Atomic Scale Simulations for Liquid Metals and Alloys: Machine Learning Potentials and Feature Selection

Inaugural thesis presented to the Faculty of Mathematics and Natural Sciences
of Heinrich heine Universiy Düsseldorf
for the degree of
Doctor of Natural Sciences
by

Johannes Erik Sandberg
from Värmdö Sweden

Grenoble, December 2024

From the Department of Physics
of Heinrich Heine University Düsseldorf

Printed by permission of the Faculty of Mathematics and Natural Sciences of
Heinrich Heine University Düsseldorf

Examiners:

1. Prof. Thomas Voigtmann

2. Prof. Jürgen Horbach

Date of the oral defence: December 17$^{\text{th}}$ 2024

In memory of Johan Sandberg

I miss you

Acknowledgements

An academic work like this is not possible without the help of other people. While I have been told that I am a man of few words, I will here, to the best of my ability, use those few words to thank some of those who have helped, or supported me, along the way.

First and foremost I want to thank my supervisors, Prof. Noel Jakse, Prof. Thomas Voigtmann, and Dr. Emilie Devijver. This doctoral work could not have been completed without your invaluable guidance. I could not have hoped for a better team of PhD supervisors. Working with you has been truly a joy.

I have been fortunate to be part of the SOLIMAT project, and want to thank all of my collaborator there. Special thanks go out to Fan and Kathi, for always being willing to share their expertise on the experimental side of things. Beyond that, I wish to also thank any other researchers with whom I have had fruitful discussions, whether at workshops, conferences, or seminars.

When focusing on a doctoral work, it is easy to forget about ones social life. This is especially true when coming out of a global pandemic, and moving to a new country where you do not speak the local language. Fortunately, this ended up being not the case for me, and I thus want to thank all the good friends I have made in Grenoble, especially Ashna and João-Paulo, but also Sébastien, Otavio, Alaa, Anatoli, Kaoutar, among others. I am sad that I had to leave halfway through this project, but incredibly happy to have gotten to know all of you, and will forever look back fondly on my time in Grenoble.

Moving to Cologne took some adjustment, and so I would like to thank everyone at the institute for material physics in space who helped to create such a welcoming atmosphere there. I especially wish to thank my good friends Stephan and Linnea, for keeping me company at the institute. I wish you both the best of luck with your own doctoral work, and hope it is not too lonely there without me. Asbjørn and Leon deserve thanks for their work as PhD student representatives at the institute while I was there. Also I would like to thank Gwen, Mélanie, Nisha, Malo, Nuria, Will, and all other people who kept me company for Tuesday morning coffee (and the occasional tiramisoup).

I wish to thank the members of my thesis follow-up committee, Jörg Baschnagel and David Rodney, for helping to make sure that progress was made on this work. I would like also to thank all the members of my defense jury, for taking time out of their busy schedules to participate in my defense. Prof. Jörg Behler and Dr. Julien Lam deserve special thanks for agreeing to act as referees, for thoroughly reading this thesis, and taking the time to write the reports on it. Likewise Prof. Jürgen Horbach, and Prof. Thomas Voigtmann, for acting as examiners on the German side.

Finally I thank the German Academic Exchange Service, DAAD, for generously funding this work through the DLR-DAAD Fellowship No. 509, as well as german aerospace center, DLR, for funding during the final stretch of this work

Abstract

Understanding how atomic interactions give rise to the macroscopic properties of materials is a main goal of material science. To this end, molecular dynamics simulations are a powerful tool, allowing for directly following the trajectory of atoms in the simulated system, and being well suited for dynamical processes such as diffusion and the early stages of crystallization. This, however, requires the accurate modeling of the interaction between atoms. Empirical interatomic potentials are limited in accuracy, and often fail to reproduce experimental results. *Ab initio* simulation, while adhering more closely to the underlying quantum-mechanical origin of interactions, is severely limited in scalability, preventing its use in many critical applications. In the past decade, machine-learning potentials, trained on data from *ab initio* simulation, have become an important method for enabling scalable simulations with *ab initio* level accuracy. Still, practical *ab initio* methods rely on approximations, necessitating at some point a connection to be made to experimental results.

In this thesis I develop a machine-learned potential for the binary Al-Ni alloy, building upon a previously trained potential for pure Al. For this, the entire process is covered, from construction of the training dataset, to design and training of the potential in the Behler Parrinello high-dimensional neural network framework. The potentials are validated against experimental data for transport coefficients in the liquid state, and applied to the study of homogeneous crystal nucleation from the undercooled liquid, through large-scale molecular dynamics simulation far beyond the reach of *ab initio* simulation. A significant result of this is the elucidation of the origins of the nucleation pathway into the body-centered cubic B2 phase of equiatomic Al-Ni. I further implement an active feature selection method for such high-dimensional neural network potentials, based on the adaptive group lasso. This allows for reducing the number of input features, taking into account model predictions, allowing for training faster and more explainable potentials. Part of this is the training of a potential for Boron, serving as a particularly complex model system, useful for the evaluation of descriptors and machine-learning potential frameworks.

Résumé

Comprendre la relation entre les interactions à l'échelle atomique et les propriétés macroscopiques des matériaux est l'un des objectifs principaux en science des matériaux. Dans ce but, les simulations de dynamique moléculaire sont un outil puissant, permettant de suivre directement la trajectoire de phase des atomes et sont de ce fait bien adaptées pour les processus dynamiques tels que la diffusion et les premiers stades de la crystallisation. Cette approche nécessite cependant une modélisation précise des interactions entre les atomes. Les potentiels interatomiques empiriques sont limités en précision et reproduisent souvent assez mal certains résultats expérimentaux. Les simulations ab initio, fondés sur une description quantique des interactions, est sévèrement limitée en termes de taille et de temps simulés, ce qui limite son utilisation pour de nombreuses applications. Au cours de la dernière décennie, les potentiels basés sur apprentissage machine, entrainés sur des données de simulation ab initio, sont devenus une méthode importante pour permettre des simulations de grande ampleur avec une précision proche des calculs ab initio. Néanmoins, les méthodes ab initio reposent sur des approximations, nécessitant à un certain point une connexion aux résultats expérimentaux.

Dans cette thèse, je développe un potentiel par apprentissage machine pour les métaux purs et l'alliage binaire liquide Al-Ni. Pour ce faire, une procédure qui va de la construction de l'ensemble de données d'entraînement à la conception et la formation du potentiel au moyen d'un réseau de neurones de haute dimension proposé par Behler-Parrinello. Les potentiels sont validés par comparaison avec l'expérience et appliqués à l'étude de la nucléation cristalline homogène à partir du liquide en surfusion, par simulation de dynamique moléculaire de grande échelle, bien au-delà de la portée de la simulation ab initio. Un résultat significatif de ce travail est l'élucidation des origines du chemin de nucléation dans la phase cubique centrée B2 de Al-Ni à la composition équiatomique. Un accent particulier est également mis sur le développement d'une méthode de sélection adaptative des descripteurs pour cette approche appelée adaptative group lasso afin de permettre la construction de potentiels plus efficaces et plus explicables. Dans ce cadre, un potentiel pour le bore, qui sert de système modèle particulièrement complexe, utile pour l'évaluation des descripteurs et l'apprentissage machine pour les potentiels.

Contents

Abbreviations

ML Machine Learning

MLIP Machine Learning Interaction Potential

NN Neural Network

NNP Neural Network Potential

HDNNP High-Dimensional Neural Network Potential

EAM Embedded Atom Model

DFT Density Functional Theory

PES Potential Energy Surface

MD Molecular Dynamics

AIMD *Ab Initio* Molecular Dynamics

RDF Radial Distribution Function

MSD Mean Square Displacement

CNA Common Neighbor Analysis

SGD Stochastic Gradient Descent

LR Learning Rate

MSE Mean Square Error

SF Symmetry Function

ACSF Atom Centered Symmetry Function

GL Group Lasso

AGL Adaptive Group Lasso

CNT Classical Nucleation Theory

Chapter 1

General Introduction

A central aim of material science is to understand the microscopic origins of material properties. For metallic materials, as they are usually manufactured from a high-temperature melt, it is necessary to have a proper understanding of the liquid properties, and of the solidification process, and how this effects the microstructure of the solid [1]. A key process to understand is the earliest stages of crystallization, or nucleation, when microscopic crystal nuclei first emerge from the undercooled melt [2]. Due to the transient nature of nucleation events, and it being an inherently atomic-scale phenomenon, nucleation is difficult to study experimentally. As such, molecular dynamics simulation is a key tool for obtaining a deeper understanding of nucleation [3], with the ultimate aim of being able to control nucleation rates and pathways, thus obtaining new ways to design materials through control of their microstructure.

Molecular dynamics simulations require accurate models of the interatomic interaction. Furthermore, for solidification, it is crucial to accurately represent both the solid and liquid. While classical potentials are often applied to nucleation, as their computational performance allow for large scale simulations to observe rare nucleation events, their lack of transferability between phases poses a problem in this setting [3]. *Ab initio* simulation [4], being closer to the quantum-mechanical origins of the interatomic interaction, often achieve better accuracy, and are inherently more transferable. Such simulations are, however, severely limited to relatively small system sizes. This makes direct *ab initio* molecular dynamics studies of nucleation virtually impossible.

In order to overcome the limitations of *ab initio*, Machine Learning (ML) is nowadays frequently used [5, 6, 7, 8]. By training ML models, such as neural networks [9] and Gaussian processes [10], on *ab initio* energies and forces, it is possible to accelerate quantum accurate simulations, nearing the speed of classical potentials. Such Machine-Learned Interaction Potentials (MLIPs) have opened up new possibilities of studying nucleation phenomena at the level of *ab initio* [8, 11].

While much quicker than *ab initio*, there is still reason for optimizing for faster MLIPs, especially in the context of nucleation. This is a major motivator for a careful choice of the descriptors used as input to the ML models. This can be done in a data-driven way, through feature selection, a collection of methods for selecting which input features of a ML model are the most important. Such methods were previously applied for linear potentials [12], based on

the Lasso [13], but were not applicable to the NN Potential (NNP) case, due to the nonlinear nature of the estimator. Other feature selection methods proposed for nonlinear MLIPs have mainly consisted of filters [14], acting only on the dataset, without taking into account model predictions during selection. This motivates the need for embedded methods, that allow for selecting features of NNPs as part of the training process. Beyond allowing for faster potentials, the use of feature selection is a potential first step towards more explainable models. It has also been shown that feature selection can improve the generalization performance of linearized potentials [15], opening for the possibility of similar behavior in NNPs.

The primary research aim of this thesis is to use MLIPs to study the properties and behavior of liquid metals and alloys. Specifically, we focus on pure Al and Al-Ni as model systems. In particular, we choose to study the homogeneous nucleation process in these systems. This consists of both practical work, training potentials and applying them in simulations, and methodological development of a feature selection method for HDNNPs.

The remainder of this thesis will be structured as follows. The next chapter will provide further background, and introduce the primary methods used in the thesis. Chapters 3 through 5 constitute the main part of the thesis, in the form of a set of papers produced during the research that has led up to this dissertation. Chapter 3 consists of a study of homogeneous nucleation in pure Al, studied using a HDNNP [16]. This is followed up in Chapter 4 with a similar study for binary Al-Ni, with a HDNNP trained as part of this thesis [17]. Chapter 5 covers our work on developing an adaptive feature selection method for HDNNPs, based on the adaptive group lasso [18]. Finally, Chapter 6 provides the main conclusions, and outlook.

Chapter 2

Methods and Background

2.1 Introduction

Macroscopic quantities of material generally contain on the order of 10^{23} atoms, give or take a few orders of magnitude. Individually describing the state of each is impossible. However, through the introduction of statistical mechanics, in fact predating the discovery of the atom, one can still make progress, by using statistical methods to study the aggregate behavior of the system. Ultimately, what determines the macroscopic properties of an atomic or molecular system is the interaction between the atoms, often modeled as an interatomic potential. Early interatomic potentials only included two-body terms. Such pair potentials can be very computationally efficient, and potentials such as hard-spheres and the Lennard-Jones potential are still in use today as model systems, used to gain qualitative and general insights. This efficiency comes at a cost, however, as ignoring the many-body interactions, arising from the underlying many-body electron system, severely limits the validity of pair-potentials. Only in some cases, such as noble gasses, are quantitatively accurate parameterizations found, and it is known that some properties cannot be captured by any pair potential [19], necessitating the inclusion of explicit many-body interactions in more sophisticated potentials. This led to the development of many-body potentials. For metals, the Embedded Atom Model [20] (EAM) potentials have been highly successful. Nevertheless, their accuracy is often limited.

As interatomic interactions are primarily electronic in origin, an exact treatment requires the solution of the combined quantum-mechanical system of electrons and atoms. This is in practice not possible, but a significant simplification is offered by the Born-Oppenheimer, or adiabatic, approximation [21], assuming that the comparably low-mass electrons act on a sufficiently smaller time-scale compared to the atoms, allowing for separating the two and treating them separately. One can then solve the electronic system for a static ionic background. As the electronic system is assumed to instantaneously relax on the timescale of the ionic movements, the forces on the ions can be obtained via the Feynman-Hellman theorem [22]. In principle, classical interatomic potentials try to emulate the potential energy surface one would obtain like this. A more direct approach, however, is to actually solve the electronic structure problem. Such *Ab Initio* Molecular Dynamics (AIMD) was originally

developed by Car and Parrinello [23] in the mid 1980s, around the same time as many-body potentials started to appear.

While treating the ions as a classical system, separately from the electronic one, is fundamental to AIMD, the electronic system can in general not be exactly solved, requiring further approximations. A range of methods exists for solving the electronic structure problem, such as the Hartree-Fock method [24], Density Functional Theory (DFT) [25, 26], coupled cluster calculations [27], to name a few. For simulations of liquid metals, the most common approach is DFT. Originally developed in the 1960s by Hohenberg and Kohn, earning the latter a Nobel prize in 1998 [28], DFT is in principle an exact reformulation of quantum mechanics in terms of the particle density, as opposed to the wave function. Through the Hohenberg-Kohn theorems, all ground-state properties can be regarded as functionals of the ground-state density. Besides being a fundamental insight into the nature of quantum mechanics, interesting in it's own regard, the density is inherently a much simpler object than the wave-function. The downside is that, although the theory is exact, and the density functional provably exists, its exact form is unknown.

As they explicitly model the physics behind interatomic interactions, *ab initio* calculations typically have a much closer agreement with real systems, compared to classical potentials. They also benefit from the fact that the electrons are treated on an equal footing, allowing for arbitrary combinations of atomic species, in arbitrary configurations; AIMD is an inherently transferable method. This is in sharp contrast to classical potentials, which are designed to specific combinations of atom types, and often have poor transferability [29]. Nevertheless, these benefits over classical potentials comes with a vastly higher computational cost, and much worse scaling with system size. While simulations with classical potentials like EAM scale almost linearly with number of atoms, and are regularly used to simulate millions of atoms over nanosecond timescales [30, 31], AIMD simulations rarely go beyond a few hundred atoms, and typically reach picosecond timescales. During the past two decades, a new paradigm has emerged, wherein ML methods are applied to train potentials on *ab initio* data. Such MLIPs offer a way to bridge between classical potentials and *ab initio*.

Although having a long history [32], in the last decade ML has seen widespread adoption in many areas of society. This is also the case in the sciences. A major driving factor for this interest has been the successes of deep learning, enabled by the development of new techniques, better hardware, and access to larger datasets. While Neural Networks (NNs), the technology behind deep learning, has been the focus of much of the recent interest in ML, many other methods exist. Linear models still serve a purpose in many situations where explainability is key, as despite work towards making them more explainable, NNs are still largely considered black-box methods.

Early methods for constructing the Potential Energy Surface (PES) using ML trace back to the 1990s [33], but were limited in applicability. These early ML potentials relied on a single NN for predicting the potential energy of the atomic system, using a set of descriptors. A number of drawbacks exist for these early methods. There exist a number of symmetries that the PES must satisfy. First, it is invariant under the $E3$ symmetry group, such that translations and rotations of the system must yield identical predictions of the

energy. While this can be ensured by utilizing an embedding into $E3$-invariant descriptors, a more problematic issue is permutation invariance. Swapping two same-species atoms yields an equivalent configuration, but swapping two inputs of a NN, corresponding to descriptors of different atoms, will not, in general, satisfy this symmetry. As the number of inputs to a NN is fixed, these early ML potentials are applicable only to fixed-size systems, and were in practice limited to small systems The major breakthrough, ultimately setting the stage for the current proliferation of MLIPs, came with the introduction of High-Dimensional NN Potentials (HDNNPs) by Behler and Parrinello in 2007 [9]. In their approach, the PES is approximated as a sum over local atomic contributions, which in turn can be expressed via a NN, taking as input an atomic fingerprint describing the local atomic environment around a central atom. By letting the atomic fingerprint be invariant under required symmetries, this approach leads to an $E3$ and permutation invariant approximate PES, that can be scaled up to arbitrary system sizes.

The Behler Parrinello HDNNP marks the transition to what has been termed the second generation of NNPs [34], and led to a proliferation in methods for constructing MLIPs. While early works were mainly using NNs, following the introduction of HDNNPs, other ML methods started to be applied. These include kernel-based models such as the Gaussian Approximation Potential [10], and linear potentials [12, 35]. A variety of descriptors have been likewise proposed, beyond the original Atom Centered Symmetry Functions (ACSFs) [36], with examples including Smooth Overlap of Atomic Position [37] and the Atomic Cluster Expansion [38]. Beyond predefined descriptor types, the last decade has also seen the introduction of descriptor-free methods such as the Deep Potentials [39], and various graph-NN-based models [40]. These models can be regarded as still utilizing a form of descriptors, although ones learned as part of training. Recent developments have also led to equivariant models, showing promising results in terms of accuracy, data efficiency, and stability [41, 42].

In addition to new types of potential, and new descriptors, an important development has been in the introduction of active learning to construct, and iterate on, training datasets for MLIPs [43]. Nowadays there are a number of accessible frameworks for training potentials of different types. Likewise, many frameworks for sharing data, and also potentials, are starting to appear. With this maturing of the field, MLIPs are starting to see widespread use to tackle problems in material science.

One area in which there is great potential for MLIPs to overcome the inherent limitations of both classical and *ab initio* simulations, is the study of solidification, and in particular nucleation [3]. As liquids are cooled below their melting temperature, the liquid becomes meta-stable, and will eventually transition to a solid. The phase transition from undercooled liquid to solid is initiated at some location, with the formation of a crystal seed, which then grows, driving the phase transition of the full system. Often this process is initiated by impurities, termed heterogeneous nucleation. Homogeneous nucleation, the spontaneous formation of nuclei in the precursor phase, due to thermodynamic fluctuations, is a much more rare event. As it is an inherently microscopic process, atomistic simulations is a relevant tool for studying this process, but suffer from a few significant drawbacks. Homogeneous nucleation involves rare nucleation events, with the undercooled liquid being metastable for

often very long time scales, even at deep undercooling. This is in sharp contrast with the typically rapid solidification that occurs after the nucleation event. As such, simulations over very long timescales, measured in nanoseconds, and use of very large systems, ideally several millions of atoms, are necessary. As such, nucleation is far out of reach for *ab initio*. However, as nucleation necessarily involves multiple phases, and transitions between them, with potentially a great sensitivity to the details of the potential for obtaining quantitatively accurate results, classical potentials are also not ideally suited. There is thus a very relevant use case here for MLIPs.

2.2 Molecular Dynamics Simulations

2.2.1 Classical MD

In Molecular Dynamics (MD) simulations, the dynamics of the atomic system is simulated, following the motions of all the particles as they move over time. The state of the system, at a given time t, will be described by the positions $\mathbf{r}$, and velocities $\mathbf{v}$, of all particles in the system. In some systems, additional degrees of freedom can also be included, for instance magnetic moments, or internal degrees of freedom in coarse-grained models.We will, however, here only consider the basic setting where the system is fully defined by the positions and velocities. Although computer technology has seen exponential improvements in memory and processing power for the last decades, storing the phase-space position of 10^{23} atoms in a computer memory, let alone performing calculations on them, is clearly impossible. As such, simulated systems are necessarily small compared to their real-world analogues. One important consideration is the boundary of the system. In macroscopic systems, boundary effects can typically be ignored, due to the vanishingly small ratio of surface atoms relative to bulk atoms. To avoid the introduction of a surface in the simulated system, it is common to impose periodic boundary conditions, emulating more closely the bulk system.

