Metallurgica* **2,** 242–249 (1954).

263. Steininger, J. Thermodynamics and calculation of the liquidus-solidus gap in homogeneous, monotonic alloy systems. *Journal of Applied Physics* **41,** 2713–2724 (1970).

264. Lu, Z., Klein, B. & Zunger, A. Atomic short-range order and alloy ordering tendency in the Ag-Au system. *Modelling and Simulation in Materials Science and Engineering* **3,** 753 (1995).

265. Curtarolo, S., Morgan, D. & Ceder, G. Accuracy of *ab initio* methods in predicting the crystal structures of metals: a review of 80 binary alloys. *Calphad* **29,** 163–211 (2005).

266. Ozoliņš, V., Wolverton, C. & Zunger, A. Cu-Au, Ag-Au, Cu-Ag, and Ni-Au intermetallics: first-principles study of temperature-composition phase diagrams and structures. *Physical Review B* **57,** 6427 (1998).

267. Zarkevich, N. A., Tan, T. L., Wang, L.-L. & Johnson, D. D. Low-energy antiphase boundaries, degenerate superstructures, and phase stability in frustrated fcc Ising model and Ag-Au alloys. *Physical Review B* **77,** 144208 (2008).

268. Zhang, Y., Kresse, G. & Wolverton, C. Nonlocal first-principles calculations in Cu-Au and other intermetallic alloys. *Physical Review Letters* **112,** 075502 (2014).

269. Okamoto, H., Chakrabarti, D., Laughlin, D. & Massalski, T. The Au-Cu (gold-copper) system. *Journal of Phase Equilibria* **8,** 454–474 (1987).

270. Feraoun, H., Aourag, H., Grosdidier, T., Klein, D. & Coddet, C. Development of modified embedded atom potentials for the Cu–Ag system. *Superlattices and Microstructures* **30,** 261–271 (2001).

271. Chakraborty, B. & Xi, Z. Atomistic Landau theory of ordering and modulated phases in Cu-Au alloys. *Physical Review Letters* **68,** 2039 (1992).

272. Zunger, A., Wang, L., Hart, G. L. & Sanati, M. Obtaining Ising-like expansions for binary alloys from first principles. *Modelling and Simulation in Materials Science and Engineering* **10,** 685 (2002).

273. Tian, L.-Y. *et al.* Exchange-correlation catastrophe in Cu-Au: a challenge for semilocal density functional approximations. *Physical Review Letters* **117,** 066401 (2016).

274. Pandey, S. *et al.* Thermodynamics and kinetics of ordered and disordered Cu/Au alloys from first principles calculations. *Journal of Alloys and Compounds* **809,** 151615 (2019).

275. Tian, L.-Y. *et al.* CPA descriptions of random Cu-Au alloys in comparison with SQS approach. *Computational Materials Science* **128,** 302–309 (2017).

276. Tian, L.-Y. *et al.* Density functional theory description of random Cu-Au alloys. *Physical Review B* **99,** 064202 (2019).

277. Stockdale, D. The solid solutions of the copper-silver system. *Journal of the Institute of Metals* **45,** 127–155 (1931).

278. Rogers, J. X-ray study of copper silver alloys. *Institute of Metals* **57,** 173 (1935).

279. Heycock, C. T. & Neville, F. H. Complete freezing-point curves of binary alloys containing silver or copper together with another metal. *Proceedings of the Royal Society of London A* **60,** 160–164 (1897).

280. Broniewski, W. & Kostacz, S. On the alloys of Ag–Cu. *Comptes Rendus Hebdomadaires des Séances de l'Académie des Sceinces* **194,** 973–975 (1932).

281. Yasuhara, A. & Sannomiya, T. Atomically localized ordered phase and segregation at grain boundaries in Au–Ag–Cu ternary alloy nanoparticles. *The Journal of Physical Chemistry C* **126,** 1160–1167 (2022).

282. Kogachi, M. & Nakahigashi, K. Phase relations in the $AuCu_{1-y}Ag_y$ and $Au(Cu_{1-y}Ag_y)_3$ ternary systems. *Japanese Journal of Applied Physics* **24,** 121 (1985).

283. Uzuka, T., Kanzawa, Y. & Yasuda, K. Determination of the AuCu superlattice formation region in gold-copper-silver ternary system. *Journal of Dental Research* **60,** 883–889 (1981).

284. Buschow, K. The importance of ternary intermetallic compounds in science and technology. *Journal of Alloys and Compounds* **193,** 223–230 (1993).