With some model of the interaction between atoms, in the form of forces between them, often expressed via an interatomic potential $V(\mathbf{r})$, the trajectory of the system is obtained by solving Newton's equations of motion, numerically. Perhaps the most common integration scheme for MD simulations is the Verlet algorithm [44], in the velocity form. The positions $\mathbf{r}(t)$, velocities $\mathbf{v}(t)$, and accelerations $\mathbf{a}(t)$, are iterated as [45]

$$\mathbf{v}(t + \frac{1}{2}\Delta t) = \mathbf{v}(t) + \frac{1}{2}\mathbf{a}(t)\Delta t \tag{2.1}$$

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t + \frac{1}{2}\Delta t)\Delta t \tag{2.2}$$

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t + \frac{1}{2}\Delta t) + \frac{1}{2}\mathbf{a}(t + \Delta t)\Delta t. \tag{2.3}$$

The trajectory of the systems, as time series of the particle positions, and possibly velocities, can be further analyzed to calculate physical properties of the system. Thermodynamic quantities are typically also of interest. The potential and kinetic energies are straightforwardly

obtained from the interatomic potential, and the velocities and masses of the particles. Temperature is usually obtained from the kinetic energy K, using the equipartition principle

$$K = \frac{3}{2} N k_B T, \tag{2.4}$$

and pressure is typically obtained via the virial equation,

$$PV = N k_B T + \frac{1}{3} \left\langle \sum_{i=1}^{N} \mathbf{r}_i \cdot \mathbf{F}_i \right\rangle, \tag{2.5}$$

where $\mathbf{F}_i$ is the force acting on particle i, and the brackets denote ensemble averaging. Here, the volume is well defined by the simulation box and the periodic boundary condition. Often one wants to control the temperature and pressure in a simulation, or to simulate different ensembles such as NVT or NPT. The temperature is typically controlled via a thermostat. One of the most commonly used thermostats is the Nosé-Hoover one, which adds an additional degree of freedom to the system, representing an external heat bath [46]. Similarly, the volume of the simulation box can be treated as a degree of freedom, allowing for the pressure to be controlled.

Static Quantities

The Radial Distribution Function (RDF) $g(r)$ describes the probability of finding a particle at distance r from another particle at $r = 0$. In the case of multi-component systems, there exists also for each pair of species $\mu\nu$ a partial RDF $g_{\mu\nu}(r)$, which is the same, but for finding a particle of type μ around a central particle of type ν. Formally, the RDF is given by [47]

$$\rho g(r) = \frac{1}{N} \left\langle \sum_{i}^{N} \sum_{j=1, j\neq i}^{N} \delta(r - r_{ij}) \right\rangle, \tag{2.6}$$

with r_{ij} the distance between atoms i and j, and ρ the number density of the system. Likewise for a multicomponent system

$$x_\mu x_\nu \rho g_{\mu\nu}(r) = \frac{1}{N} \left\langle \sum_{i}^{N_\mu} \sum_{j=1}^{N_\nu} \delta(r - r_{ij}) \right\rangle, \tag{2.7}$$

with $x_\mu = N_\mu / N$ the fraction of the system that is of type μ. In practice, these can be computed from snapshots of a MD simulation as a histogram of inter-particle distances, averaged over all particles in the system, and over several timesteps. An important use of the RDF is to calculate the mean coordination number, by integrating

$$4\pi r^2 \rho g(r) \tag{2.8}$$

up to the first minimum of $g(r)$. Second and third coordination numbers, and so on, can likewise be obtained by integrating to the second and third minima, respectively. Being a

very straightforward quantity to calculate during simulation, the RDF can often serve as a first test of the ability of MLIPs to reproduce physical properties.

The RDF can be regarded as the real-space density-density correlation function. Considering instead the reciprocal space, one obtains the static structure factor

$$S(\mathbf{k}) = \frac{1}{N} \left\langle \sum_{ij} \exp i\mathbf{k} \cdot (\mathbf{r}_i - \mathbf{r}_j) \right\rangle . \tag{2.9}$$

In isotropic liquids, this depends only on the magnitude of the wave vector, such that $S(\mathbf{k}) = S(k)$. The structure factor can be related to the Fourier transform of the RDF as

$$S(k) = 1 + \rho h(k), \tag{2.10}$$

where $h(r) = g(r) - 1$. This quantity is particularly important, as it can be directly related to measurements in scattering experiments.

In the multicomponent case, analogously to the RDF, one can define, in several ways, a partial structure factor. In analogy to (2.9), one can write [47]

$$S_{\mu\nu}(q) = \frac{1}{N} \left\langle \sum_{i=1}^{N_\mu} \sum_{j=1}^{N_\nu} \exp \left[i\mathbf{q} \cdot (\mathbf{r}_i - \mathbf{r}_j) \right] \right\rangle . \tag{2.11}$$

From these partial structure factors, the Bhatia-Thornton structure factors [48] can be obtained as

$$S_{nn}(q) = S_{11}(q) + 2S_{12}(q) + S_{22}(q) \tag{2.12}$$
$$S_{nc}(q) = x_2 S_{11}(q) - x_1 S_{22}(q) + (x_2 - x_1) S_{12}(q) \tag{2.13}$$
$$S_{cc}(q) = x_2^2 S_{11}(q) + x_1^2 S_{22}(q) - 2x_1 x_2 S_{12}(q), \tag{2.14}$$

which capture the reciprocal space correlations between fluctuations of particle number and concentrations. Notably, the long-wavelength limit of the concentration-concentration structure factor allows to analyze the mixing behavior and chemical ordering of mixtures [49], and is used to calculate the interdiffusion in binary mixtures.

Dynamic Quantities

As the evolution of the system is observed over time, MD simulations are well suited for the study of dynamics, and transport coefficients. One of the simplest such quantities is the self-diffusion coefficient. This can be calculated through the Einstein relation, via the Mean Square Displacement (MSD)

$$R_\mu^2(t) = \frac{1}{N_\mu} \sum_{i=1}^{N_\mu} \left\langle \left[\mathbf{r}_i(t + t_0) - \mathbf{r}_i(t_0) \right]^2 \right\rangle_{t_0} , \tag{2.15}$$

where the brackets indicate averaging over the starting time t_0. This is asymptotically $\propto t$, for large times, with the prefactor determining the self-diffusion coefficient

$$D_\mu = \lim_{t\to\infty} \frac{1}{6t} R_\mu^2(t). \tag{2.16}$$

In the binary system, in addition to the self-diffusion of each species, there is also the interdiffusion constant, which describes the diffusion process driven by concentration gradients. Like the self-diffusion constants, this can be calculated from an Einstein relation, but due to the collective nature of the quantity one can no longer average over atoms, thus it is often more useful to calculate the interdiffusion via a Green-Kubo relation. This can be done by calculating the concentration-concentration current

$$J_{\mu\nu}(t) = x_\nu \sum_{i=1}^{N_\mu} \mathbf{v}_i(t) - x_\mu \sum_{i=1}^{N_\nu} \mathbf{v}_i(t). \tag{2.17}$$

From this one obtains the correlation function

$$C_{\mu\nu} = \langle J_{\mu\nu}(t+t_0) \cdot J_{\mu\nu}(t_0) \rangle_{t_0}, \tag{2.18}$$

the integral of which determines the interdiffusion constant

$$D_{\mu\nu} = \frac{1}{3NS_{cc}(k=0)} \int_0^\infty C_{\mu\nu}(t)dt, \tag{2.19}$$

where S_{cc} is the Bhatia-Thornton concentration-concentration structure factor (2.14). The inter-diffusion coefficient can be factored into a purely kinetic part, called the Onsager factor

$$L = \frac{1}{3Nx_ix_j} \int_0^\infty C_{ij}(t)dt, \tag{2.20}$$

and the thermodynamic factor

$$\Phi = \frac{x_ix_j}{S_{CC}(q\to 0)}, \tag{2.21}$$

capturing the two primary different contributions to the interdiffusion process.

The shear viscosity can also be calculated from MD simulations, in a variety of ways [50]. One of the more straightforward methods is via the off-diagonal stress-stress correlation function [51]. As for the interdiffusion, the Green-Kubo approach is often more useful in practice,

$$\eta = \frac{V}{k_BT} \int_0^\infty \langle P_{xy}(t_0) P_{xy}(t_0+t) \rangle_{t_0} \, dt. \tag{2.22}$$

In an isotropic liquid, the different components xy, xz, and yz, can be used interchangeably, and averaged over to improve the statistics.

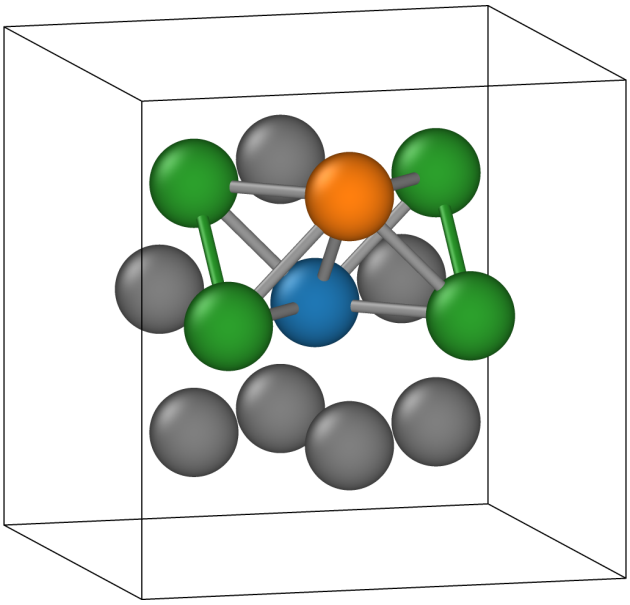

Figure 2.1: Atom (blue), and its nearest neighbors in an FCC crystal, illustrating the common neighbor analysis. The bond with a neighboring atom (orange) is analyzed based on shared neighbors and the bonds between them (green), as detailed in the text. The CNA indices are, in this case, $(4, 2, 1)$.

Structure Identification

In the analysis of a MD simulation, it is often useful to be able to distinguish between different structures, as well as to identify specific structures of a given type. Some examples include identifying crystal nuclei from the undercooled liquid during liquid-crystal phase transitions [52, 53, 30], identification of crystal defects [54], identification of local structural orderings in the liquid phase [55], to name a few. There exists a variety of different methods for this. While recently, ML has started to see use to tackle this problem [53], traditional structure analysis methods are still the norm.

The Common Neighbor Analysis (CNA) method [56] assigns to each bond between two atoms a set of indices, based on the shared neighbors of the two atoms. Typically, these bonds are assigned based on the first neighbor shell, extracted from the pair distribution in the liquid. Different schemes can be used to assign indices to a given bond, but following the indexing of Faken and Jónsson [52], three indices are assigned. The first index is simply the number of shared neighbors of the pair, i.e. the number of atoms which are bonded to both. The second index is the total number of bonds between the shared neighbors. Finally, the third index is the longest chain of bonded shared neighbors. By computing the indices of all bonds of an atom, one can compare to the indices of reference structures. An example for an FCC crystal is illustrated in Figure 2.1, showing a central atom, marked in blue, and its nearest neighbors. Here, the orange atom has four shared neighbors with the central atom, marked green, and there are two bonds between the shared neighbors, marked also in green,

with the longest unbroken line of bonds between the shared neighbors being of length one. Thus the index assigned to the bond between the blue and orange atom is $(4, 2, 1)$, which serves as a signature of bonds in an FCC structure.

Another technique for structure analysis is given by the Steinhardt bond-order parameters [57]. Each atom i is here assigned a set of numbers

$$q_{lm}(i) = \frac{1}{N(i)} \sum_{j=1}^{N(i)} Y_{lm}(\mathbf{r}_{ij}), \tag{2.23}$$

with j indexing the $N(i)$ nearest neighbors of atom i, $\mathbf{r}_{ij}$ being the displacement of nearest neighbor j from i, and Y_{lm} being the spherical harmonics. As such, these numbers are essentially the average spherical harmonics of the bond of the central atom. From these, the rotationally invariant bond-order parameters can be calculated as

$$q_l = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^{l} |q_{lm}|^2}. \tag{2.24}$$

Different crystal structures will have different distributions of the parameters, with q_4 and q_6 being commonly used to distinguish the liquid and different crystal structures. The parameters in (2.24) have the drawback, however, that their distributions over different common crystal structures (FCC, HCP, BCC) have a large overlap between each other, and with the liquid [58]. This can be improved upon by averaging the values of q_l over bonded neighboring atoms, also carrying information on the second neighbor shell. These averaged bond-order parameters are then given as

$$q_l = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^{l} \left| \frac{1}{N(i)+1} \sum_{k=0}^{N(i)} q_{lm}(k) \right|^2}, \tag{2.25}$$

where the inner sum from $k = 0$ to $N(i)$ runs over all neighboring atoms, and the central atom itself. This averaging provides much sharper distributions, with less overlap allowing for clearer distinction between different crystal phases and the liquid [58]. As for CNA, it can also be beneficial to use relaxed configuration, in order to extract the inherent structures and reduce the impact of thermal noise.

2.2.2 Ab Initio MD

At the microscopic level, with no externally applied fields, the combined ion-electron system is described by the following Hamiltonian:

$$H = -\frac{\hbar^2}{2m_e} \sum_i \nabla_i^2 - \sum_I \frac{\hbar^2}{2m_I} \nabla_I^2 + \frac{1}{2} \sum_{i \neq j} \frac{e^2}{R_{ij}} + \sum_{i,I} \frac{Z_I e^2}{R_{iI}} + \frac{1}{2} \sum_{I \neq J} \frac{Z_I Z_J e^2}{R_{IJ}}. \tag{2.26}$$

Here indices i, I run over the electrons and ions in the system, respectively. The first two terms provide the kinetic energies of the electrons and ions respectively, with m_e being the electron mass, and m_I the mass of ion I. The last three terms, in turn, provide the Coulomb interaction between electrons, electrons and nuclei, and between nuclei. Here, Z_I is the atomic number of ion I, e is the unit charge, and $R_{ij,iI,IJ}$ is the absolute distance between electrons i, j, electron i and ion I, and ions I, J, respectively. While this Hamiltonian serves as the basis for the atomic scale interactions in a material, neglecting relativistic and quantum electrodynamic effects, solving this combined ion-electron problem quantum-mechanically is not typically feasible. This is often not necessary, however, due to the large difference in mass between the electron and most ions, allowing for an approximate separation into an electronic and an ionic Hamiltonian, through the Born-Oppenheimer approximation [21]. Due to the much smaller electron mass, the electronic system acts on a significantly shorter timescale compared to the ions, which can be treated as stationary for the purposes of solving the electronic structure problem.

While a fully quantum-mechanical treatment of the electronic degrees of freedom is necessary, the ions in most materials are heavy enough that they can to a good approximation be treated as classical particles. An argument for the validity of this classical approximation, following [47], can be made using the thermal wavelength

$$\Lambda = \sqrt{\frac{2\pi\hbar^2}{k_B T m_I}}. \tag{2.27}$$

Using room-temperature Al as a test case, one finds that $\Lambda \approx 0.2$ Å, which which is sufficiently shorter than the typical interatomic distances that the atoms can be taken as classical without too much worry.

With the basic setting for AIMD now in place, the question now remains, how does one solve the electronic structure problem? In the case of liquid metals, the de-facto standard method used is Kohn-Sham DFT [26]. DFT is at it's core a reformulation of quantum mechanics, in which all observables are viewed as functionals of the particle density. This is, in turn, founded on the two Hohenberg-Kohn theorems [25], the proofs of which are tantalizingly simple. The first theorem states that for any system of interacting particles, in an external potential $V(\mathbf{r})$, the ground state density $n(\mathbf{r})$ uniquely determines $V(\mathbf{r})$, up to a constant. Since the potential, in turn, fully determines the Hamiltonian, which in turn determines all the properties of the system, it follows that all ground state properties are determined by the ground state density. The second Hohenberg-Kohn theorem states that there exists a universal functional of the density, $F_{HK}[n]$, such that for any external potential $V(\mathrm{r})$, the functional $E[n] = F_{HK}[n] + \int V(\mathrm{r})n(\mathrm{r})d^3r$ has a global minimum that is the ground state energy, and is minimized by the ground state density. In practice, however, while the functional $F_{HK}[n]$ must exist, it's actual form is unknown.

The step towards making DFT into more than just a interesting theoretical curiosity, but one of the primary tools of electronic structure calculations, came with the Kohn-Sham self-consistent scheme [26]. The basic idea of this scheme is to introduce an auxiliary system of non-interacting particles, subject to some some unknown potential V_{KS}, such that the

density is the same as in the interacting system. In the auxiliary system, the energy functional becomes

$$E_{KS}[n] = T_{KS}[n] + \int V(\mathrm{r})n(\mathrm{r})d^3r + E_H[n] + E_{xc}[n], \tag{2.28}$$

with T_{KS} the kinetic energy of the noninteracting particles, and E_H being the Hartree energy

$$E_H[n] = \frac{1}{2}\int \frac{n(\mathrm{r})n(\mathrm{r}')}{|\mathrm{r}-\mathrm{r}'|}d^3rd^3r'. \tag{2.29}$$

The final term, E_{xc}, contains all the effects of exchange and correlation, not otherwise present in the noninteracting system. The potential V_{KS} can then be written as

$$V_{KS}(\mathrm{r}) = V(\mathrm{r}) + V_H(\mathrm{r}) + V_{xc}(\mathrm{r}), \tag{2.30}$$

with $V_H(\mathrm{r})$, and $V_{xc}(\mathrm{r})$ being defined as the functional derivatives of $E_H[n]$ and $E_{xc}[n]$ respectively.

The question now remains, how to, in practice, calculate the xc potential. While in principle DFT is an exact theory, it is here that it becomes inexact, in the sense that some approximation has to be made for V_{xc}. One of the earliest such approximations is the Local Density Approximation (LDA), proposed already by Kohn and Sham in their foundational paper [26]. In LDA, the exchange correlation energy is assumed to be given as

$$E_{xc}^{\mathrm{LDA}}[n] = \int \varepsilon_{xc}^{\mathrm{hom}}(n(\mathrm{r}))d^3r, \tag{2.31}$$

with $\varepsilon_{xc}^{\mathrm{hom}}(n)$ being the exchange correlation energy density of the homogeneous electron gas, with density n. In other words, the local xc energy contribution of each point in space, is taken to be the same as in the uniform electron gas with the same density as at that point. Although, arguably, the simplest possible approximation, LDA is surprisingly good in practice. Another common form is the Generalized Gradient Approximation (GGA), in which the local exchange correlation density depends on not only the density, but also on it's gradients. A variety of different GGA functionals exist in the literature, such as the Perdew Wang [59], and PBE [60] functionals, among others. While LDA and GGA are often sufficient for simulations of liquid metals, and thus for the work presented herein, they are generally not sufficiently accurate for more complex systems. As with so many other fields, DFT has in recent years seen the introduction of ML methods, used to construct potentially more accurate functionals [61, 62].

With a suitable approximation for the xc functional, and corresponding xc potential, in place, the auxiliary Kohn Sham system can be solved. This is generally done in a self-consistent manner, wherein an initial guess is made for the ground state density. This is then used to compute the effective potential, for which the noninteracting system is solved. From the solution one obtains a new ground state density, which can be fed back into the procedure as a new initial guess. This process is iterated until self-consistency is reached.

The way AIMD is often performed in practice, is through the Car-Parrinello method [23]. In this method, the dynamics of the system is based on the Lagrangian [63]

$$\mathcal{L}_{CP} = \frac{1}{2}\sum_I m_I \dot{\mathbf{R}}_I^2 + \frac{\mu}{2}\sum_i < \dot{\Phi}_i|\dot{\Phi}_i > -E(\mathbf{R},\Phi) + \sum_{ij}\lambda_{ij}(< \Phi_i|\Phi_i > -\delta_{ij}), \tag{2.32}$$

with Φ the electron orbitals, often in a plane-wave basis, μ a parameter, $E(\mathbf{R},\Phi)$ the energy functional, with the ion-ion and ion-electron electrostatic interactions added, and λ_{ij} Euler constants introduced to ensure orthogonality of the orbitals. This results in the equations of motion

$$M_I\ddot{\mathbf{R}}_I = -\frac{\partial E(\mathbf{R},\Phi)}{\partial \mathbf{R}_I} + \sum_{ij}\lambda_{ij}\frac{\partial}{\partial \mathbf{R}_I} < \Phi_i|\Phi_j > \tag{2.33}$$

$$\mu\ddot{\Phi}_i = -H(\mathbf{R},\Phi)\Phi_i + \sum_i \lambda_{ij}\Phi_j. \tag{2.34}$$

Here, $H(\mathbf{R},\Phi)$ is the Kohn-Sham single-particle Hamiltonian, with the external field defined by the ionic positions. The constant of motion, in this case, is no longer just the total energy of the ions $\frac{1}{2}\sum_I m_I\dot{\mathbf{R}}_I^2 + E(\mathbf{R},\Phi)\hat{=}E_{\text{phys}}$, but also contains the kinetic energy of the electronic system $T_e = \frac{\mu}{2}\sum_i < \dot{\Phi}_i|\dot{\Phi}_i >$, such that $E_{\text{phys}} + T_e$ is conserved during time evolution of the system.

Despite the many benefits of AIMD over classical MD, including the closer connection to the underlying physics behind the interatomic interaction (2.26), accuracy, transferability, and so on, the need to introduce electronic degrees of freedom vastly increases the computational demand of the method, with cubic scaling in the number of atoms, as a rule of thumb [64]. In practical calculations, only a few hundred atoms can be treated, largely dependent on the number of valence electrons in the species under consideration. The next section will be focused on how to circumvent this issue.

2.3 Machine Learning Interatomic Potentials

2.3.1 Feed Forward Neural Networks

A NN can be considered as a collection of computational units, called neurons. Each neuron implements a simple computation, taking a vector of inputs $\mathbf{x}$, multiplying by a vector of weights $\mathbf{w}$, adding a bias w_0, and passing it through a nonlinear activation function f, returning

$$y = f\left(\mathbf{w}^T\mathbf{x} + w_0\right). \tag{2.35}$$

These neurons are connected together, with the output of some neurons being used as an input to others, to construct a NN. In practice, some ways of connecting the neuron are more useful than others, with one of the most common being the feed-forward NN. In a feed-forward NN, the neurons are organized into layers, with the neurons of each layer taking

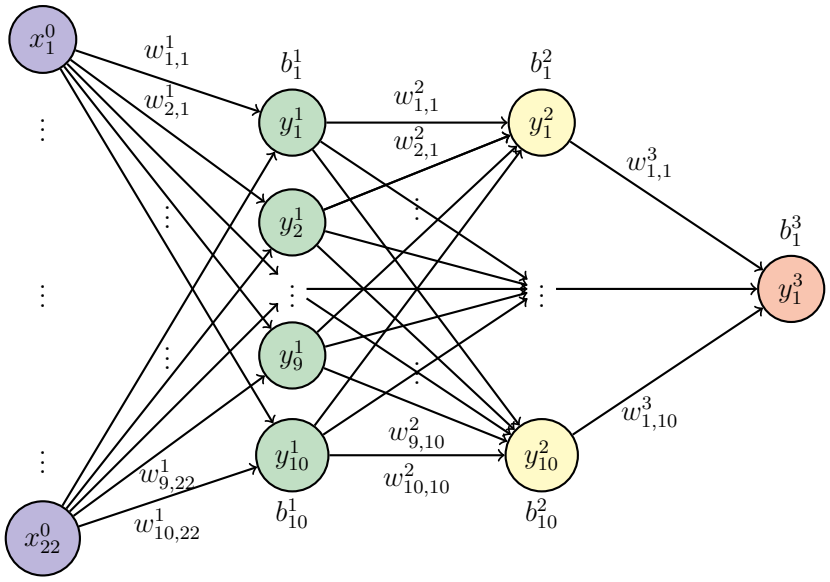

Figure 2.2: Schematic representation of a feed-forward neural network architecture. Here, with 22 inputs x_i^0, one output y_i^3, and two hidden layers of width 10. The outputs of the hidden layers are marked $y_i^{\{1,2\}}$, and the model weights $w_{i,j}^k$ are assigned to connections going into their respective neurons, with biases b_i^k written by the corresponding nodes.

inputs only from previous layers, often visualized as a weighted directed graph, illustrated in Figure 2.2. The neurons are then represented as graph nodes, with the weights being assigned to the ingoing edges. This yields a nested nonlinear function, parameterized by the weights and biases of all the neurons in the network, mapping the inputs of the first layer to the outputs of the final layer. Under some assumptions, it can be shown that feed-forward NNs are universal approximators, meaning that for any given (reasonably well-behaved) function, it can be approximated arbitrarily well by a feed-forward NN of sufficient complexity [65, 66]. As with the Hohenberg Kohn theorem introduced in the previous subsection, although this provides some theoretical grounding to the use of feed-forward NNs as ML models, it does not provide much useful information on how to actually find the NN that best approximates a given function.

In practice, feed-forward NNs are trained by optimizing a loss function $L(W, X_{\text{train}})$, with respect to the model weights W, for a given training dataset X_{train}. What loss function should be used is generally dependent on the task that the model is to solve, with the Mean Squared Error (MSE) being the typical choice for regression tasks. A simple way to perform this optimization is through gradient descent, exploiting the fact that the NN can be efficiently differentiated through backpropagation [67]. Specifically, the weights are updated as

$$W^{t+1} = W^t - \eta \nabla_W L(W^t | X_{\text{train}}), \tag{2.36}$$

with W^t the weights at training step t, and the step size η typically referred to as the Learning Rate (LR). Although the LR can be chosen before training, it is common to allow it to vary

as training progresses. Beyond empirical results showing that improved convergence can be achieved as such [68, 69], it can also be argued that a lower LR should allow for better local exploration close to local minima, while a higher LR allows for better global exploration early during training.

Although, in principle, the derivatives of the loss function should be evaluated over the entire training dataset, this is often not practically feasible. In practical training settings, datasets are often too large to fit into memory. Typically, the training dataset is instead randomly partitioned into smaller subsets, called minibatches. During training one then iterates over the minibatches, updating the weights using gradients calculated on only that batch. After each iteration over the minibatches, referred to as an epoch, the dataset is repartitioned into a new set of minibatches. This approach is commonly referred to as Stochastic Gradient Descent (SGD). In addition to allowing for training with datasets that are too large to fit into memory, the stochastic nature of SGD also allows for potentially better exploration of the loss landscape [70]. As stated above, the LR is an important hyperparameter, which must be tuned for optimal training. Likewise, SGD introduces the minibatch size as a hyperparameter.

While supervised ML methods are trained by optimizing the performance of the chosen estimator on a training dataset, the aim is to achieve a low error on unseen data. In other words, the aim is to obtain a model that can generalize. It is a well known result from statistical learning that the generalization error can be decomposed into two main types, the bias error and the variance, and that there is generally a tradeoff between the two [71]. Broadly speaking, the bias represents the inability of the model to fully fit the data, while the variance represents the sensitivity of the model to small variations in the data. For a simpler model, generally the bias error will be larger, and the variance lower. As model complexity is increased, bias decreases, but variance increases. Ideally, one wishes to strike a balance between the two. For feed-forward NNs, a simple way to avoid overfitting with an overparameterized model, is to monitor the error on a validation dataset, as training is progressing. This validation dataset should be separate from the dataset used to train the model, and provides an estimate of the generalization error. One can then end training once the error on this dataset, the validation error, starts to increase, even while the training error is still decreasing. Typically, due to the stochastic nature of training, a patience is also used, allowing the error to increase for a certain number of epochs, if it then decreases below it's smallest value. Another way to help avoid overfitting is through the use of regularization. This is typically done by applying a regularizing term to the loss function, penalizing large weight values. The most common version is L2 regularization, based on the L2 norm, added to the loss function as

$$L'(W|X_{train}) = L(W|X_{train}) + \gamma \sum_{w \in W} |w|^2, \tag{2.37}$$

as is also done in ridge regression [71]. This is closely connected to weight decay [72], where instead the weight updates are modified as

$$W^{t+1} = (1 - \alpha)W^t - \gamma \nabla_W L(W^t|X_{\text{train}}), \tag{2.38}$$

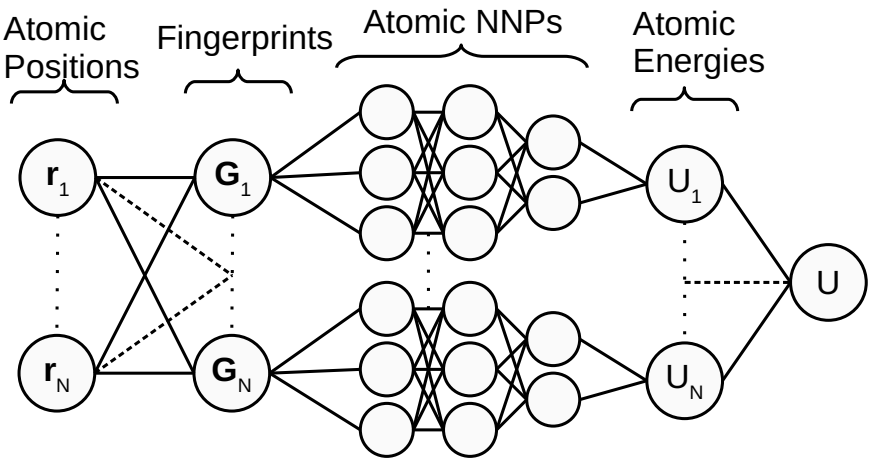

Figure 2.3: Illustration of a high-dimensional neural network potential. Atomic positions $\mathbf{r}$ are used to construct an atomic fingerprint $\mathbf{G}_i$ for each atom i, fed into a feed-forward neural network to produce atomic energies $U_i(\mathbf{G}_i)$, whose sum gives the potential U energy of the atomic system.

such that the weights are exponentially decayed, if not promoted by the second term. For SGD, these two methods are equivalent, as can be easily seen by differentiating regularization term in (2.37). However, for optimizers with momentum, and adaptive step lengths, like ADAM [73], this equivalence is no longer true. This is, in fact, a reason for the observed poor generalization errors of ADAM compared to SGD, motivating the introduction of explicit weight decay in ADAM optimization, often referred to a ADAMW [74].

2.3.2 High Dimensional Neural Network Potentials

A fundamental assumption of second generation MLIPs is that the atomic interaction is short-sighted, with each atom interacting predominantly with the atoms in its local environment, allowing the PES to be decomposed into a sum over short-ranged atomic contributions,

$$U \approx \sum_{s \in \chi} \sum_{i=1}^{N_s} U_{s,i}. \tag{2.39}$$

Here χ is the set of elements in the system, with N_s the number of atoms of species s, with $U_{s,i}$ the local energy contribution of atom i of species s. This allows for solving two of the major issues with first generation potentials. Firstly, it is clear that permutation of same-species atoms will have no effect on the total potential energy in (2.39), as long as the atomic environments are correctly accounted for. Secondly, assuming the local contributions $U_{s,i}$ remain valid in larger systems than trained on, scaling to different number of atoms

becomes possible, allowing for large scale simulations far beyond the practical limits of *ab initio*.

A variety of estimators have been used to construct $U_{s,i}$, although here we focus on feed-forward NNs, as in the original HDNNP approach [9]. A visualization of the HDNNP architecture is shown in Figure 2.3. By definition, the local atomic energy contributions $U_{s,i}$ are taken to depend only on the positions of atoms within some neighborhood centered on atom i, within some cutoff distance r_c from the central atom. Using the atomic coordinates, or interatomic displacements, directly in a ML method is, however, problematic. An immediate problem is the fact that the number of atoms in the neighborhood will vary during a typical MD trajectory, leading to a variable-dimensional feature space. It is also necessary to ensure that permutation of neighboring atoms leads to identical predictions. In most MLIP methods, this is dealt with by mapping the atomic coordinates onto a set of descriptors, called an atomic fingerprint. A variety of different descriptor types have been proposed since the rise of MLIPs, but we here focus on the ACSFs [36], introduced in the original paper on HDNNPs [9]. Beyond MLIPs, descriptors, offering a low-dimensional view of atomic local environments, or summarizing chemical or material structures, were use in other areas before the advent of MLIPs [75]. By using descriptors that are invariant under translations and rotations of the system, one also ensures the invariance of the potential under such transformations. There exist two main types of ACSFs, radial and angular. Radial Symmetry Functions (SFs) are constructed as

$$G_i^2 = \sum_j e^{-\eta(r_{ij}-r_s)^2} f_c(r_{ij}), \tag{2.40}$$

and angular SFs as

$$G_i^5 = 2^{1-\zeta} \sum_{j,k} (1 - \Lambda \cos\theta_{ijk})^\zeta e^{-\eta((r_{ij}-r_s)^2+(r_{ik}-r_s)^2+(r_{jk}-r_s)^2} f_c(r_{ij}) f_c(r_{ik}) f_c(r_{jk}). \tag{2.41}$$

Here r_{ij} is the distance between atoms i, j, θ_{ijk} the angle spanned between atoms j, k viewed from atom i, and $\eta, r_s, \zeta, \Lambda$ are all parameters. The function $f_c(r)$ is a cutoff function, that goes from $f_c(0) = 1$, to $f_c(r > r_c) = 0$, with sufficient smoothness. It should be mentioned that other forms exist than the ones described above, including the wide version of the angular SF [36], and weighted versions that try to incorporate chemical information [76] into the fingerprint. In the end, different components of the fingerprint vector will consist of G^5 and G^2 functions, with different parameter values chosen beforehand. Generally the aim is to choose the parameters such that they cover a wide range. Commonly, for the radial functions, a range of widths η is used, centered on the central atom with $r_s = 0$, or a fixed width is used, with r_s scanning over a grid of radii to probe various distances around the atom.

Training a HDNNP follows much like for other feed-forward NNs, but there are some specifics that should be mentioned. In practice, training a HDNNP is a supervised regression problem, and the loss function is typically the MSE between the predicted and *ab initio* energies, and most often also forces. Just like with the weights, the gradient of the NN with respect to its inputs can be efficiently calculated through backpropagation, allowing for calculation for the forces, as well as their gradients with the model weights, by applying

backpropagation twice. Higher order derivatives of the potential, i.e. stresses, can also be included in the loss function, in a much similar way. There is one important point, for the evaluation of the error on the energy, when configurations of many different sizes are utilized. As the energy is extensive, scaling with the number of atoms, it is common to normalize it by the number of atoms in the configuration. It should be mentioned, however, that errors typically also scale with roughly the square root of the number of atoms, shown empirically in [77]. This scaling of the error can be justified by viewing the predicted atomic energies as having some randomly distributed error around an optimal value, and then applying the central limit theorem. Although this implies that energies should be normalized by $\sqrt{N}$ in the error metrics, energy per atom is still conventionally used.

2.4 Feature Selection

One of the most crucial steps in the design of an MLIP is the choice of descriptors. There is, however, a tension between having a large set of atomic features, thus providing more information on atomic environments, and the increased computational cost. Beyond a certain point, adding more features provides diminishing returns, while also contributing to overfitting, and negatively affecting generalization [76]. Manually designing a set of features can be time-consuming, and requires a lot of experimentation. In doing so, it can be difficult to ensure that features are optimal, especially for nonorthogonal basis sets such as the BP SFs. Although more modern deep-learning-based models, such as *Deep Potentials* and Graph NN potentials, do not require a predefined set of descriptors, they can be considered as learning a set of features as part of training. The number of such learned features is then, still, a parameter that must be tuned, subject to the same dichotomy between computational efficiency, and accuracy.

An alternative approach to manually designing a feature set can be found in feature selection. In principle, the task of finding the optimal subset out of a set of features is a combinatorial optimization problem, prohibiting a direct search in any realistic application. A variety of feature selection methods have, however, been developed, in order to make this problem tractable [78]. Some specific methods, of relevance for this thesis, will be detailed in the following subsections. Typically these methods can be divided into three types.

- *Filter Methods:* Selecting features by optimizing some criteria over the dataset, before training takes place.
- *Wrapper Methods:* Using a search method to traverse the space of possible feature subsets, evaluating each subset by training a model on it.
- *Embedded Methods:* Selecting features during the training process, for instance through adding a regularization term to the loss function.

Early use of embedded feature selection to MLIPs was made by Seko *et al* [12], fitting linear models via the LASSO method [13]. In this approach, an L1 regularization term is added to the loss function, penalizing the linear weights of the model. As training is

performed, dependent on the regularization strength, the weights corresponding to redundant features will go to zero, and can be subsequently discarded.

For nonlinear MLIPs, much of the work on feature selection has been focused on filter methods. A central work in this direction was due to Imbalzano et al [14], introducing the CUR method. This approach relies on the CUR decomposition [79], wherein the data matrix is given a lower-dimensional representation, similar to the well known singular value decomposition. In contrast to singular value decomposition, the low dimensional matrix in CUR decomposition consists of only rows and columns of the original matrix, allowing for selecting either features or datapoints. The criterion, in this case, is the reconstruction error, which should be minimized. The Pearson Correlation (PC), and Furthest Point Sampling (FPS) methods, based on minimizing the Pearson correlation, and maximizing Euclidean distances, respectively between features, were also introduced in that same article.

The CUR, FPS, and PC methods are unsupervised, in the sense that they do not take into account the training targets, only the input data. While they can be used to reduce redundancy between features, they cannot identify features that, although useful for distinguishing atomic environments, might be less important for predictions. To remedy this, the CUR and FPS methods were extended with a supervised component through Principal Covariates Regression [80]. One of the contributions of this thesis, presented as a paper in Section 5, has been to develop an embedded approach to feature selection for nonlinear MLIPs, namely the Behler-Parrinello HDNNP. This method is based on the Adaptive Group Lasso (AGL) [81], applied to NNs, and will be covered in more detail below.

2.4.1 Unsupervised Filter Methods

One peculiarity of the MLIP setting is the fact that each configuration contains multiple realizations of the atomic features, one for each atom in the system. Further, in multi-component systems, each species has its own set of features, realized once for each atom of that species, in a given configuration. For supervised filter methods, this is a potential problem. This is not the case for unsupervised methods, which have therefore seen use for MLIPs.

A very simple approach to unsupervised feature selection is the PC approach of Imbalzano *et al* [14], which serves as a useful reference to evaluate more sophisticated methods. The Pearson correlation between two random variables is given by the ratio between their covariance and the product of their standard deviation. In this case

$$\rho_{ij} = \sum_{k \in \{\text{atoms}\}} \frac{(G_{i,k} - \overline{G}_i)(G_{j,k} - \overline{G}_j)}{\sigma_i \sigma_j}, \tag{2.42}$$

with k indexing all atoms of a given species, across all considered configurations, $G_{i,j}$ being feature i of atom k, with $\overline{G}_i$ and σ_i being its mean and standard deviation across all atoms and configurations. To select features with the PC method, one selects an initial feature out of a predefined set of candidates. Additional features are then added to the set via an iterative process. Each unselected candidate feature is scored by calculating the average

Pearson coefficient with respect to all the previously selected features. The feature with the lowest score is then added to the selected features, and the process is repeated until a sufficient number of features have been chosen.

The CUR method, proposed by Imbalzano *et al* [14], relies on the CUR decomposition [79]. Let X be the matrix containing as rows the features of each atom, of a given species, in each considered configuration. The CUR decomposition transforms this matrix into a smaller-dimensional space

$$X \approx CUR. \tag{2.43}$$

Although similar to other matrix decompositions such as the Singular Value Decomposition (SVD), here C and R are constrained to be composed of rows and columns of the original matrix. These rows and columns are to be chosen, such that the reconstruction error

$$\epsilon = \frac{\|X - CUR\|_F}{\|X\|_F}, \tag{2.44}$$

with $\|X\|_F = \sqrt{\mathrm{Tr}(XX^\dagger)}$ denoting the Frobenius norm, and $X^\dagger$ the hermitian conjugate of X, is minimized. In practice each feature i, corresponding to the column i of the data matrix X, is given a score

$$\pi_i = \sum_{j=1}^{k} (\nu_i^j)^2, \tag{2.45}$$

with k the number of features that are not yet selected, and ν_i^j component i of the jth singular vector of X. It is also possible to use a fix $k = 1$, allowing for a faster computation. To select features with this method, the score of each feature is calculated, then the one with the highest importance is selected. Next, to avoid selecting nearly identical features, the remaining columns are orthogonalized with respect to the selected one. The weights are then re-evaluated, and the process continued until a desired number of features are selected.

2.4.2 Supervised Embedded Methods

The LASSO [13] is a technique for training sparse linear models. For a linear regressor

$$\hat{y}(X|\beta) = \beta_0 + \sum_{j=1}^{M} \beta_j X_j, \tag{2.46}$$

with β the weights of the model, X the input features, the LASSO estimate is given by the ordinary least squares

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{N} \left(y_i - \hat{y}(X^i|\beta)\right)^2, \tag{2.47}$$

subject to the constraint

$$\sum_{j=1}^{M} |\beta_j| \leq t. \tag{2.48}$$

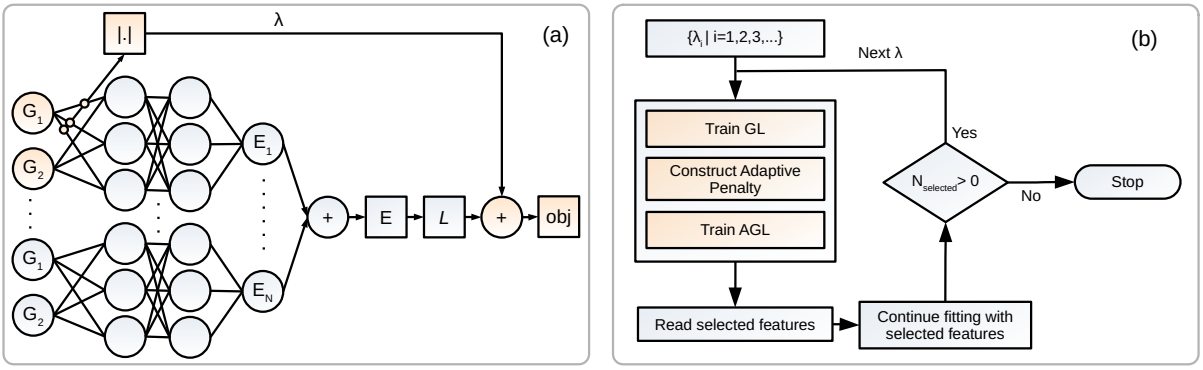

Figure 2.4: Schematic of feature selection for HDNNPs with AGL. (a) Illustration of a NNP, with GL penalty added to the first feature, used to construct the objective function. (b) Flowchart of the feature selection process.