285. Pearson, W. B. The crystal chemistry and physics of metals and alloys. *Journal of Applied Crystallography* **6,** 306–307 (1972).

286. Schubert, K. *et al.* Einige strukturelle ergebnisse an metallischen phasen. *Naturwissenschaften* **43,** 248–249 (1956).

287. Hart, G. L., Nelson, L. J. & Forcade, R. W. Generating derivative structures at a fixed concentration. *Computational Materials Science* **59,** 101–107 (2012).

288. Wang, R. *et al.* Machine learning guided discovery of ternary compounds involving La and immiscible Co and Pb elements. *npj Computational Materials* **8,** 258 (2022).

289. Sun, H. *et al.* Machine learning-guided discovery of ternary compounds containing La, P, and group 14 elements. *Inorganic Chemistry* **61,** 16699–16706 (2022).

290. Meisel, K. Rheniumtrioxyd. III. mitteilung. über die kristallstruktur des rheniumtrioxyds. *Zeitschrift für Anorganische und Allgemeine Chemie* **207,** 121–128 (1932).

291. Downs, R. T. & Hall-Wallace, M. The American Mineralogist crystal structure database. *American Mineralogist* **88,** 247–250 (2003).

292. Bochkarev, A. *et al.* Efficient parametrization of the atomic cluster expansion. *Physical Review Materials* **6,** 013804 (2022).

293. Attarian, S., Morgan, D. & Szlufarska, I. Thermophysical properties of FLiBe using moment tensor potentials. *Journal of Molecular Liquids* **368,** 120803 (2022).

294. Unruh, D., Meidanshahi, R. V., Goodnick, S. M., Csányi, G. & Zimányi, G. T. Gaussian approximation potential for amorphous Si : H. *Physical Review Materials* **6,** 065603 (2022).

295. Larsen, A. H. *et al.* The atomic simulation environment—a python library for working with atoms. *Journal of Physics: Condensed Matter* **29,** 273002 (2017).

296. Tavares, S., Yang, K. & Meyers, M. A. Heusler alloys: past, properties, new alloys, and prospects. *Progress in Materials Science* **22,** 101017 (2023).

297. Gladyshevskii, E. & Kuz'ma, Y. B. The compounds Nb_4FeSi, Nb_4CoSi, Nb_4NiSi and their crystal structures. *Journal of Structural Chemistry* **6,** 60–63 (1965).

298. Barzilai, S., Toher, C., Curtarolo, S. & Levy, O. Molybdenum-titanium phase diagram evaluated from *ab initio* calculations. *Physical Review Materials* **1,** 023604 (2017).

299. Levy, O., Hart, G. L. & Curtarolo, S. Uncovering compounds by synergy of cluster expansion and high-throughput methods. *Journal of the American Chemical Society* **132,** 4830–4833 (2010).

300. Settles, B. *Active Learning* (Springer, Cham, 2012).

301. Lunghi, A. & Sanvito, S. A unified picture of the covalent bond within quantum-accurate force fields: from organic molecules to metallic complexes' reactivity. *Science Advances* **5,** eaaw2210 (2019).

302. Akiba, T., Sano, S., Yanase, T., Ohta, T. & Koyama, M. Optuna: a next-generation hyperparameter optimization framework in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Anchorage, 2019).

303. Bergstra, J., Bardenet, R., Bengio, Y. & Kégl, B. Algorithms for hyper-parameter optimization in *Proceedings of the 25th International Conference on Neural Information Processing Systems* (Granada, 2011).

304. Bergstra, J., Yamins, D. & Cox, D. Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures in *Proceedings of the 30th International Conference on Machine Learning* (Atlanta, 2013).

305. Coello, C. A. C., Ozaki, Y., Tanigaki, Y., Watanabe, S. & Onishi, M. Multiobjective tree-structured parzen estimator for computationally expensive optimization problems in *Proceedings of the 2020 Genetic and Evolutionary Computation Conference* (Cancún, 2020).

306. Van der Oord, C., Sachs, M., Kovács, D. P., Ortner, C. & Csányi, G. Hyperactive learning for data-driven interatomic potentials. *npj Computational Materials* **9,** 168 (2023).

307. Gubaev, K., Podryabinkin, E. V. & Shapeev, A. V. Machine learning of molecular properties: locality and active learning. *The Journal of Chemical Physics* **148** (2018).

308. Novikov, I. S., Gubaev, K., Podryabinkin, E. V. & Shapeev, A. V. The MLIP package: moment tensor potentials with MPI and active learning. *Machine Learning: Science and Technology* **2,** 025002 (2020).

309. De Aguiar, P. F., Bourguignon, B., Khots, M., Massart, D. & Phan-Than-Luu, R. D-optimal designs. *Chemometrics and Intelligent Laboratory Systems* **30,** 199–210 (1995).

310. Smith, K. On the standard deviations of adjusted and interpolated values of an observed polynomial function and its constants and the guidance they give towards a proper choice of the distribution of observations. *Biometrika* **12,** 1–85 (1918).

311. Goreinov, S. A., Oseledets, I. V., Savostyanov, D. V., Tyrtyshnikov, E. E. & Zamarashkin, N. L. How to find a good submatrix in *Matrix Methods: Theory, Algorithms And Applications* (World Scientific, Singapore, 2010), pp. 247–256.

312. Faber, F., Lindmaa, A., Von Lilienfeld, O. A. & Armiento, R. Crystal structure representations for machine learning models of formation energies. *International Journal of Quantum Chemistry* **115,** 1094–1101 (2015).

313. Ward, L. *et al.* Including crystal structure attributes in machine learning models of formation energies via Voronoi tessellations. *Physical Review B* **96,** 024104 (2017).

314. Goodall, R. E. & Lee, A. A. Predicting materials properties without crystal structure: deep representation learning from stoichiometry. *Nature Communications* **11,** 6280 (2020).

315. Jha, D. *et al. ElemNet*: deep learning the chemistry of materials from only elemental composition. *Scientific Reports* **8,** 17593 (2018).

316. Sotskov, V., Shapeev, A. V. & Podryabinkin, E. V. A machine learning potential-based generative algorithm for on-lattice crystal structure prediction. *arXiv preprint arXiv:2306.03989 [cond-mat.mtrl-sci]* (2023).

317. Bernstein, N., Csányi, G. & Deringer, V. L. De novo exploration and self-guided learning of potential-energy surfaces. *npj Computational Materials* **5,** 99 (2019).

318. Kharabadze, S., Thorn, A., Koulakova, E. A. & Kolmogorov, A. N. Prediction of stable Li-Sn compounds: boosting ab initio searches with neural network potentials. *npj Computational Materials* **8,** 136 (2022).

319. Seko, A. Machine learning potentials for multicomponent systems: the Ti-Al binary system. *Physical Review B* **102,** 174104 (2020).

320. Kelvinius, F. E., Armiento, R. & Lindsten, F. Graph-based machine learning beyond stable materials and relaxed crystal structures. *Physical Review Materials* **6,** 033801 (2022).

321. Law, J. N., Pandey, S., Gorai, P. & St. John, P. C. Upper-bound energy minimization to search for stable functional materials with graph neural networks. *JACS Au* **3,** 113–123 (2023).

322. Paleico, M. L. & Behler, J. Global optimization of copper clusters at the ZnO $(10\bar{1}0)$ surface using a DFT-based neural network potential and genetic algorithms. *The Journal of Chemical Physics* **153,** 054704 (2020).

323. Yamashita, T. *et al.* Crystal structure prediction accelerated by Bayesian optimization. *Physical Review Materials* **2,** 013803 (2018).

324. Deringer, V. L., Pickard, C. J. & Csányi, G. Data-driven learning of total and local energies in elemental boron. *Physical Review Letters* **120,** 156001 (2018).

325. Hart, G. L. & Forcade, R. W. Algorithm for generating derivative structures. *Physical Review B* **77,** 224115 (2008).

326. Hart, G. L. & Forcade, R. W. Generating derivative structures from multilattices: algorithm and application to hcp alloys. *Physical Review B* **80,** 014120 (2009).

327. Haghighatlari, M. *et al.* NewtonNet: a newtonian message passing network for deep learning of interatomic potentials and forces. *Digital Discovery* **1,** 333–343 (2022).

328. Chen, C., Ye, W., Zuo, Y., Zheng, C. & Ong, S. P. Graph networks as a universal machine learning framework for molecules and crystals. *Chemistry of Materials* **31,** 3564–3572 (2019).

329. Ong, S. P. *et al.* Python materials genomics (pymatgen): a robust, open-source python library for materials analysis. *Computational Materials Science* **68,** 314–319 (2013).