It is useful to rewrite this in an equivalent form [71], using a Lagrange multiplier

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} \left(y_i - \hat{y}(X^i|\beta) \right)^2 + \lambda \sum_{j=1}^{M} |\beta_j| \right\}. \tag{2.49}$$

For the unconstrained least squares, obtained by setting $\lambda = 0$ in eq. (2.49), it is possible to obtain a closed-form solution. This is likewise the case for ridge regression, obtained by replacing the absolute value in the last term of eq. (2.49) by the square. For the LASSO, this is however not the case, and so iterative methods like gradient descent must be employed. A problem, however, arises from the non-smooth nature of the regularization term for $\beta_i = 0$.

One way to solve this is by use of a proximal algorithm. Proximal operators [82] are a powerful tool for nonsmooth and constrained optimization problems. Given a function $f(x)$, the proximal operator of f is defined by

$$\text{prox}_{\lambda f}(x) = \arg\min_{z} \left\{ f(z) + \frac{1}{2\gamma}|z - x|^2 \right\}, \tag{2.50}$$

with $\gamma > 0$. This operator can be viewed as a generalized kind of projection operator, and in particular it can be shown that x^* is a fixed point

$$x^* = \text{prox}_{\lambda f}(x^*) \tag{2.51}$$

if and only if x^* minimizes f.

While the LASSO penalty can be applied to NNs as a regularization, although much less common than L2 regularization, there is a significant difference to the linear case that makes it unsuitable for feature selection in this setting. The L1 penalty pushes *individual* weights to vanish, which allows for feature selection in the linear case where each feature has an associated weight. For feed-forward NNs, each input, and each internal node, is associated to several weights feeding into deeper layers of the network. Introducing sparsity on the level of

individual weights does not allow for deselecting features, nor for pruning internal layers of the network, but merely reducing the number of parameters in the model.

As the issue lies in each weight being penalized individually, a way forward is offered by the Group Lasso (GL) [83]. In many applications it is useful to have subsets of features be selected or discarded collectively, as a group. It is for just these cases that GL was originally developed. Let the parameters be organized in a set of N_{groups} vectors $\boldsymbol{\beta}_j$, with $1 \leq j \leq N_{\text{groups}}$ indexing the groups. We can then obtain the GL by replacing the constraint in eq. (2.49) with

$$\sum_{j=1}^{N_{\text{groups}}} ||\boldsymbol{\beta}_j|| \leq t, \tag{2.52}$$

where $||\mathbf{v}|| = \sqrt{\sum_{i=0}^{\dim(\mathbf{v})} v_i^2}$ denotes the standard Euclidean norm. As such, the GL estimator is given by

$$\hat{\beta} = \arg\min_{\beta} \left\{ \sum_{i=1}^{N} \left(y_i - \hat{y}(X^i|\beta)\right)^2 + \lambda \sum_{j=1}^{N_{\text{groups}}} \|\beta_j\| \right\}. \tag{2.53}$$

In the case of NNs, a natural grouping of the weights is provided by the network architecture, namely parameters corresponding to a given feature, or to a given node [84]. The proximal operator for the Euclidean norm, to be applied once for each parameter group $\boldsymbol{\beta}_j$, is [85, 84]

$$\text{prox}_{\lambda\|\cdot\|}(\boldsymbol{\beta}_\text{j}) = \left(1 - \frac{\lambda}{\|\boldsymbol{\beta}_j\|}\right)_+ \boldsymbol{\beta}_j. \tag{2.54}$$

There are some issues with the GL. For NNs, the GL is unable to consistently deselect irrelevant features [86]. The LASSO, likewise, is in some circumstances inconsistent, justifying an adaptive approach [87]. While in the original LASSO and GL, a single regularization parameter is used for each parameter, the adaptive LASSO [87] and AGL [81] allow each feature to be penalized by a different parameter. To estimate the feature-dependent regularization strength λ_j, for NNs, the GL+AGL approach of Dinh and Ho [86] proposes to first train an estimator using the non-adaptive GL. Letting the input weights corresponding to feature j of a NN be denoted $\boldsymbol{\omega}_j^0$, the GL estimator is obtained by optimizing

$$\text{obj}(W) = L(W) + \frac{\lambda}{M} \sum_{j=1}^{M} ||\boldsymbol{\omega}_j^0||, \tag{2.55}$$

where W denotes the weights of the NN, L is a loss function, and M is the number of features. The way this penalty is applied in the HDNNP setting is illustrated in Figure 2.4(a). Letting $\hat{\boldsymbol{\omega}}_j^0$ the input weights obtained in this first training run, the adaptive penalty is obtained through scaling the penalty for each feature by the Euclidean norm of these,

$$\text{obj}(W) = L(W) + \frac{\lambda}{M} \sum_{j=1}^{M} \frac{||\boldsymbol{\omega}_j^0||}{||\hat{\boldsymbol{\omega}}_j^0||}. \tag{2.56}$$

This way of choosing the adapted weights is not unique, and in principle other approaches can be considered. Unlike for linear models, the nonlinear nature of NNs make theoretical results harder to obtain. The GL+AGL is, however, shown to be feature selection consistent [86], in the sense that for any $\delta > 0$, there is a smallest training dataset size N_δ s.t. all irrelevant features are discarded, and all relevant features are selected, with probability $1 - \delta$. In practice, when we refer to AGL for NNs, we will implicitly refer to the GL+AGL approach described above. The overall procedure for selecting features with this method involves performing a hyperparameter search over different values of λ, with the overall process shown in Figure 2.4(b). After training a model with a given penalty, the penalty is removed, and training continued with the selected features only. This continued training is performed in order to obtain an unbiased estimate of the model performance, as the L1 penalty is known to shrink also relevant weights, and thus slightly degrade model performance [88].

LassoNet [89] is another embedded feature selection method for NNs, inspired by the LASSO. Here a specific NN architecture is used, namely the residual NN. Being often used to avoid the vanishing gradient problem in deep NNs [90], the residual architecture adds bypass connections between layers of a feed-forward NN. In LassoNet, these connections are between the input and output layers of the network, and the residual network can be considered as an estimator with the NN constituting a non-linear component, and the bypass forming a linear component. With $y(W, \mathbf{X})$ being a feed-forward NN, parameterized by weights W, and inputs $\mathbf{x}$, the residual NN considered in LassoNet is given by

$$y(\boldsymbol{\theta}, W, \mathbf{x}) = y(W, \mathbf{x}) + \sum_j \theta_j x_j, \tag{2.57}$$

with $\boldsymbol{\theta}$ being the bypass weights. The model is then trained as in LASSO, with L1 regularization applied to the bypass weights only. Specifically, the weights are given by

$$(\boldsymbol{\theta}, W) = \arg\min_{\boldsymbol{\theta}, W} \left\{ L_(\boldsymbol{\theta}, W) + \lambda \sum_{j=1}^{M} |\theta_j| \right\}, \tag{2.58}$$

with L again the loss function evaluated over the training dataset, and M the number of features. This minimization is, however, subject to the constraint

$$||\boldsymbol{\omega}_j^{\mathbf{0}}||_\infty \leq \mu |\theta_j|, \; j = 1, \ldots, M, \tag{2.59}$$

with $||\boldsymbol{\omega}_j^{\mathbf{0}}||_\infty$ denoting the taxicab norm of $\boldsymbol{\omega}_j^{\mathbf{0}}$, i.e. the largest absolute value of any component of $\boldsymbol{\omega}_j^{\mathbf{0}}$, and μ being a hyperparameter used to control the relative importance of the linear and nonlinear components of the model.

Although LassoNet was originally planned to be used in this thesis, we ultimately opted for AGL instead. The ultimate reasoning for this is the fact that LassoNet requires changes to the standard feed-forward architecture commonly used for NNPs. While there is no reason to believe the residual feed-forward architecture would be unsuitable for NNPs; most interfaces to simulation frameworks [91] use the more traditional network architecture. Thus, the use of LassoNet would pose a potential obstacle to simulations. The AGL has also been found to

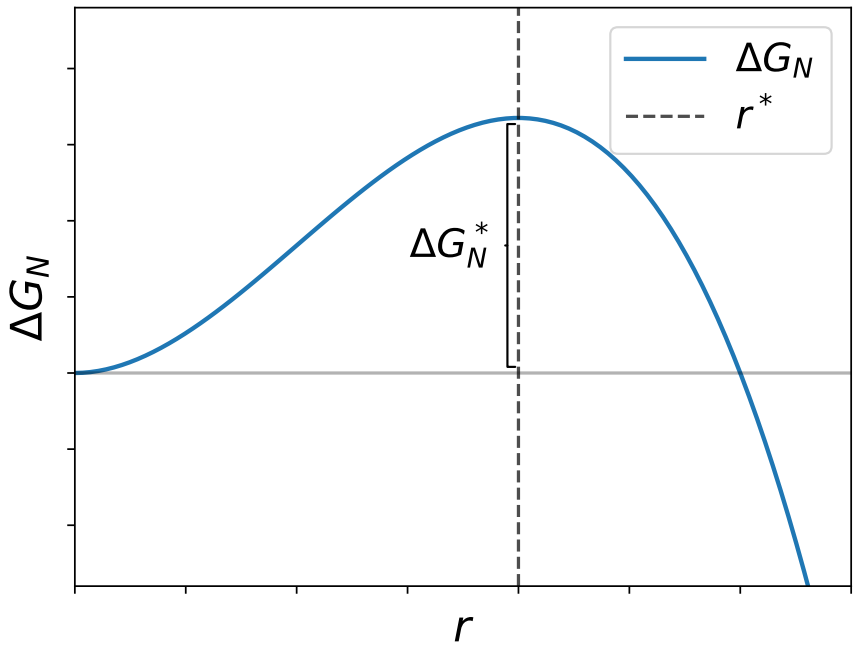

Figure 2.5: Schematic visualization of the free-energy of formation, in CNT, as function of nucleus size, in arbitrary units. The critical radius r^* is marked by a dashed line, with the free-energy barrier ΔG_N^* marked.

be competitive with LassoNet [86]. It could, however, be possible to use the LassoNet just to select features, with a more traditional architecture used to train a second model on the selected features.

2.5 Nucleation

Nucleation is the process whereby a phase transition is initiated through the formation of microscopic clusters of atoms in a new phase, called a nucleus, inside the thermodynamically unstable mother phase. Although nucleation occurs within many different types of phase transitions, we will in this work only consider the liquid-solid transition in which a crystal emerges from an undercooled melt [3]. In the presence of impurities, as is often the case in the real world, these often serve to catalyse the emergence of nuclei. This setting is referred to as heterogeneous nucleation. In contrast, nucleation in absence of impurities is termed homogeneous.

The Classical Nucleation Theory (CNT), dating back almost a century [92, 93, 94], remains to this day one of the main ways of conceptualizing the nucleation process [95]. In CNT, a core assumption is made, that the nucleus, once formed through a thermodynamic fluctuation, can be treated as a macroscopic object. The nucleus is thus regarded as a bulk volume of crystal, with a phase interface separating it from the mother phase. Assuming a free energy

difference between the two phases ΔG_V, and a free energy ΔG_S required to construct the interface, the the formation free-energy of the nucleus can be written out

$$\Delta G_N = \Delta G_V + \Delta G_S = \frac{4\pi r^3}{3}\Delta g_V + 4\pi r^2 \gamma_S. \tag{2.60}$$

Here, the second equality assumes a spherical nucleus, with radius r, a free energy difference per volume $\Delta g_V = \Delta G_V / V$, and a surface free energy γ_S. Note that here $\Delta g_V = g_{\text{solid}} - g_{\text{liquid}} < 0$. The competition between these two terms lead to a free energy barrier, with a maximum corresponding to the critical radius

$$r^* = -\frac{2\gamma_S}{\Delta g_V}, \tag{2.61}$$

and, assuming a number density ρ in the solid phase, a critical nucleus size

$$n^* = -\frac{32\pi\rho}{3}\frac{\gamma_S}{\Delta g_V}. \tag{2.62}$$

The size-dependence of the free-energy of formation is shown, schematically, in Figure 2.5. If a nucleus appears from the melt, with fewer atoms than the n^*, the surface term dominates, and the nucleus dissolves back into the melt. Only if the nucleus is large enough that the bulk term dominates does it grow, and initiate crystallization of the system. The height of the free energy barrier can further be written down as

$$\Delta G_N^* = \frac{16\pi}{3}\frac{\gamma_S^3}{\Delta g_V^2}. \tag{2.63}$$

With some further assumptions, including the assumption that the nucleus size obeys a Markov process, and that the growth of a supercritical nucleus occurs on a much greater timescale than that required for the nucleus to appear spontaneously, the spontaneous nucleation rate is given by [3]

$$\mathcal{I} \propto \exp\left(-\frac{\Delta G_N^*}{k_B T}\right), \tag{2.64}$$

being the probability, per unit time and volume, of a critical nucleus forming.

A common deviation from the simple picture given by CNT is in multistep nucleation. While in CNT there is a direct transition to from the liquid, to the thermodynamically stable crystal state, one can also imagine nucleation pathways wherein a precursor nucleus of a different crystal phase appears. Such a two-step process is known to occur in Lennard Jones liquids [96, 97], and in some metallic systems using classical potentials [30, 53, 98]. There are theoretical arguments, based on the Landau theory [99], for why a BCC crystal structure would be favored during early stages of nucleation, even if the stable crystal is of a different structure.

Chapter 3

Homogeneous Nucleation in Pure Aluminium

The following chapter is based on a paper published as part of this doctoral work [100]. The aim of the paper was the development, and exploitation, of a HDNNP for pure Al, specifically for the study of homogeneous nucleation. As this was a collaborative work, with many authors, some comments are warranted regarding my personal contributions. Specifically, the HDNNP model was trained before the start of my doctoral work, by A. Saliou and N. Jakse, based on *ab initio* data performed by the latter [101, 102, 103, 104]. With the potential in hand, my contribution was to perform the molecular dynamics simulations, and the majority of the analysis thereof. The exceptions to this are the simulations with the ANI-Al potential [105], performed by L. Granz, and the calculation of the intermediate scattering function, and dynamic structure factor, which were performed by N. Jakse, using trajectories obtained from my simulations. I made major contribution to the writing of the manuscript, primarily the results, with additional contributions to other parts of the text. The text presented here has been minimally edited, in order to unify the references with the rest of this thesis.

Machine Learning Interatomic Potentials for Aluminium: Application to Solidification Phenomena

Noel Jakse[1], Johannes Sandberg[123], Leon F Granz[23], Anthony Saliou[1], Philippe Jarry[4], Emilie Devijver[5], Thomas Voigtmann[23], Jürgen Horbach[6], Andreas Meyer[27]

[1]Université Grenoble Alpes, CNRS, Grenoble INP, SIMaP, F-38000 Grenoble, France
[2]Institut für Materialphysik im Weltraum, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 51170
[3]Department of Physics, Heinrich-Heine-Universität Düsseldorf, Universitätsstraße 1, 40225 Düsseldorf, Germany
[4]C-TEC, Parc Economique Centr'alp, 725 rue Aristide Bergès, CS10027, Voreppe 38341 CEDEX, France
[5]Université Grenoble Alpes, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France
[6]Institut für Theoretische Physik II, Heinrich-Heine-Universität Düsseldorf, Universitätsstraße 1, 40225 Düsseldorf, Germany
[7]Institut Laue-Langevin (ILL), 38042 Grenoble, France
DOI: 10.1088/1361-648X/ac9d7d

Abstract

In studying solidification process by simulations on the atomic scale, the modeling of crystal nucleation or amorphisation requires the construction of interatomic interactions that are able to reproduce the properties of both the solid and the liquid states. Taking into account rare nucleation events or structural relaxation under deep undercooling conditions requires much larger length scales and longer time scales than those achievable by *ab initio* molecular dynamics (AIMD). This problem is addressed by means of classical MD simulations using a well established high dimensional neural network potential trained on a set of configurations generated by AIMD relevant for solidification phenomena. Our dataset contains various crystalline structures and liquid states at different pressures, including their time fluctuations in a wide range of temperatures. Applied to elemental aluminium, the resulting potential is shown to be efficient to reproduce the basic structural, dynamics and thermodynamic quantities in the liquid and undercooled states. Early stages of crystallization are further investigated on a much larger scale with one million atoms, allowing us to unravel features of the homogeneous nucleation mechanisms in the fcc phase at ambient pressure as well as in the bcc phase at high pressure with unprecedented accuracy close to the *ab initio* one. In both case, a single step nucleation process is observed.

3.1 Introduction

Apart from steels, aluminium and its alloys represent the most used and attractive structural

metallic materials due to their specific properties such as low weight, low energy cost of remelting, and the possibility of almost complete recycling. Therefore, these materials represent a major axis of the energy transition [106]. An intimate understanding of its condensed phase properties is founded upon a description of the atomic level structure and dynamics, and requires an accurate representation of chemical bonding [4]. This is of utmost importance in order to tackle phenomena such as phase changes and solidification process during which a liquid morphs into a solid either by crystallization or amorphisation [2, 107], showing eventually a change in electronic structure as a metal-to-semiconductor transition [108, 109]. First-principles approaches, essentially through the Density Functional Theory (DFT) [110, 111], represent the dedicated framework especially with the breakthrough provided by *ab initio* molecular dynamics (AIMD) simulations [23] in combining atomic dynamics with DFT. Despite its enormous success in many complex chemical bonding environments [4], DFT implementations are limited to a few hundred atoms over time scales less than 1 ns [101, 112] on current large-scale supercomputing facilities, impeding its use for phenomena at length and time scales typical of solidification [3].

The desire to bridge typical scales of the electronic structure to those of the properties under investigation has led to deriving interatomic potentials with semi-empirical functional forms that average out or otherwise model electronic degrees of freedom. For metallic materials of interest here, various approaches were proposed, starting in the second half of the 20th Century with pair-potentials based on a nearly-free electron gas description of the electronic structure [113] using simple models within the pseudopotential theory (PT) [114, 115, 116, 117]. It was later acknowledged that it was impossible for pair potentials to describe on the same footing the structure, dynamic, and thermodynamic properties in the liquid and solid state [29], with inherent mechanical instability under shear for crystals. Many-body approaches such as the Embedded-Atom Model (EAM) [118, 119], modified EAM (MEAM) [120] of current widespread use as well as the Reactive Force Field (ReaxFF) [121], just to name a few among many others [122], can be considered as successful in this respect. Fitting the parameters of these potentials is most often oriented towards describing crystalline phases and transitions between some of them [123, 124], more rarely taking a full account of the liquid state [31]. This leads to a lack of transferability [29] and a limited ability to tackle phenomena involving several phases such as crystal nucleation [3].

Over the last decade, impressive progress was made in designing potentials from electronic structure calculations using supervised Machine Learning (ML) methods [125, 126, 127, 128, 129, 130, 131]. There are now standard libraries for the ML training [132] that can be used in combination with molecular dynamics (MD) simulation packages [91] such as LAMMPS [133] or in combination with workflow environments such as ASE [134]. On-the-fly ML force field methods have been also proposed [135] and implemented directly into *ab initio* codes in order to bypass most of electronic-structure calculation steps [136]. Different ML techniques have been used, ranging from simple linear regression (LR) methods such as the spectral neighbor analysis potential (SNAP) method [35, 130] to highly non-linear regression methods using High Dimensional Neural Networks [9, 132] (HDNN) or Kernel Regression (KR) [10, 137, 138]. The designed potentials reach in general an accuracy close to the *ab initio* calculations from

which the database was formed, with exceptional results for the description of relative stability between crystalline phases [139] and defects [124]. However, approaches taking full account of liquid and crystalline states remain scarce [140, 139, 109, 141] and are often limited to the objective of a good description of the melting point. The main reason for this stems from the fact that the chosen *ab initio* configurations should cover all situations, as ML techniques may become less reliable outside the training domain [125]. It becomes even more crucial for crystal nucleation occurring under deep undercooling conditions with a strong evolution of the liquid structure with respect to that above melting, showing an increasing icosahedral [101] ordering and structural heterogeneity [142] triggering homogeneous nucleation [143].

Machine-learning potentials for aluminium were designed very recently to describe essentially the properties of the solid states [144, 145, 105] and the melting temperature [144, 105]. Different approaches were put forward respectively with a Gaussian kernel regression [144] and a deep NN [145] with a dataset built from configurations extracted from AIMD simulations at various temperatures, and an active-learning approach with periodical retraining of the NN [105]. ML potentials were initially trained using the DFT energies starting from the work of Behler and Parrinello on bulk Si [9]. It was subsequently pointed out that the learning process could benefit from a wealth of additional information if the three components of the force and six components of the stress per atom are taken into account [125, 129], while one has only a single energy value per simulated configuration. Still in some works, only the forces have been used for the training showing that properties like the vibrational properties in the solid states can be reproduced, but they remain insufficient to get full account of thermodynamic quantities [138, 144]. Thus whether additional information enhances the training or not can still be questioned, also given the fact that the relative importance of the energy, forces and stresses for estimating the Mean-Square Error (MSE) or the Root-MSE (RMSE) introduce two additional free training parameters [125]. Moreover, the question of the transferability of a ML potential taking into account both the liquid and solid phases as mentioned above, remains essentially unexplored for aluminium. This aspect is also of importance when dealing with solidification phenomena.

The aim of the present work is to develop a ML potential for pure aluminium dedicated to the description of condensed phases, namely liquid and solid states for temperatures up to 8000 K and pressures up to 300 GPa and therefore able to deal with solidification phenomena. For this purpose, a HDNN was developed on the basis of well-known and robust Behler and Parrinello's approach [9, 125]. The latter was trained on a data set generated by DFT-based simulations for the main crystalline structures and liquid states covering the targeted pressure and temperature domain, including their time fluctuations by an appropriate sampling of phase space trajectories [146]. It is shown that training the HDNN on sampled AIMD trajectories using solely energy labels leads to an accurate description of the structure, dynamics and thermodynamics in the investigated domain. More specifically, the single-particle as well as the collective dynamics are well reproduced, which is an essential ingredient in describing solidification. The resulting potential is then applied here to solidification processes, namely amorphization and early stages of crystal nucleation, allowing us to unravel features of the homogeneous nucleation mechanisms at ambient as

well as high pressure.

The remaining part of the paper is organized as the following. In Sec. 3.2, the specific features of the HDNN, the training procedure as well as the basic assessment of the potential on independent DFT and experimental thermodynamic data are outlined. Sec. 3.3 is devoted to the test of the potential's accuracy in describing some structural properties, the dynamics of the liquid state in the investigated pressure-temperature domain, as well as the homogeneous nucleation. Finally, in Sec. 3.4, the main outcomes of the work are given.

3.2 Computational background

3.2.1 Constructing a machine learning potential

In the last three decades, many potentials of pure Al and its alloys for the use in atomistic simulations been developed [147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160], using the Morse potential ansatz [147], the PT method [155], embedded-atom method (EAM) [149, 150, 151, 154, 156, 158], the modified EAM (MEAM) [120, 153, 160], and many-body approaches such as COMB3 [159]. However, only very few studies have employed the ML approach [138, 144, 145], none of them taking systematically into account the properties of the liquid state and checking the dynamics which is very important for the solidification aspects. This is precisely one of the aims in building a ML-based potential here.

Among the various approaches put forward to design ML potentials [126, 130], the choice was made to set up a high-dimensional neural network built in a similar way to the one proposed by Behler and Parrinello [9] and Zhang *et al.* [39]. This well established approach has been proven successful for pure silicon [109] as well as water [139] to model reliably both their properties in the liquid and solid states. As a detailed description can be found for instance in the tutorial review by Behler [125] among others, the focus is made mainly on the specifics of our scheme. The main part consists in a supervised learning task from a relevant sample of atomic configurations with known energy generated by AIMD in various crystalline and liquids structures in the desired temperature and pressure domain. In the corresponding portion of Potential Energy Landscape (PEL), each configurational energy is considered as a sum of individual atomic energies determined from their local atomic environment within a cut-off radius r_S often extending beyond the first-neighbor atomic shell, and taken here to be 6.4 Å for Al, corresponding to at least the second neighbor shell for all the considered thermodynamic states. This decomposition allows us to train N Neural Networks (NN), each of them being assigned to an individual local atomic environment. The NN are then combined to recover the energy of the whole configuration of N atoms.

The individual NN is defined by the same network topology for given atomic species. It specifies the number of neurons formally named here y_i^l and their connectivity through the weights, $w_{i,j}^l$, and called a Multi-Layer Perceptron (MLP). The weights associated with each node pair are optimized during the learning process by a back-propagation technique [161]. Thus, each of the M layers within the neural network consists of sets of nodes receiving multiple inputs from the previous layer and passing outputs to the next layer. Here a fully

connected network is used, in which every output of a layer is an input for every neuron in the next layer. The corresponding mathematical description is as follows: the input signals are linearly combined before being activated by function f to give each output y_i^l of a given fully connected layer l as

$$y_i^l = f\left(\sum_{j=1}^{M_{l-1}} w_{i,j}^l y_j^{l-1} + b_i^l\right), \tag{3.1}$$

where M_l refers to the size of the l-th layer, *i.e.* the number of its neurons. Note that positive weights enhance connections while negative weights tend to inhibit them. Most of the activation functions are chosen to have a range in either $[0, 1]$ or $[-1, 1]$ and modulate the amplitude of the output. The activation function f is applied element-wise and is taken as the softplus function $f(x) = \log(1 + e^x)$. Back-propagation is used to update the network weights and their gradients.

The input layer of a NN takes values representative of one local atomic environment in the form of a feature vector whose dimension is then equal to the number of its nodes. The feature vector is built on the basis of Behler-Parrinello (BP) descriptors [9] to represent the radial and angular arrangements of atoms in the local structure using Gaussian symmetry functions having the translational and rotational invariance. For aluminium, the number of components of the BP feature was chosen to be 22, comprising of 12 radial and 10 angular components, as described in more detail in the Supplementary Information (SI) file. Then, the NN architecture for aluminum is $22 \times 10 \times 10 \times 1$ with 2 hidden layers of 10 nodes each.

The NN was coded using KERAS module from the TENSORFLOW PYTHON package [162] in the regression mode. NNs of all N atoms of a configuration are then associated using the ADD module to form the HDNN which is obviously invariant to permutation of atoms. The HDNN is then trained on the single DFT energy of the whole configuration.

3.2.2 Building the dataset

Designing an appropriate dataset is the crucial and demanding step for the construction of the NN interatomic potential. Various strategies can be put forward to construct it, which were reviewed very recently [163]. Here, it was built from AIMD simulations that were partly taken from our previous works [101, 102, 103, 104] and extended here to have a better representation of the undercooled liquid region, the crystalline fcc configurations up to the melting point at zero pressure, and crystalline fcc, bcc, and hcp up to 300 GPa with and without defects. The different thermodynamic states and structures as well as the number of configurations sampled in each case are given in Tables SI and SII in the Supplementary Information file. In practice, for each temperature and pressure a number of 100 to 1000 configurations are sampled from the corresponding AIMD trajectory. In total, 24300 configurations of $N = 256$ atoms were gathered in the database that enabled us to cover solid and liquid states at ambient pressure as well as liquid samples at temperatures up to 8000 K and pressures up to 300 GPa. Non-equilibrium trajectories in the undercooled region were also generated to take into account crystal nucleation and solidification processes in the ML fitting procedure.

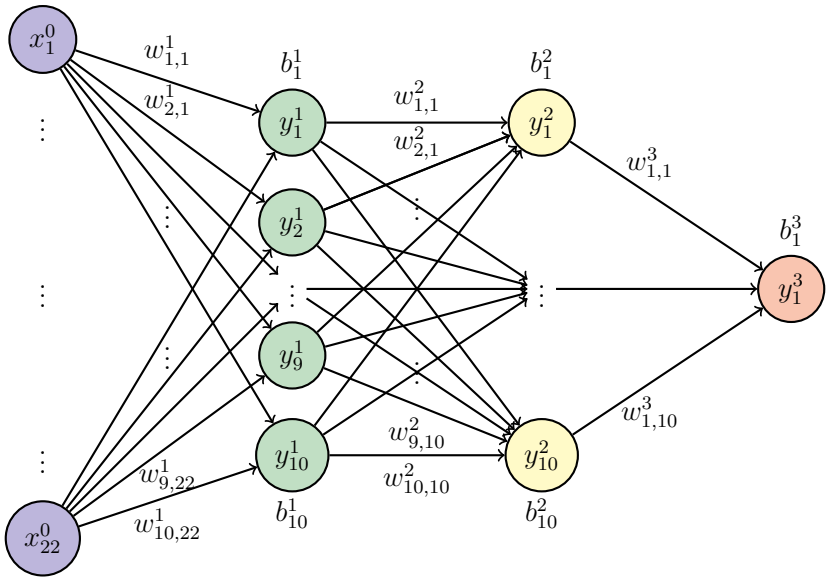

Figure 3.1: Schematic representation of the feed-forward neural network built as a densely connected multi-layer perceptron. The input layer x_i^0 will be fed with the atomic BP feature of the training set, and the output is the atomic energy. The neural network contains two hidden layers with superscript 1 and 2, respectively. The two layers are composed of 10 neurons each. The weights w_{ij}^k and bias b_i^k are optimized during the training (see text).

For the sake of self-consistency, the main technical details of the AIMD simulations are recalled here. They were performed by means of the Vienna Ab initio Simulation Package (VASP) [164]. The local density approximation (LDA) [165, 166] within projected augmented plane-waves [167] was applied to all simulations with a plane-wave cutoff of 241 eV. For the liquid states, only the Γ-point is used while for crystalline states, the Γ-centered grid of k-points in the irreducible part of the Brillouin zone was set to $2 \times 2 \times 2$ following the Monkhorst–Pack scheme [168, 169]. All the simulations were performed with $N = 256$ atoms placed in a cubic simulation box (except for the hcp crystal where an orthorhombic box was used) with standard periodic boundary conditions (PBC). Newton's equations of motion were solved numerically with Verlet's algorithm in the velocity form with a time step of 1.5 fs, and phase-space trajectories were constructed within the canonical ensemble (NVT), by means of a Nosé thermostat to control the temperature T. The temperature evolution in the undercooled states was obtained by quenching the system stepwise down to 600 K with a temperature step of 50 K. For each temperature, the simulation cell was resized according to the experimental density [170] and the run was continued for 30 ps before performing the next quench, resulting in an average cooling rate of 3.3×10^{12} K/s. The calculated pressures for all the temperatures studied here were in the range ±1 GPa generally, so that on average a quasi constant pressure during the quenching is observed. For temperatures ranging from $T = 1000$ K to 600 K, the run was continued for equilibration during a time up to 200 ps. A similar procedure was applied for heating the fcc crystal from 10 K to 900 K.

Several aspects deserve attention in the perspective of building the ML potential. First of

all, the choice of the exchange and correlation (XC) functional for the electronic structure calculations is crucial. As the ML potential may reach an accuracy similar to the DFT calculations, it will mirror the ability of the XC functional used in predicting the properties, at least in the thermodynamic domain inside which it was trained. This has guided our choice of the Local Density Approximation (LDA) functional given the fact that the Generalized-Gradient-Approximation (GGA) overestimates the atomic volume [171]. Moreover, it was shown in our previous contributions that the LDA gives a good description of the liquid structure [101]. More importantly, atomic transport properties such as the self-diffusion coefficient, which are very sensitive to the details of the potentials, are well reproduced within the LDA compared to state-of-the-art experimental data [172, 173]. Such a good agreement with experiments was very recently confirmed on the dynamic structure factors as well as the structural relaxation times extracted from the intermediate scattering function [104]. For high pressures, it was shown [174] that the difference between LDA and GGA [175] is negligible in describing the pressure-density phase diagram of aluminium up to pressures as high as 10 TPa.

Secondly, in the perspective of performing MD simulations, care has to be taken in describing not only average thermodynamic properties but also the fluctuations around the mean value, especially in order to capture the features of local basins of the PEL [146]. This requires the sampling of a large number of configurations along AIMD phase space trajectories. Therefore, in the present work, for each of the considered thermodynamic states (see Tables SII and SIII of the SI file), 1000 configurations were generated on AIMD production runs over 40 ps.

Finally, as mentioned in the introduction, the question whether including the additional information of the forces or even the stresses in addition to the energies improves the learning process and the accuracy of the potential deserves further attention. It was shown very recently for molecular systems that forces and energies contribute equally to the convergence of the prediction errors [176]. The choice of considering energies, forces or both of them in the training may depend on factors such as the application domain, the properties of interest, the complexity of the ML tool, and the strategy in building the data from *ab initio* calculations. When making static DFT calculations on chosen configurations, including forces and/or stresses labels make more sense, especially when augmentation of information is performed by generating configurations from it by random atomic displacements. Here a strategy solely based on energy labels for the training is chosen since the data consists of sampled AIMD trajectories for each thermodynamic state whose accessible microstates explore, through the thermal fluctuations, their local basin of the PEL, taking implicitly their gradients into account [146].

3.2.3 Training the Neural Network

The supervised training is carried out using as input the BP feature vectors describing local atomic environments in each configuration. AIMD energies of these configurations are used to find the optimal set of weights and biases. The complete dataset of configurations is firstly randomized and scaled using the standard scaler of Scikit-learn, *i.e.* centering the feature

components about their mean and normalize them according to their standard deviation. It is then split into a training set of 80% of the data and a test set containing the 20% remaining part. In the training set 20% of the data are retained further to create validation sets. They are used (i) for a cross-validation procedure to estimate the performance of various NN architectures through the MSE, and (ii) to monitor the MSE on the validation data during the learning process to detect overfitting. For a given architecture, the optimization is performed using the training data without the validation set, and terminating when the validation error starts to increase. Reduction of the noise of the MSE during training is obtained by including a callback with a stepwise reduction of the learning rate. Simultaneously, a L_2 norm regularization with strength 10^{-5} is performed to reduce the model complexity, and thus to prevent overfitting. Once trained, the weights and biases are stored in a format compatible with the LAMMPS HDNNP pair-style [91].

This training stage is repeated with various NN architectures to find the optimal one capturing at best the functional dependence of the data. Evaluation of the MSE is carried out through a stochastic gradient descent minimization using the Adam optimization algorithm [161] giving a measure of the loss with a learning rate starting at 0.01 and reducing most of the time to 0.0001 during the training, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\varepsilon = 10^{-8}$. The early-stopping was performed with maximum loss variation of 10^{-6} and a patience of 45 epochs. The typical duration of the training period was about 10000 to 15000 epochs. The least MSE loss is obtained for an architecture of 10 neurons in the first and second hidden layers. A typical evolution of loss and the validation loss is shown in Fig. 3.2(a). A cross-validation performed over 5 independent trainings gives a RMSE of (1.2) meV on the per atom energy. Figure 3.2(b) displays the predictive ability of the model on the unseen data of the test set, with a high quality over the whole range of the energies.

The predictive ability of the HDNN is illustrated in Fig. 3.3 on the three forces components extracted from AIMD configurations of a simulation at 1500 K over 2 ps, with a RMSE of 0.074 eV/Å at this high temperature. At $T = 10$ K in similar sampling conditions the RMSE reduces to 0.030 eV/Å. These RMSE values are consistent with those obtained for previous trained ML potentials for Al for which forces were included explicitly in the training [144, 145]. Our results lead to similar conclusions for molecular systems [176] saying that the forces can be predicted with a good accuracy without being explicitly part of the learning process, and thus is in favor of the supervised learning strategy based only on the energy of the configurations, thus avoiding additional parameters in the loss function.

The HDNN is further tested on the prediction of the energy for configuration along a given AIMD trajectory. Reproducing fluctuations is important as they describe the derivatives of thermodynamic quantities, and for instance the specific heat as the derivative of the energy that will considered in Sec. 3.3.2. Consecutive configurations from AIMD simulation of liquids were considered at $T = 600$ K in the undercooled region, $T = 950$ K in the vicinity of the melting point, at 1500 K far above the melting point at zero pressure, and $T = 8000$ K just above the melting line for a pressure of 322 GPa, as shown in Fig. 3.4. Energy fluctuations are very well reproduced for all the temperatures, even for the extreme values for which the sampling is scarce, as they correspond to the tail of the Gaussian distribution of energy

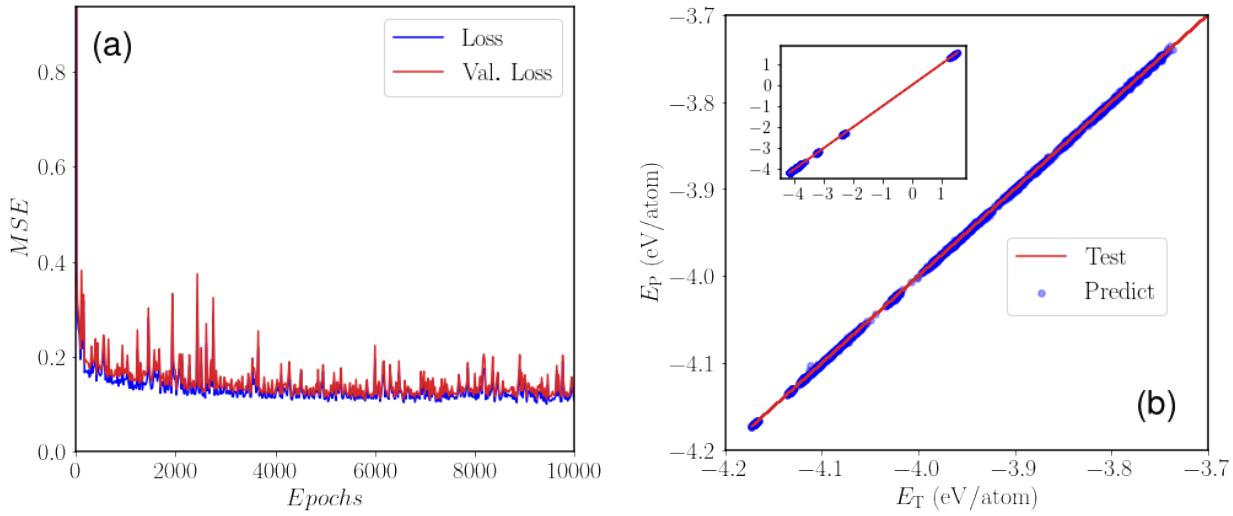

Figure 3.2: (a) Evolution of the MSE losses as a function of number of epochs for the training and validation sets. Loss and Val. Loss correspond to the evaluation of the MSE expressed in eV^2 on the training set and validation set, respectively. (b) Test-Predict curve showing the quality of the prediction on the test set for the optimized architecture $22 \times 10 \times 10 \times 1$. The red solid line represents the known output of the energies representing the know E_T=E_P line that would correspond to a perfect prediction, and the blue dots the values predicted against the known ones. The main panel corresponds to thermodynamic states at low pressures (< 5 GPa) and temperatures between 10 K and 1500 K containing either crystalline and liquid configurations. The inset corresponds to energies in the full range of temperatures and pressures (see text).

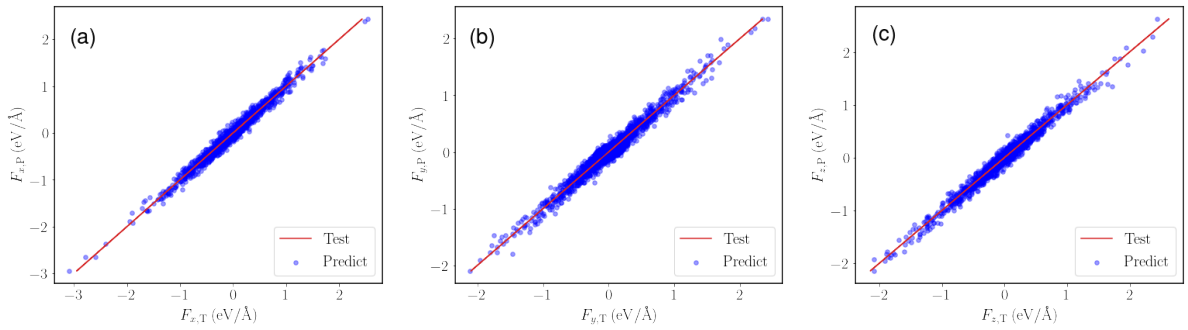

Figure 3.3: Test-Predict curves for the three components of the predicted forces against those extracted from the AIMD configurations at $T = 1500$ K over 2 ps. The red solid line and blue dots have the same meaning as in Fig. 3.2.

fluctuations. Probably the most impressive agreement is that of the simulation at $T = 8000\,\mathrm{K}$ where the energy range of the fluctuation is as large as $0.25\,\mathrm{eV/atom}$ and still very well predicted. It is worth mentioning that none of the configurations considered in Fig. 4 are included in the training set so that the energies for each temperature are purely predictive, and demonstrate the quality of the ML potential.

3.2.4 Molecular dynamics simulation

Classical MD simulations were carried out using the LAMMPS package [133]. These simulations were performed in various ensembles, namely the canonical ensemble (NVT, constant temperature, volume, and number of atoms), the isobaric-isothermal ensemble (NPT, constant temperature, pressure, and number of atoms), and the isobaric-isoenthalpic ensemble (NPH, constant pressure, enthalpy, and number of atoms). Temperature and pressure were kept constant *via* the Nose-Hoover thermostat and barostat [46, 177], respectively. In all simulations, PBC were employed in the three spatial directions. The integration of the equations of motion was done *via* Verlet's algorithm in the velocity form, choosing a time step of 1 fs. The use of our HDNN potential in LAMMPS was possible through the library-based implementation of high-dimensional neural network potentials by Singraber *et al.* [91]. To assess the quality of our HDNN potential, some of the MD simulations were also repeated with the previously published ANI-Al potential of Smith *et al.* [105].

Structural analysis is performed using the common-neighbor analysis (CNA) [56] with the indexing of Faken and Jonsson [52] and a bond-based algorithm as implemented in the OVITO software [178] where a uniform cut-off radius corresponding to the first minimum of the pair-correlation function of the liquid is applied to create bonds between pairs of particles. The CNA classifies pairs around each atom by sets of three indices: the first index represents the number of nearest-neighbors common to this pair, the second index corresponds to the number of nearest-neighbor bonds among the shared neighbors, and the third index indicates the longest chain of bonded atoms among them. For instance, 421 and 422 bonded pairs are characteristic of close packed structures fcc and hcp, respectively. The occurrence of 444 and 666 pairs, with specific proportions, signals the presence of bcc ordering. The degree of five-fold symmetry is obtained from the proportion of 555, 554 and 433 pairs, which represent perfect (555) and distorted FFS based motifs.