330. Battaglia, P. W. *et al.* Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261 [cs.LG]* (2018).

331. Schütt, K. T., Arbabzadah, F., Chmiela, S., Müller, K. R. & Tkatchenko, A. Quantum-chemical insights from deep tensor neural networks. *Nature Communications* **8,** 13890 (2017).

332. Vinyals, O., Bengio, S. & Kudlur, M. Order matters: sequence to sequence for sets. *arXiv preprint arXiv:1511.06391 [stat.ML]* (2015).

333. Ward, L. *et al.* Matminer: an open source toolkit for materials data mining. *Computational Materials Science* **152,** 60–69 (2018).

334. Dunn, A., Wang, Q., Ganose, A., Dopp, D. & Jain, A. Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Computational Materials* **6,** 138 (2020).

335. Chen, C., Zuo, Y., Ye, W., Li, X. & Ong, S. P. Learning properties of ordered and disordered materials from multi-fidelity data. *Nature Computational Science* **1,** 46–53 (2021).

336. Kocer, E., Mason, J. K. & Erturk, H. A novel approach to describe chemical environments in high-dimensional neural network potentials. *The Journal of Chemical Physics* **150** (2019).

337. Liu, H., Dai, Z., So, D. & Le, Q. V. Pay attention to MLPs in *Proceedings of the 35th International Conference on Neural Information Processing Systems* (Online, 2021).

338. Lin, T., Wang, Y., Liu, X. & Qiu, X. A survey of transformers. *AI Open* **3,** 111–132 (2022).

339. Singraber, A., Behler, J. & Dellago, C. Library-based LAMMPS implementation of high-dimensional neural network potentials. *Journal of Chemical Theory and Computation* **15,** 1827–1840 (2019).

340. Ramachandran, P., Zoph, B. & Le, Q. V. Searching for activation functions. *arXiv preprint arXiv:1710.05941 [cs.NE]* (2017).

341. Rostoker, W. *A study of ternary phase diagrams of tungsten and tantalum* tech. rep. **59**-492 (Armour Research Foundation of Illinois Institute of Technology, 1960).

342. Turchi, P. *et al.* Application of *ab initio* and CALPHAD thermodynamics to Mo-Ta-W alloys. *Physical Review B* **71,** 094206 (2005).

343. Kieffer, R., Sediatschek, K. & Braun, H. Tungsten alloys of high melting point. *Journal of Less-Common Metals* **1,** 19–33 (1959).

344. Zhang, Y. *et al.* Microstructure and mechanical properties of Mo-Ta-W refractory multi-principal element alloy thin films for hard protective coatings. *Surface and Coatings Technology* **431,** 128005 (2022).

345. Blum, V. & Zunger, A. Prediction of ordered structures in the bcc binary systems of Mo, Nb, Ta, and W from first-principles search of approximately 3,000,000 possible configurations. *Physical Review B* **72,** 020104 (2005).

346. Kainuma, R., Ono, N. & Ishida, K. Development of NiAl(B2)-base shape memory alloys. *MRS Online Proceedings Library* **360,** 467–478 (1994).

347. Hasegawa, H. & Pettifor, D. Microscopic theory of the temperature-pressure phase diagram of iron. *Physical Review Letters* **50,** 130 (1983).

348. Sundman, B., Ohnuma, I., Dupin, N., Kattner, U. R. & Fries, S. G. An assessment of the entire Al–Fe system including $D0_3$ ordering. *Acta Materialia* **57,** 2896–2908 (2009).

349. Okamoto, H. Al-Ni (aluminum-nickel). *Journal of Phase Equilibria* **14,** 257–259 (1993).

350. Swartzendruber, L., Itkin, V. & Alcock, C. The Fe-Ni (iron-nickel) system. *Journal of Phase Equilibria* **12,** 288–312 (1991).

351. Shang, S.-L. *et al.* Forming mechanism of equilibrium and non-equilibrium metallurgical phases in dissimilar aluminum/steel (Al–Fe) joints. *Scientific Reports* **11,** 24251 (2021).

352. Dupin, N., Ansara, I. & Sundman, B. Thermodynamic re-assessment of the ternary system Al-Cr-Ni. *Calphad* **25,** 279–298 (2001).

353. Wang, W., Chen, H.-L., Larsson, H. & Mao, H. Thermodynamic constitution of the Al–Cu–Ni system modeled by CALPHAD and ab initio methodology for designing high entropy alloys. *Calphad* **65,** 346–369 (2019).

354. Ellner, M., Kek, S. & Predel, B. Ni_3Al_4—a phase with ordered vacancies isotypic to Ni_3Ga_4. *Journal of the Less Common Metals* **154,** 207–215 (1989).