An alternate way of studying the local ordering before and during nucleation is to make use of the Steinhardt bond-ordering parameters [57]. More specifically, the averaged form [58] as implemented in the Pyscal code [179] is considered here. First, for each atom i, the following vector is define

$$q_{lm}(i) = \frac{1}{N(i)} \sum_{j=1}^{N(i)} Y_{lm}(\mathbf{r}_{ij}) \tag{3.2}$$

where $N(i)$ is the number of nearest neighbors of atom i, $\mathbf{r}_{ij}$ is the displacement of nearest-neighbor atom j from i, and Y_{lm} is the spherical harmonics. From these, the averaged

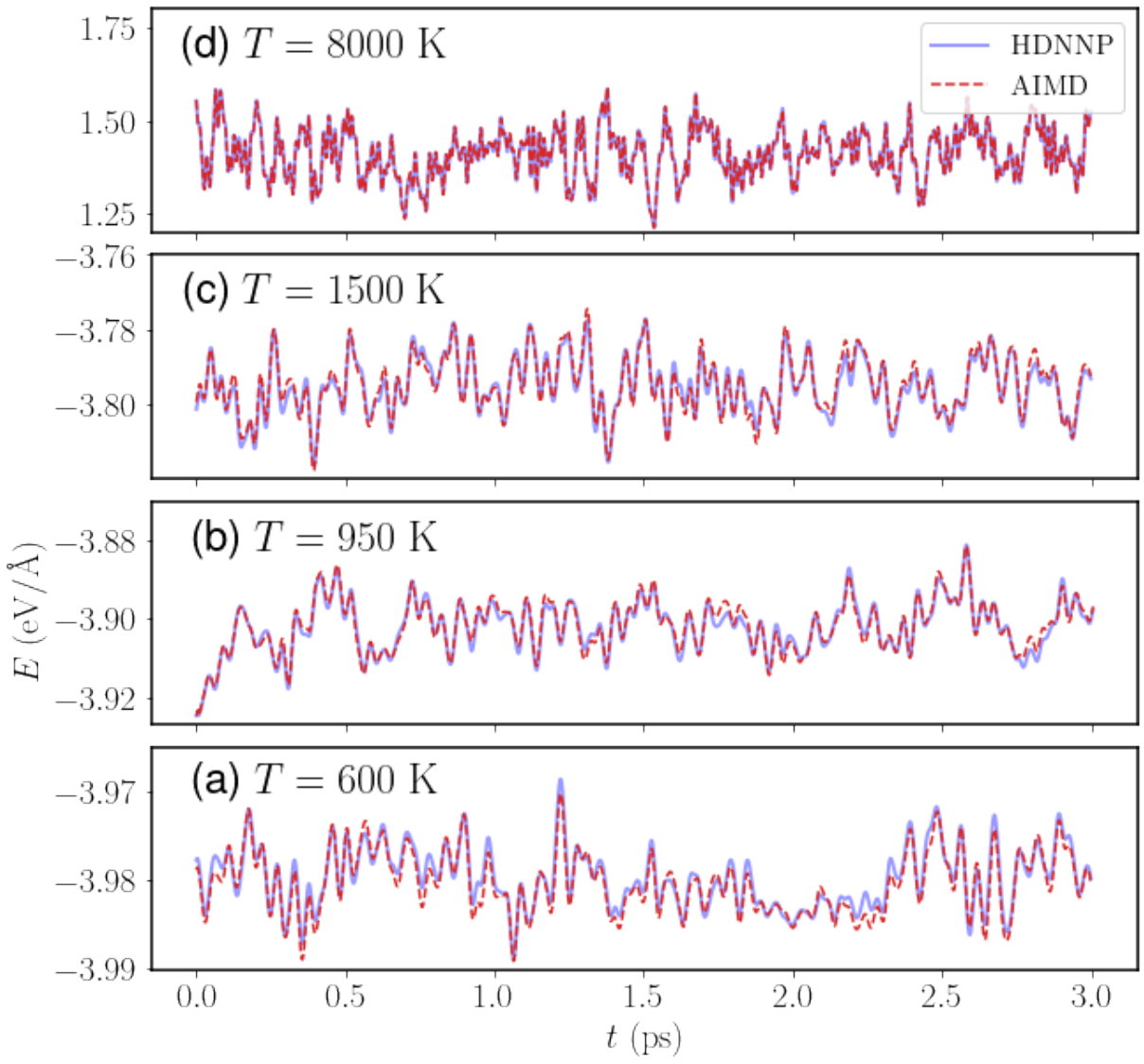

Figure 3.4: Energy per atom as a function of time from AIMD simulation of liquids and predicted by the HDNN potential with $N = 256$ atoms at $T = 600\,\mathrm{K}$ in the undercooled region, $T = 950\,\mathrm{K}$ in the vicinity of the melting point, at $1500\,\mathrm{K}$ far above the melting point at zero pressure, and $T = 8000\,\mathrm{K}$ just above the melting line for a pressure of $322\,\mathrm{GPa}$.

bond-order parameters can be defined as

$$\bar{q}_l(i) = \sqrt{\frac{4\pi}{2l+1}\sum_{m=-l}^{l}\left|\frac{1}{N(i)+1}\sum_{k=0}^{N(i)} q_{lm}(k)\right|^2} \tag{3.3}$$

where the sum from $k = 0$ to $N(i)$ includes both the atom i and its nearest neighbors. Due to being averaged over nearest neighbors, these parameters take into account not just the first coordination shell, but also the second one. To perform structural analysis using these parameters one typically selects specific values for l, with $l = 4$ and $l = 6$ being a common choice. It is then possible to compare the resulting values of $\bar{q}_l$ to those of ideal crystals in order to identify crystal structure.

3.3 Results and discussion

3.3.1 Local structure and dynamics

In a first step, the optimized HDNN potential is assessed on the local structure and dynamics. The simulations are performed at a constant volume, given in top row of Table 3.1, with $N = 10976$ atomsat selected temperatures. For the undercooled states the system is first prepared at a temperature of 1500 K, before being cooled down to the desired temperature. The other states are prepared directly at the target temperature from a fully equilibrated liquid. Following an equilibration time of 10 ps relevant quantities are calculated over a production time ranging from 100 ps to 1 ns depending on the thermodynamic state under consideration.

The pair-correlation function $g(r)$ gives the probability of finding a particle j at distances r_{ij} relative to a particle i located at the origin, and reads:

$$g(r_{ij}) = \frac{N}{V}\frac{n(r_{ij})}{4\pi r_{ij}^2 \Delta r}. \tag{3.4}$$

$n(r)$ represents the mean number of particles j in a spherical shell of radius r and thickness Δr centered on particle i. Finally, an average of $g(r)$ over all N particle i of the simulation box is performed. Integrating $4\pi r^2 \rho g(r)$, with $\rho = N/V$ up to the first minimum $g(r)$ gives access to the mean coordination number. Figure 3.5 displays the curves of $g(r)$ from NVT simulations. An excellent match with AIMD simulations is seen for all the thermodynamic states. A quantitative estimation of the deviation was obtained by calculating the MSE between the classical MD and AIMD curves for each temperature. The MSE ranges from 6×10^{-4} typically in the case of the liquid states to 1.2×10^{-2} in the case of fcc solid states. The larger MSE for the solid might come from the fact that even a very small position shift of $g(r)$ can induce a significant deviation as peaks are sharp and narrow. The curves of $g(r)$ obtained from the ANI-Al ML potential of Smith *et al.* [105] are slightly shifted to larger distances for all liquid states considered, thus overestimating the bond lengths, and their

$T(K)$	300 (s)	600	800 (s)	950	1500	2000	4500	8000 (H)	8000 (L)
V (Å^3)	16.48	18.06	16.68	18.89	20.37	14.00	10.67	7.629	8.616
P (GPa)	0	0	0	0	0	30	110	240	340
	0	0	0	0	0	27	107	227	320
N_C	-	11.93	-	11.57	11.10	12.22	12.70	12.70	12.65
	-	12.05	-	11.65	11.17	12.32	12.80	12.80	12.75
r_1	-	3.727	-	3.775	3.850	3.448	3.225	3.875	3.000
	-	3.752	-	3.800	3.825	3.548	3.225	2.950	3.075
D (Å^2/ps)	-	0.15	-	0.63	1.69	0.35	0.78	1.01	1.35
	-	0.14	-	0.62	1.61	0.37	0.68	0.93	1.34
	-	(0.10)	-	(0.46)	(1.54)	(0.33)	(0.66)	(0.96)	(1.38)

Table 3.1: Atomic volume V, pressure P, coordination number N_C, first minimum r_1 of $g(r)$, and diffusion coefficient D for selected temperatures. Values in second rows are from the AIMD, and those in parenthesis are the diffusion coefficient calculated from the ANI-Al potential [105]. Classical MD simulations for the diffusion were performed with $N = 256$ atoms as for the AIMD one in order to have similar finite size effects. (H), (L), and (s) specifies high and low pressure, and solid state simulations respectively.

peaks in the crystalline states are more pronounced. The average coordination numbers with the HDNN potential display a deviation from AIMD that does not exceed 0.2 (see Table 3.1). Nevertheless, a comparison of $g(r)$ of the present HDNN potential to those obtained with published ML potentials is considered. Fig. 3.5(c) shows that our potential leads to overall better results than the ML potential of Kruglov *et al.* [144] as compared to their *ab initio* simulations and experimental data at $T = 1000$ K. In Fig. 3.5(d) our potential leads to results very close to the recent ANI-Al potential of Ref. [105], and in good agreement with experiments [180] at $T = 1123$ K, $T = 1183$ K and $T = 1273$ K. It is worth mentioning that both mentioned ML potentials have been trained using only the forces [144] or using the energies and forces [105], contrary to the present potential. Additional comparison with the most widely used EAM [156] and MEAM [153] potentials is shown in Fig. S1 in the Supplementary Information File for the same thermodynamic states. They are shown to perform less well than the HDNN potential, as assessed by a t-statistics, and especially for the high pressures.

Beside the local structural properties, dynamic properties represent a stringent test as they are even more sensitive to the details of the potentials. Among these, diffusion plays an important role in the solidification process [3, 181] and was evaluated here through the mean-square displacement (MSD)

$$R^2(t) = \frac{1}{N}\sum_{l=1}^{N}\left\langle [\mathbf{r}_l(t+t_0) - \mathbf{r}_l(t_0)]^2 \right\rangle_{t_0}, \tag{3.5}$$

where $\mathbf{r}_l(t)$ denotes the position of atom l at time t and N is the number of atoms. In addition

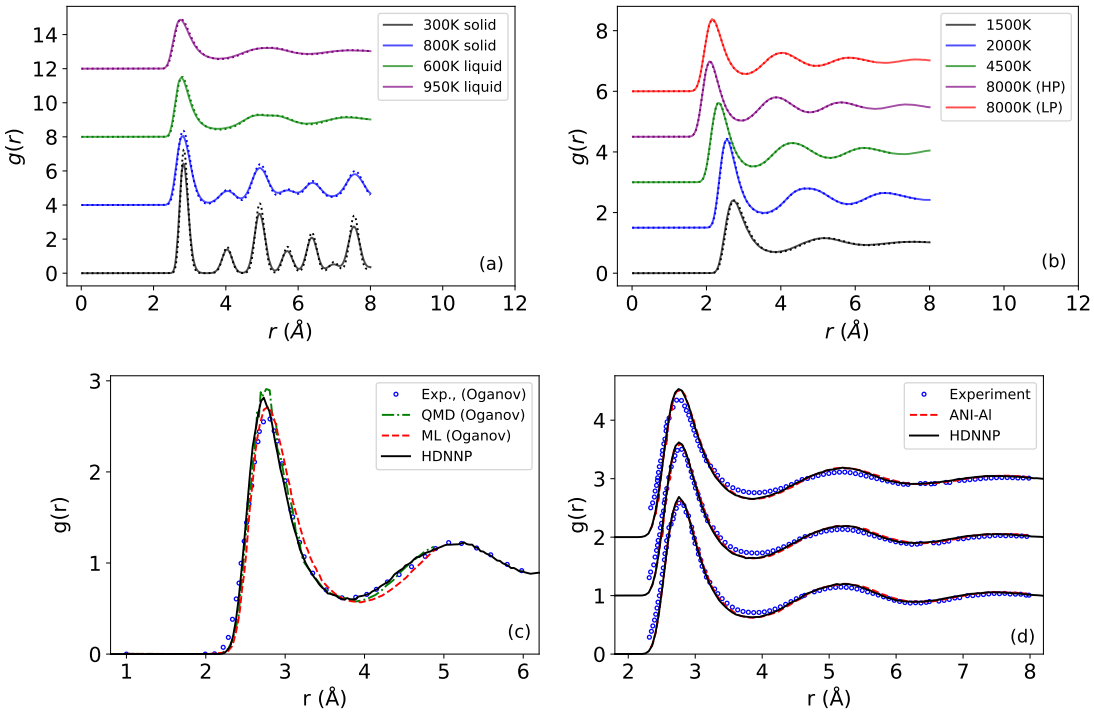

Figure 3.5: Pair-correlation function for various temperatures and pressures (a) for low temperature liquid and solid states at room pressure and (b) for high temperature and pressure liquid states. Curves for 800 K, 600 K, 950 K, 2000 K, 4500 K, 8000 K (340 GPa) and 8000 K (240 GPa) are shifted upwards by an amount of 4, 8, 12, 1.5, 3, 4.5 and 6, respectively. The solid lines are results with the HDNN potential, the dotted lines to the ANI-Al potential, and the dashed lines with corresponding colors are those of the AIMD simulations. High-temperature simulations ran for 500 ps (4500 K, 8000 K (HP)) and 250 ps (8000 K (LP)), with others running for 1 ns. (c) Comparison in the liquid at $T = 1023$ K to simulations with the ML potential of Ref. [144] as well as their *ab initio* calculations and experimental data. (d) Comparison in the liquid at $T = 1123$ K, $T = 1183$ K and $T = 1273$ K to simulations with the ANI-Al potential of Ref. [105] as well as experimental data of Ref. [180].

to the mean over all atoms, an averaging over time origins t_0 as indicated by the angular brackets is performed. The self-diffusion coefficient D is determined from the slope of the linear behavior at long times of the MSD. In Fig. 3.6, the MSD is shown for temperatures in the stable and undercooled liquid states at ambient pressure as well as for temperatures along the melting line for pressures up to 340 GPa. The overall trends of AIMD curves are well reproduced by the HDNN potential for all temperatures and pressures as can be seen in Table 3.1. It is worth mentioning that our potential gives a better prediction of the diffusion coefficients than the ANI-Al potential at ambient pressure and low temperatures where solidification phenomena occur. This is mainly due to the fact that the ANI-Al potential was trained with the PBE functional which is known to underestimate the diffusion coefficient of aluminium as shown in Ref. [101] and this also consistent with the overestimation of the bond length at a given density mentioned above. The MSD curves show a ballistic regime at very short times ($t < 0.05$ ps), followed by a diffusive regime at long times. For the lower temperatures at ambient pressures and at high pressures a well-known caging effect [181] takes place after the ballistic motion and delays the diffusive regime, which is well predicted by the HDNN potential with respect to the AIMD.

The collective dynamics is examined by means of the Intermediate Scattering Function (ISF) $F(Q,t)$ and its time Fourier transform $S(Q,\omega)$, the dynamic structure factor that can be measured by means of Neutron Diffraction. Q represents the wave-number and ω the frequency. Fig. 3.6(c) shows the good agreement of the ISF between the HDNN potential and the AIMD results in the liquid state at $T = 950$ K and $T = 1300$ K for $Q_0 = 2.65$ Å^{-1} corresponding to the position of the first maximum of the static structure factor $S(Q)$. The good match is confirmed for $S(Q,\omega)$ in Fig. 3.6(d) for both temperatures. Further, the comparison with Neutron diffraction data [104] demonstrates that the *ab initio* calculations as well as the HDNN potential predict the dynamic properties of liquid aluminium quite accurately.

The real predictive character of the HDNN potential is assessed for $T = 950$ K, $T = 2000$ K and $T = 8000$ K at the lowest pressure: These thermodynamic states were not included in the training, but the accuracy is similar to the other states for the pair-correlation function and the diffusion. This is all the more true for results at $T = 1123$ K, $T = 1183$ K, $T = 1273$ K and $T = 1300$ K for the comparison with other ML potentials and experiments in Figs. 3.5 and 3.6, as well as for the pressure for all states shown in Table 3.1 with a deviation less than 7% at the highest ones, which is remarkable since the forces were not included in the training.

3.3.2 Thermodynamic properties

Important quantities for solidification phenomena are the latent heat of fusion [182, 183] as well as the densities of the solid and liquid phases at the melting temperature, T_M. Its determination requires the calculation of the enthalpy difference between the liquid and solid branches at T_M. The temperature evolution of the enthalpy at ambient pressure for the solid and liquid branches of the HDNN potential are shown in Fig. 3.7(a), obtained *via* simulation of $N = 2048$ atoms in the NPH ensemble. The simulation is started with a perfect fcc crystal at $T = 300$ K and heated stepwise with a temperature step of 50 K

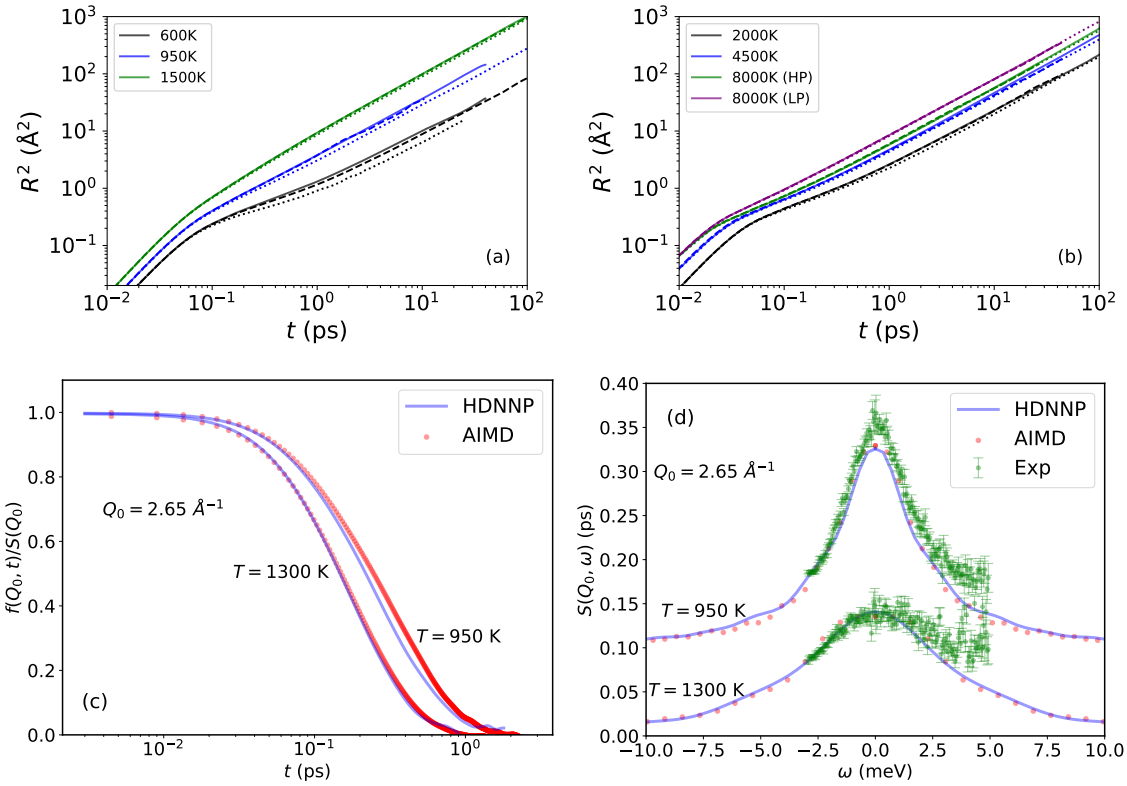

Figure 3.6: Mean-square displacement for various temperatures and pressures: (a) for high temperature and pressure liquid states and (b) for low temperature liquid and solid states at room pressure. The solid lines correspond to the HDNN potential, the dotted lines to the ANI-Al potential [105], and the dashed lines are the AIMD simulations. Classical MD simulations for the MSD were performed with $N = 256$ atoms as for the AIMD one in order to have similar finite size effects. (c) Intermediate scattering function in the liquid state at at $T = 950$ K and $T = 1300$ K for wave vector $Q_0 = 2.65$ Å^{-1}. (d) Corresponding dynamic structure factor at the same temperatures and wavevector that are compared to the neutron diffraction experiments at $T = 943$ K and $T = 1293$ K from Ref. [104].

with an average heating rate of 10^{12} K/s. At each temperature, a simulation is performed over 50 ps (25 ps equilibration and 25 ps production) during which an average value of the enthalpy is calculated. The increase of temperature is repeated until a dynamic melting is observed at $T = 1250$ K. The latter value is noticeably higher than the thermodynamic melting temperature $T_M = 970$ K obtained from liquid-solid interface (LSI) simulations (see Sec. 3.3.3) due to overheating effects. For the liquid branch, the simulations are started at $T = 1600$ K with an equilibrated configuration after the heating process. The same procedure as for the solid branch is followed but with a step-wise cooling down to 300 K that is called here the slow cooling. Above $T = 1250$ K, the difference in the enthalpy from the heating and cooling processes is negligible, indicating that there is no reminiscence of the crystalline state. Below 600 K the liquid undergoes a partial crystallization during cooling. Then, the cooling procedure for the liquid branch is repeated with a higher cooling rate of 10^{13} K/s to avoid crystallization, and a glass transition is seen at $T_G = 378$ K inferred from a crossover between the liquid and glassy branches as shown in Fig. 3.7(a).

For such a high cooling rate of 10^{13} K/s, the time to reach each temperature is shorter than its corresponding nucleation time. This is illustrated from the time temperature transition (TTT) curve plotted in Fig. 3.7(b). It is determined for a system of approximately 1 million atoms (see Sec. 3.3.4) by measuring the time it takes until 40% of the atoms are identified as in a solid state by CNA. At 600 K, the measurement is repeated from the initial configuration, but with new velocities picked from a proper Maxwell distribution, to estimate the variability between measurements.

From the liquid and solid branches an enthalpy of melting of 11.67 kJ/mol is found, which compares reasonably well to the experimental value of 13.34 kJ/mol [184]. Taking the numerical derivative of the solid branch yields a value of the specific heat at constant pressure, C_P, of 0.99 J/g/K which is also in good agreement with the experimental value of 0.91 J/g/K. For the liquid a value of 1.158 J/g/K is obtained, which is in the range of experimental data between 1.03 J/g/K and 1.18 J/g/K close to the latest assessed values of 1.127 J/g/K [184]. The specific heat is a typical derivative quantity that depends on the fluctuations of the enthalpy [46, 177]. The very good agreement is a strong indication that including the time fluctuations from AIMD is a fruitful strategy to describe at least the thermodynamics.

Regarding densities, the HDNN potential gives a value of 0.0587 Å^{-3} and 0.0547 Å^{-3}, respectively in the solid and liquid at its melting point $T_M = 970$ K, giving rise to a density change of 0.004 Å^{-3}. These values compare well to the respective experimental values [185, 170, 184] of 0.0573 Å^{-3} and 0.05306 Å^{-3} with a density change of 0.0042 Å^{-3}. At the experimental melting temperature, the calculated density change remains essentially unchanged, and the densities in both phases deviate only by 2% with respect to the measurements.

3.3.3 Liquid-solid interfaces

Liquid-solid interface simulations are performed for the purpose of determining the melting line by the two-phase coexistence. The procedure follows the approaches proposed in Refs. [186, 187, 188, 189, 190, 191, 192, 193, 194] and is similar to the protocol used in

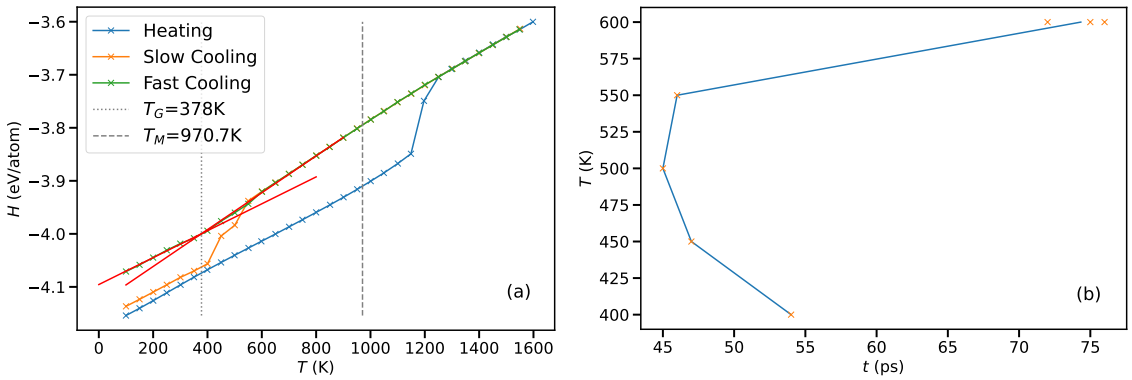

Figure 3.7: (a) Enthalpy as a function of temperature for the solid (from heating) and liquid branches (from cooling) at ambient pressure as described in the text. The red lines indicate the slope of the amorphous and the supercooled states as a guide for the eyes for the crossover between the liquid and amorphous regimes, and marked by the vertical dotted line. The vertical dashed line marks the melting temperature obtained from the the LSI simulations. (b) The TTT curve, as described in the text. The orange crosses are individual measurements of the time it takes to reach 40% solidification, with the blue line connecting the averages for each temperature.

Ref. [31]. A simulation cell containing $N = 5488$ atoms is set up with an initial crystalline configuration with a shape corresponding to $28 \times 7 \times 7$ primitive cells on which PBC are applied to the three directions of space. Starting at zero pressure, this system is heated and equilibrated at constant pressure to a temperature of 50 K below a guess of the melting temperature. Half of the simulation cell in the x direction is further heated and maintained at a much higher temperature until a complete melting is observed. The liquid part is then cooled down and equilibrated at a temperature 50 K above the guess, thus creating a solid-liquid coexistence containing two crystal-melt interfaces due to the PBC. The simulation of the entire system is pursued in the isobaric-isoenthalpic ensemble so that the temperature of the LSI is an internal parameter free to evolve toward a steady state corresponding to the thermodynamic melting temperature if both phases survive. The simulation is continued for 1 ns, and the average melting temperature is determined on the last 100 ps when a steady position of the two interfaces is observed. If a complete melting or solidification occurs, the procedure is started over again with a refined guess of the melting temperature. This procedure is repeated with subsequent higher pressures by first shrinking the volume of the whole simulation cell from the coexistence configuration at the preceding pressure and then increasing the temperature at constant pressure to a new guess of the melting line.

The melting curve of aluminium was measured [195, 196] up to 80 GPa using diamond anvil cells (DAC) and even higher at 125 GPa by means of shock experiments [197]. In Fig. 3.8, the results obtained from the HDNN are compared to these experimental data [195, 196, 197], the *ab initio* based equation of states (EOS) [174] as well as the AIMD two-phase approach [198] for which the GGA for the XC functional and 512 atoms were used. At ambient pressure,

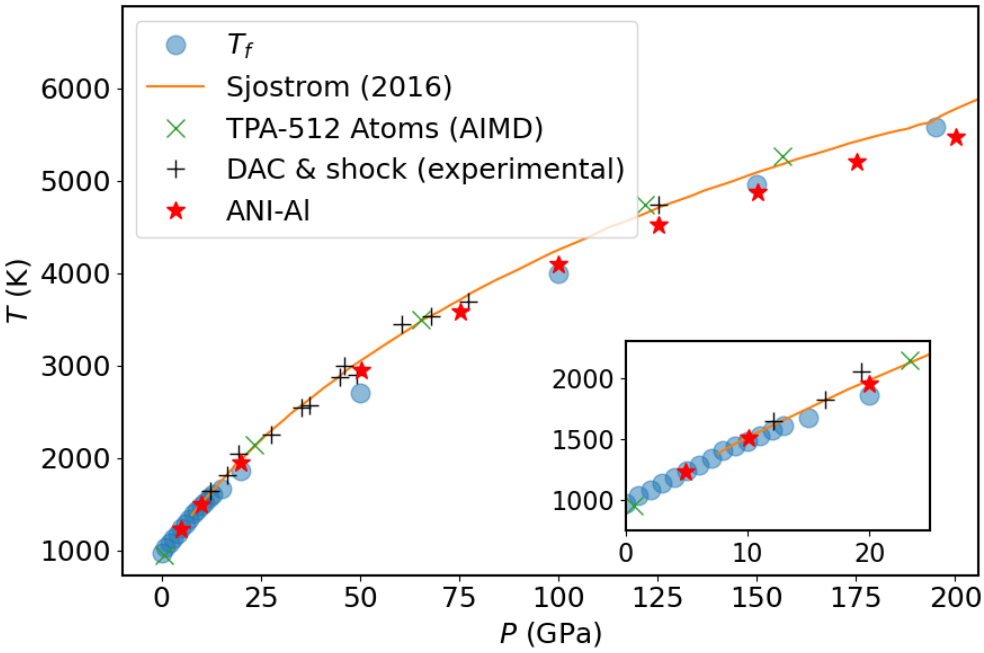

Figure 3.8: Melting curve for aluminium as a function of pressure. The blue circles obtained from our LSI simulations with the HDNN are compared to the experimental data [195, 196, 197], the AIMD two-phase approach [198], the ANI-Al ML potential [105], and the equation of state of Sjostrom et al. [174]. The Inset highlights the lower pressure range up to 25 GPa.

the HDNN potential yields a value of $T_M = 970$ K which overestimates the experimental one of 933 K by 5%. This is also the case for the AIMD [198] to a lesser extent, recalling that the GGA was used and overestimates the atomic volume [171, 198]. With increasing pressure, the melting curve from the HDNN potential slightly underestimates the experiments as well as the EOS. By using two different sizes and shapes, negligible influence on the determination of the melting curve was found, confirming earlier results on pure Zr [31]. Noticeably, the reliability of the present potential on the melting line up to 200 GPa is then assessed, even if high pressure thermodynamic states included in the training set are really scarce. Interestingly, the HDNN curve is similar to the one obtained with the ANI-Al ML potential by Smith *et al.* [105].

3.3.4 Homogeneous Nucleation

Finally, the homogeneous nucleation is investigated and depicted in Fig. 3.9. As pointed out in our preceding work [31], a more accurate investigation require the use of large enough simulation boxes, with typically 1 million atoms or more. This allows the occurrence of multiple nuclei during the nucleation process. Therefore, the system of 1000188 atoms at ambient pressure is analyzed further along the 600 K isotherm used for the determination of

the TTT curve in 3.7(b). At such a high degree of undercooling $\Delta T = (T_M - T)/T_M = 0.38$, an extremely fast nucleation process is observed [183, 182]. Similarly, such a fast homogeneous nucleation is seen at the high pressure of 200 GPa along the 4000 K with $\Delta T = 0.24$. Inherent structure configurations shown in Figs. 3.9 (c)-(j) were first analyzed using the common-neighbor analysis [52] and only atoms with a crystalline environment (fcc, hcp, and bcc) are shown. As expected, nucleation occurs showing growing nuclei in the fcc ordering with hcp stacking faults at ambient pressure. At 200 GPa, nucleation starts with nuclei having a bcc order with sometimes some fcc ordering at their boundary that transform back to the bcc structure during the growth. These nucleation pathways are pretty much consistent with the (P, T) phase diagram [174] showing the reliability of the HDNN potential designed here.

The onset of nucleation occurs at about 30 ps at ambient pressure and 4 ps at 200 GPa, earlier that the nucleation time defined for the construction of the TTT curve above. Using the averaged Steinhardt Order Parameters q_4 and q_6 [58], embryos with atoms having a crystalline ordering showed that they dissolve back to the liquid with a size less than 90 atoms in both cases. The latter value does not represent *per se* the size critical nucleus but rather a lower limit. As expected at ambient pressure, the main crystalline phase during the growth was identified as fcc, as can be seen in Fig. 3.10(a), but a significant hcp ordering also appears during the nucleation and remains as stacking fault after complete solidification of the simulation box as shown in Fig. 3.9. At high pressure, the onset of nucleation occurs in the bcc ordering, with fcc and hcp ordering at their surface at later stages.

It is an open question in general, whether the homogeneous nucleation process follows the Landau Theory in which the bcc precursor is favored in the early stages of crystal nucleation [99] or the Ostwald step rule [199] according to which a primary crystal phase could be different from the the fcc one. From the distributions of the q_6 and q_4 shown in Figs. 3.10(b) and 3.10(c) only fcc ordering emerges at the onset of nucleation. Our findings show that aluminium follows a single step process with an onset of homogeneous nucleation showing emerging embryos with a fcc ordering. The resulting nuclei grow in a rather patchy shape with a small amount of hcp stacking fault defects. This nucleation the scenario is different from the Lennard-Jones case [96, 97] which follows the Landau theory and the Ostwald step rule. The present large-scale molecular dynamics results with close ab initio accuracy allows us to assess very recent molecular dynamics simulations [200, 53] with EAM empirical potentials. Our findings further show that such a single step nucleation pathway also occurs at high pressure with bcc ordering in the emerging nuclei.

3.4 Conclusion

In the present work, a machine learning potential for pure aluminium by means of a high dimensional neural network on the basis of the well-known Behler-Parrinello approach [9, 91] was developed. This ML potential is devoted to the description of condensed phases, namely liquid and solid states at ambient pressure as well as those at pressures up to 300 GPa with resulting temperatures as high as 8000 K. A crucial point was the training of the potential with a data set generated by DFT-based simulations not only to cover the targeted domain

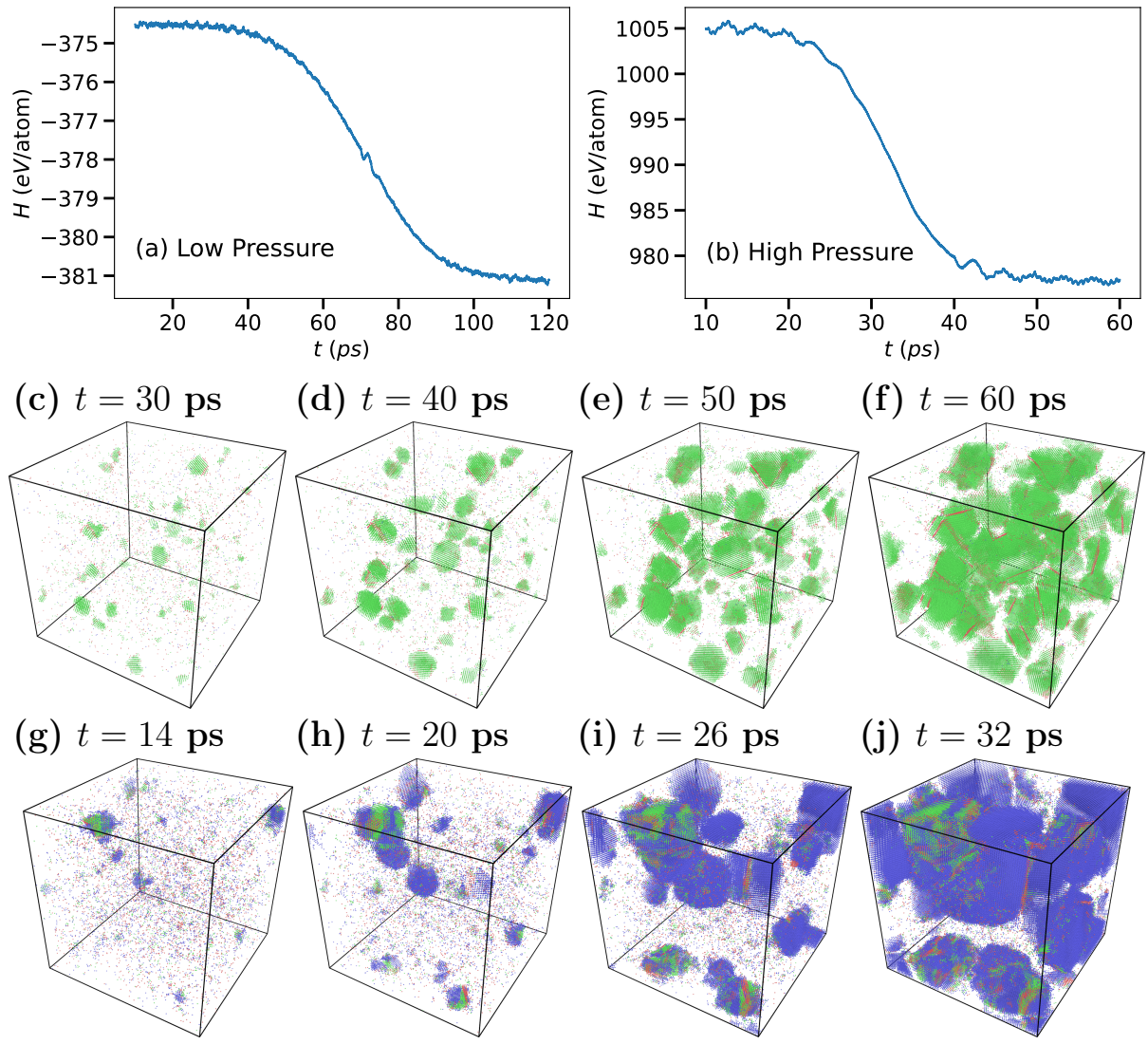

Figure 3.9: Homogeneous nucleation of deeply undercooled aluminium along the $T = 600\,\mathrm{K}$ and 4000 K isotherms, respectively for ambient pressure and 200 GPa. Time evolution of the enthalpy (a) at ambient pressure, and (b) at 200 GPa. Snapshots of the simulation as various times during the nucleation (c) to (f) at ambient pressure, and (g) to (j) at 200 GPa. Only the atoms with crystalline ordering in the sense of the common-neighbor analysis are drawn: fcc (green); hcp (red); bcc (blue).

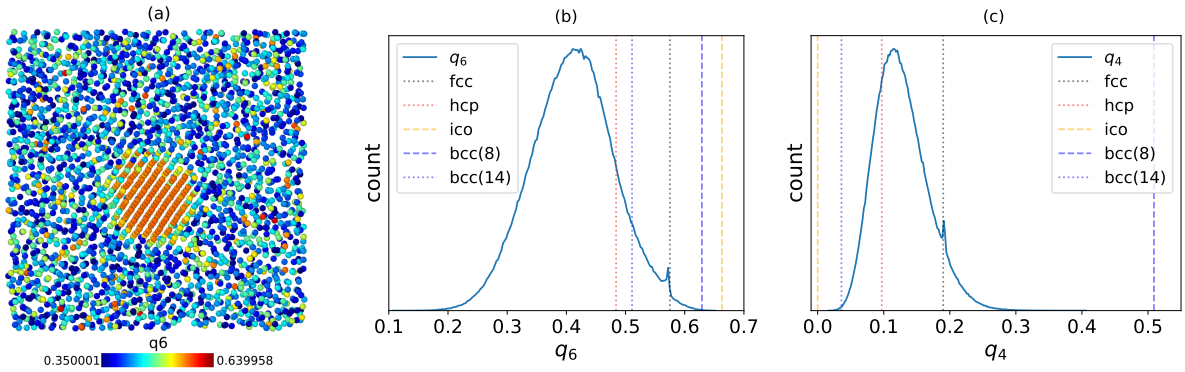

Figure 3.10: (a) First identified nucleus at the onset of nucleation in a small box extracted from the snapshot at 30 ps (Fig. 3.9) for ambient pressure. Atoms are colored according to he value of the q_6 averaged Steinhardt order parameter. (b-c) Corresponding histograms for all atoms in the simulation box of the q_6 and q_4 parameters showing clearly that nucleation starts with only a fcc ordering.

of thermodynamic states for a question of transferability but also to consider for each of them in a physical meaningful manner the time fluctuations by an appropriate sampling of phase space trajectories obtained by *ab initio* molecular dynamics. This allows to include in the training the relevant accessible microstates of the considered thermodynamic states. Another approach based on metadynamics was shown to be efficient in selecting the relevant configurations to train the neural network [109].

The HDNN potential thus obtained was shown to be efficient in reproducing the structural, dynamics as well as thermodynamic quantities in the liquid, undercooled and crystalline states at ambient pressures as well as in the liquid state at high pressure up to 300 GPa, including the melting line. One important outcome is that a reliable ML potential could be obtained without including explicitly the forces in the training by using an appropriate sampling of AIMD trajectories. The procedure was shown for Al and HDNN to perform well, giving a RMSE on forces similar to what is current obtained. The early stages of the homogeneous crystal nucleation was further investigated on a scale much larger than what is possible from the *ab initio* molecular dynamics but with a similar accuracy. Results show that aluminium follows a single step nucleation process with an emerging fcc ordering and hcp stacking fault defects, confirming recent works using large scale molecular dynamics [200, 53], and also consistent with very recent simulations on nucleation during cooling [201]. A single step nucleation pathway with bcc nuclei is also observed at high pressure.

Finally, the fact that the HDNN potential keeps a good accuracy even in domains where the thermodynamic states in the training set are scarce opens up a research line based upon active learning for regression approaches to reduce efficiently the training set. Dynamical properties such as the diffusion coefficients considered here are sensitive to the details of the potential and should be introduced in the training procedure in a more direct way than through the choice of the XC functional in the DFT calculations. This would represent a real step forward in designing ML potentials. Extending the present methodology to other

systems is under progress.