355. Davey, T., Tran, N.-D., Saengdeejing, A. & Chen, Y. First-principles-only CALPHAD phase diagram of the solid aluminium-nickel (Al-Ni) system. *Calphad* **71,** 102008 (2020).

356. Silman, G. Compilative Fe–Ni phase diagram with author's correction. *Metal Science and Heat Treatment* **54,** 105–112 (2012).

357. Mishin, Y., Mehl, M. & Papaconstantopoulos, D. Phase stability in the Fe–Ni system: investigation by first-principles calculations and atomistic simulations. *Acta Materialia* **53,** 4029–4041 (2005).

358. Zhang, L. *et al.* Thermodynamic properties of the Al–Fe–Ni system acquired via a hybrid approach combining calorimetry, first-principles and CALPHAD. *Acta Materialia* **57,** 5324–5341 (2009).

359. Eleno, L., Frisk, K. & Schneider, A. Assessment of the Fe–Ni–Al system. *Intermetallics* **14,** 1276–1290 (2006).

360. Lippmann, S. *et al.* Synthesis of pure intermetallic phases on the example of the ternary phase τ_1 in the system Al-Fe-Ni. *Intermetallics* **105,** 107–112 (2019).

361. Lemmerz, U., Grushko, B., Freiburg, C. & Jansen, M. Study of decagonal quasicrystalline phase formation in the Al-Ni-Fe alloy system. *Philosophical Magazine Letters* **69,** 141–146 (1994).

362. Okamoto, H., Schlesinger, M. & Mueller, E. *Alloy Phase Diagrams* (ASM International, Novelty, 2016).

363. Witusiewicz, V. *et al.* Experimental study and thermodynamic re-assessment of the binary Fe–Ta system. *Intermetallics* **19,** 1059–1075 (2011).

364. Rajkumar, V. & Kumar, K. H. Gibbs energy modeling of Fe–Ta system by Calphad method assisted by experiments and *ab initio* calculations. *Calphad* **48,** 157–165 (2015).

365. Du, Z.-T., Zou, L., Guo, C.-P., Ren, X.-P. & Li, C.-R. Phase stability of $Fe_{23}Zr_6$ and thermodynamic reassessment of Fe–Zr system. *Journal of Iron and Steel Research International,* 1375–1389 (2021).

366. Stein, F., Sauthoff, G. & Palm, M. Experimental determination of intermetallic phases, phase equilibria, and invariant reaction temperatures in the Fe-Zr system. *Journal of Phase Equilibria* **23,** 480–494 (2002).

367. Ohodnicki Jr, P., Cates, N., Laughlin, D., McHenry, M. & Widom, M. *Ab initio* theoretical study of magnetization and phase stability of the $(Fe, Co, Ni)_{23}B_6$ and $(Fe, Co, Ni)_{23}Zr_6$ structures of $Cr_{23}C_6$ and $Mn_{23}Th_6$ prototypes. *Physical Review B* **78,** 144414 (2008).

368. Mihalkovič, M. & Widom, M. *Ab initio* Calculations of cohesive energies of Fe-based glass-forming alloys. *Physical Review B* **70,** 144107 (2004).

369. Lafaye, P., Toffolon-Masclet, C., Crivello, J.-C. & Joubert, J.-M. Thermodynamic modelling of the Fe–Sn–Zr system based on new experiments and first-principles calculations. *Journal of Alloys and Compounds* **821,** 153200 (2020).

370. Gao, F. *et al.* Thermodynamic modeling of the Bi–Ca and Bi–Zr systems. *International Journal of Materials Research* **113,** 419–427 (2022).

371. Okamoto, H. The Bi-Zr (bismuth-zirconium) system. *Bulletin of Alloy Phase Diagrams* **11,** 295–297 (1990).

372. Boström, M., Lind, H., Lidin, S., Niewa, R. & Grin, Y. Synthesis, crystal structure, phase relations and chemical bonding analysis of the new Nowotny chimney-ladder compound $ZrBi_{1.62}$. *Solid State Sciences* **8,** 1173–1180 (2006).

373. Mihalkovič, M. & Widom, M. Structure and stability of Al_2Fe and Al_5Fe_2: first-principles total energy and phonon calculations. *Physical Review B* **85,** 014113 (2012).

374. Yang, C. *et al.* Large language models as optimizers. *arXiv preprint arXiv:2309.03409 [cs.LG]* (2023).