Acknowledgments

We acknowledge the CINES and IDRIS under Project No. INP2227/72914, as well as CIMENT/GRICAD for computational resources. This work was performed within the framework of the Centre of Excellence of Multifunctional Architectured Materials "CEMAM" ANR-10-LABX-44-01 funded by the "Investments for the Future" Program. This work has been partially supported by MIAI@Grenoble Alpes (ANR-19-P3IA-0003). Fruitful discussions within the French collaborative networks in high-temperature thermodynamics GDR CNRS 3584 (TherMatHT) and in artificial intelligence in materials science GDR CNRS 2123 (IAMAT) are also acknowledged. We thank J. Smith and K. Barros for their kind help in setting up the simulations with their ANI-Al potential from Ref. [105]. J. S. acknowledges funding from the German Academic Exchange Service (DAAD) through the DLR-DAAD programme, grant No. 509.

Chapter 4

Homogeneous Nucleation in Binary Aluminium Nickel

The following chapter is based on a preprint, prepared during this doctoral work [17]. In this paper we train a potential for binary Al-Ni alloys. The potential is subsequently used to study homogeneous nucleation in this system. The *ab initio* simulations used to construct the dataset, the training of the potential, as well as all simulations and further calculations, were performed by me. Writing of the paper was shared between me and N. Jakse. Section 4.5 is not part of the preprint, but covers further simulations for this system, examining the interdiffusion.

Homogeneous Nucleation of Undercooled Al-Ni Melts via a Machine-Learned Interaction Potential

Johannes Sandberg[123], Thomas Voigtmann[23], Emilie Devijver[4], Noel Jakse[1]

[1]Université Grenoble Alpes, CNRS, Grenoble INP, SIMaP, F-38000 Grenoble, France
[2]Institut für Materialphysik im Weltraum, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 51170
[3]Department of Physics, Heinrich-Heine-Universität Düsseldorf, Universitätsstraße 1, 40225 Düsseldorf, Germany
DOI: 10.48550/arXiv.2410.07886
[4]Université Grenoble Alpes, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France

Abstract

Homogeneous nucleation processes are important for understanding solidification and the resulting microstructure of materials. Simulating this process requires accurately describing the interactions between atoms, which is further complicated by chemical order through cross-species interactions. The large scales needed to observe rare nucleation events are far beyond the capabilities of *ab initio* simulations. Machine-learning is used for overcoming these limitations in terms of both accuracy and speed, by building a high-dimensional neural network potential for binary Al-Ni alloys, which serve as a model system relevant to many industrial applications. The potential is validated against experimental diffusion, viscosity, and scattering data, and is applied to large-scale molecular dynamics simulations of homogeneous nucleation at equiatomic composition, as well as for pure Ni. Pure Ni nucleates in a single-step into an fcc crystal phase, in contrast to previous results obtained with a classical empirical potential. This highlights the sensitivity of nucleation pathways to the underlying atomic interactions. Our findings suggest that the nucleation pathway for AlNi proceeds in a single step toward a B2 structure, which is discussed in relation to the pure elements counterparts.

4.1 Introduction

The macroscopic properties of a material are closely tied to its microstructure, and understanding their relationship is a central topic in materials science [202]. On a fundamental level, the observed properties originate from the chemical and physical interactions that shape the atomic arrangements and ultimately determine these properties. It is therefore vital to understand the microscopic nature of crystal nucleation, and growth when the liquid morphs into the solid, if one wants to control the microstructure. Especially, the early steps of the microscopic nucleation process cannot generally be directly observed in experiment, with the very recent exception for Fe-Pt binary metallic nanoparticles by atomic electron tomography [203]

that still require to be combined with atomic scale simulations. Often, systems such as colloids [204] are use as model systems studied experimentally for nucleation, still necessitating a computational treatment [3]. Due to the presence of solvent-mediated many-body interactions, it is also not clear whether the kinetic pathways are the same in colloids as in metallic or molecular systems [205]. Simulations at the atomic scale such as Molecular Dynamics (MD) remain one of the dedicated tools that allow for the direct *in silico* study of critical nuclei, their growth, and the local structures in which they emerge from the melt, as reported for generic models as well as metals and alloys [206, 207, 208, 209, 210, 211, 31, 53, 30, 212].

Due to the low probability of observing rare crystal nucleation events, large scale simulations are required. For pure metals and alloys of interest here, empirical or semi-empirical interaction potentials, such as embedded atom models [118, 119], modified embedded atom models [120, 153], and reactive force fields [213], are traditional choices for such simulations, owing to their high performance. There are, however, often large discrepancies between such force fields and experimental results [214, 3]. A better approximation can in principle be achieved by first-principles approaches. *Ab initio* simulations based on Density Functional Theory (DFT) [25, 26], through Car-Parrinello *ab initio* molecular dynamics (AIMD) [23, 4], provide a more faithful representation of the interatomic interaction. Those ab initio simulations enable the accurate study of arbitrary mixtures of species in various phases, and have seen success across much of material physics [215, 216, 217, 218, 101, 219, 142, 220, 221, 222, 223]. Unfortunately, computational cost and poor scaling with system size put *ab initio* simulation at odds with the large scale required for observing rare nucleation events.

Machine Learning Interatomic Potentials (MLIPs) were introduced to overcome the limitations of both classical potentials, and *ab initio* simulations. Since the seminal work of Behler and Parrinello [224], a wide range of methods for constructing data-driven potentials have been proposed. These range from linear-regression-based models [35, 12, 225, 15], kernel-based methods such as Gaussian approximation potentials [10, 137, 226], descriptor-based neural network approaches such as the Behler-Parrinello High Dimensional Neural Network Potentials (HDNNP) [224, 227, 34], Deep Potentials [39], and more recently graph neural network potentials [228, 41]. By fitting a machine-learning regression model to *ab initio* potential energy surfaces, it is possible to effectively interpolate them at a computational cost that is orders of magnitude lower. Molecular dynamics studies of nucleation, which require both accurate potentials capable of representing multiple phases and long, large-scale simulations to capture rare nucleation events, benefit greatly from MLIPs [8, 16].

Al-Ni alloys are known for their mechanical performance, thermal stability, and advanced functional properties, making them versatile in both structural and functional applications. The structural ordering in Al-Ni melts has therefore drawn a lot of interest over the past decades, aimed at understanding the structure-dynamics relationship, and in particular the impact of chemical short-range ordering [229, 230, 231, 232, 233, 234, 235]. Meanwhile, the challenges of experimentally measuring Al self-diffusion coefficients make MD simulations a vital tool for studying this system, and Al alloys more broadly. Molecular dynamics simulations using the first EAM potential by Mishin [236] were used to study self and inter diffusion in Al-Ni across various compositions. These simulations [233] found different diffusivities between

the two species, even in Ni-rich compositions, in contrast to the *ab initio* results of [235]. Recently, classical simulations using a new version of the EAM potential of Mishin [237] investigated nucleation into the bcc-based B2 structure at equiatomic composition, and the two-step nucleation of pure Ni into fcc via bcc precursors driven by local icosahedral ordering [30]. Nucleation at AlNi and $AlNi_3$ compositions were studied using a novel structure identification technique, showing a two-step process for the Ni-rich composition, where chemical ordering of the final phase appearing before bond-orientational ordering during nucleation onset [98].

A reliable description of early stages of crystal nucleation requires an accurate description of the structural and the diffusion in the liquid as well as the crystalline states. Improvement of the interatomic interactions in Al-Ni alloy at the *ab initio* accuracy can be achieved in the framework of machine learning, which has proven to be able to tackle nucleation phenomena for pure aluminium [16].

In the present work, homogeneous crystal nucleation pathways are studied for $Al_{50}Ni_{50}$ as well as pure Ni. For this purpose, a HDNNP [34] is first trained on AIMD trajectories for Al-Ni alloy in the the whole composition range including pure Ni and pure Al, in the liquid states above and below the melting point as well as relevant crystalline phases at low temperatures. The trained potential is first successfully validated on the local structure as well as self- and inter-diffusion, examining the temperature and composition dependence. Homogeneous nucleation under deep undercooling is then investigated with these potentials by means of large-scale MD simulations with 135000 atoms for the two compositions. For the equiatomic composition a single-step solidification process into the B2 phase is observed. Such a nucleation pathway is discussed in view of the nucleation of their pure elements counterparts. Furthermore, in all the cases nuclei show a relatively irregular, non-spherical shape, different from what is assumed in Classical Nucleation Theory (CNT).

The remaining part of the paper is organized as follows. Section 4.2 describes the construction of the MLIP and the various properties of interest in the present work. Section 4.3 is devoted to the validation of the potential as well as the analysis of homogeneous nucleation for the two compositions as well as pure Ni. The conclusions are provided in Section 4.4.

4.2 Computational background

4.2.1 Dataset: *Ab initio* molecular dynamics trajectories

Care must be taken in designing a dataset from *ab initio* calculations, to ensure that the configurations added to the dataset are representative of those encountered during a molecular dynamics run. A straightforward way of doing this is to perform an *ab inito* molecular dynamics simulation, and sampling the resulting trajectories.

All AIMD trajectories were built in the *Vienna Ab Initio Simulation Package* (VASP) [238] within the canonical ensemble, with constant number of atoms N, constant volume V, and constant temperature T (NVT), the average temperature being fixed using a Nose-Hoover

thermostat [239, 240]. The dynamics were performed by solving numerically Newton's equations of motion using the Verlet algorithm in the velocity form with a timestep of 1.5 fs. The electronic system was treated with electron-ion interactions represented by the projector augmented wave (PAW) potentials [241, 167] and exchange and correlation effects were taken into account with the Generalized Gradient Approximation (GGA) [242] in the Perdew, Burke, and Ernzerhof (PBE) [166] formulation. The cutoff energy in the plane-wave expansion was taken as 300 eV. For such large supercells and in the liquid state, only the Γ point was used for sampling the Brillouin zone.

The dataset used in this work was constructed in such a way, with configurations drawn from a number of *ab initio* simulations performed specifically for this task. The majority of these simulations were performed at Ni compositions $x_{\mathrm{Ni}} = 0.25$, 0.5, and 0.75, as well as pure Ni, and pure Al, with the aim of covering the entire composition range. For each composition, two branches of simulations were performed, one starting from a high temperature liquid, quenching in steps of 200 K, and one branch starting from a 10 K crystalline state obtained from the *materials project* database [243], and increasing the temperature in steps of 200 K. These simulations were carried out for 60 ps, with some of the liquid alloy simulations continued up to 90 ps in order to ensure that the correct chemical ordering appears and is included in the dataset. A few runs were also performed at a higher pressure, at high temperature, to probe the short-range interaction, and improve the stability of the trained potential. In addition to these primary simulations in the liquid state, additional simulations were performed at 10 K for the stable Al_3Ni_5, Al_4Ni_3, and Al_2Ni_3 crystalline phases. The five unstable states on *materials project* closest to the convex hull were included in the same way, namely mp-1183232, mp-1228868, mp-1025044, mp-1229048, and mp-672232. Figure S1 in the supplementary information file shows the x_{Ni}-T locations of the AIMD trajectories as well as the number of configurations sampled from each of them.

4.2.2 High Dimensional Neural Network Potentials

With the *ab initio* dataset in place, the next step is to define the machine learning model used to construct our MLIP, here the Behler-Parrinello HDNNP [224, 227]. In this framework the potential energy of the many-body system is given *via* a nearsightedness approximation,

$$U \approx \sum_{i=1}^{N} U_i, \tag{4.1}$$

where N is the number of atoms in the system, and U_i gives the local energy contribution of atom i, dependent only on atoms j within some neighborhood $r_{ij} = |r_j - r_j| < r_c$ defined by some cutoff distance r_c. In the Behler-Parrinello HDNNP approach, U_i is chosen as a species-dependent neural network, taking as input an atomic fingerprint, here given by the

Behler-Parrinello Symmetry Functions (SFs) [244]:

$$G_i^2 = \sum_j e^{-\eta(R_{ij}-R_s)^2} f_c(R_{ij}) \tag{4.2}$$

$$G_i^5 = 2^{1-\zeta} \sum_{j,k} (1 + \Lambda \cos\theta_{ijk})^\zeta e^{-\eta(R_{ij}^2+R_{ik}^2+R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \; . \tag{4.3}$$

Here, R_{ij} is the distance between atoms i and j, θ_{ijk} is the angle between atoms j and k with respect to atom i, and $f_c(R_{ij})$ is defined as 0 for $R_{ij} > r_c$ and for $R_{ij} < r_c$ as a polynomial going smoothly to 0 at the neighborhood cutoff $R_{ij} = r_c$. The parameters η, ζ, Λ, and R_s allow for defining a set of features by assigning to these parameters different values.

Once the number of SFs used for each species, their parameters, as well as the architecture of the NNP of each species, were chosen, the weights and biases of the NNPs are optimized by minimizing the mean-square error of the HDNNP predictions, with respect to the *ab initio* training dataset. A validation dataset and a test dataset are considered, independent of the training dataset. The validation set is used to evaluate the extrapolation error, to guide the tuning of hyperparameters, and to decide when to end training as part of early stopping. The test dataset is then used to evaluate the final model, after training is completed. All the settings of the HDNN and the parameters of the SFs can be found in [245].

While the HDNNP fits the potential energy surface, it is often beneficial to fit not only the potential energy, but also its gradients, namely the *ab initio* forces. In this work, however, only the energy was considered for the fitting of our HDNNP model. This is to maintain continuity with our previous work on the solidification of pure Al, as well as to not potentially reduce the accuracy of the energetics, which play an important role in the phenomenon of nucleation. This will also allow for a more straightforward future extension of our previously proposed embedded feature selection method [18] to this multicomponent system. Training itself was performed using the N2P2 package [132, 91].

4.2.3 Classical simulations and analysis

Classical simulations were performed in LAMMPS [133], using the *ml-hdnnp* plugin provided by N2P2 [91]. Time integration was performed using the velocity Verlet algorithm, with a timestep of 1 fs. Simulations for validating the potential were performed in the NVT ensemble, with temperature fixed by a Nosé-Hoover thermostat, and volume pressure controlled indirectly by fixing the volume. MD simulations of homogeneous nucleation were performed in the NPT ensemble, adding a barostat fixing the pressure to ambient conditions. Visualization of the system, common neighbor analysis, and calculation of the pair-correlation functions were performed using OVITO [178]. Calculation of structure factors were performed using ISAACS [246], and the bond-order parameters were calculated with pyscal [179].

4.2.4 Structural and dynamic properties

Basic structural information of the melt can be obtained from the pair-correlation functions (PCF) [247, 248]. The partial PCF, $g_{ij}(r)$, gives the probability of finding a particle of type i

at a distance r from a particle of type j, and can be obtained from simulation by measuring the number of j particles in a spherical shell of thickness Δr and radius r. The ensemble average $n_{ij}(r)$ then gives

$$g_{ij}(r) = \frac{N_j}{V} \frac{n_{ij}(r)}{4\pi r^2 \Delta r}, \tag{4.4}$$

with V the volume of the simulation box, containing N_j total number of j particles. A Fourier Transform of these partials leads to the partial structure factors

$$S_{ij}(q) = \frac{1}{N} \left\langle \sum_{k=1}^{N_i} \sum_{l=1}^{N_j} \exp\left[i\mathbf{q} \cdot (\mathbf{r}_k^i - \mathbf{r}_l^j) \right] \right\rangle, \tag{4.5}$$

written here in the Faber-Ziman form [247]. Weighting the $S_{ij}(q)$ with appropriate scattering lengths from neutron or x-ray diffraction leads to the corresponding total Structure factors $S_N(q)$ or $S_X(q)$.

The self-diffusion of atoms in the melt is obtained *via* the Mean Squared Displacement (MSD) at time t given by

$$R_i^2(t) = \frac{1}{N_i} \sum_j^{N_i} \left\langle \left[\mathbf{r}_j(t + t_0) - \mathbf{r}_j(t_0) \right]^2 \right\rangle, \tag{4.6}$$

where the the sum runs over atoms of species i, $\mathbf{r}_j(t)$ denotes the position of atom j at time t, and $< . >$ averages over the initial timestep t_0. From this quantity, the self-diffusion D_i can be extracted via the Einstein relation

$$D_i = \lim_{t \to \infty} \frac{R^2(t)}{6t}. \tag{4.7}$$

The shear viscosity of the system is calculated via a Green-Kubo relation from the stress-stress autocorrelation function

$$\eta = \frac{V}{k_B T} \int_0^\infty dt \, \langle P_{xy}(t) P_{xy}(0) \rangle, \tag{4.8}$$

with V the volume of the system, T the temperature, k_B the Boltzmann constant, and $P_{xy}(t)$ the off-diagonal component of the stress tensor at time t. Note that, for an isotropic system, we can average over the xy, xz, and yz components in evaluating the ensemble average.

4.3 Results and discussion

4.3.1 Structure and Dynamics

This section is devoted to the validation of the trained HDNNP on the structural properties, such as the pair-correlation functions and the total and partial structure factors, as well as the dynamics through the self-diffusion coefficients and the viscosity. Comparison with AIMD

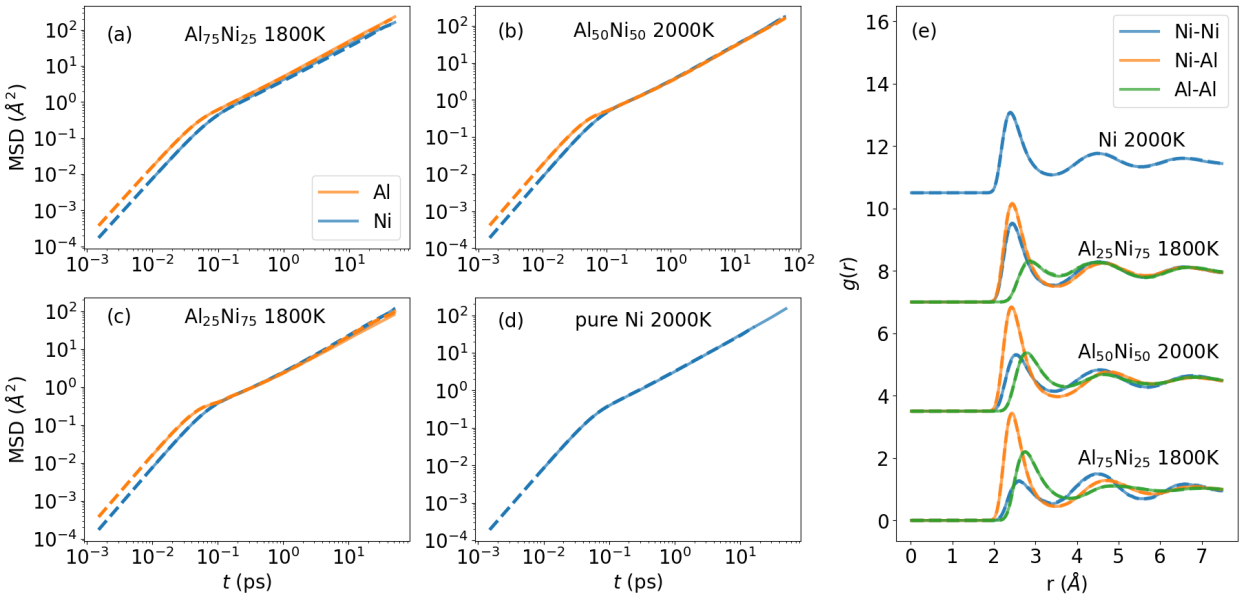

Figure 4.1: Comparison between NNP and AIMD results for the melt. NNP results are shown as solid lines, AIMD as dashed lines. MSD values are for (a) 1800 K for $Al_{75}Ni_{25}$, (b) 2000 K for $Al_{50}Ni_{50}$, (c) 1800 K for $Al_{25}Ni_{75}$, and (d) 2000 K for pure Ni. PCFs are shown in (e) for the same temperatures, with subsequent compositions shifted upwards by 3.5, 7 and 10.5.

Table 4.1: Self diffusion coefficients extracted from the mean square displacements in Figure 4.1 for each Ni composition, x_{Ni}. Values in parentheses are from AIMD.

x_{Ni}	0.25	0.50	0.75	1.0
T (K)	1800	2000	1800	2000
D_{Ni} $[10^{-9}m^2/s]$	5.108 (5.578)	5.003 (4.858)	3.884 (3.640)	4.980 (4.977)
D_{Al} $[10^{-9}m^2/s]$	7.598 (7.414)	4.527 (4.316)	2.646 (3.043)	-

assesses the quality of the training procedure, while comparison with experiments tests the reliability of the *ab initio* simulation scheme.

As a first evaluation of the potential, the HDNNP is compared to *ab initio* simulations, by performing a set of simulations with $N = 256$ atoms corresponding to the AIMD system size (to exclude finite-size effects in the comparison). A set of configurations were taken from the end of the *ab initio* trajectories as an initial state. The temperature, matching that used in *ab intio*, was chosen to be in the liquid, just above the experimental liquidus line. After initializing the atom velocities according to a Maxwell-Boltzmann distribution, an equilibration of 100 ps was performed, followed by a 200 ps production run.

The mean-square displacements are shown in Figure 4.1 (a)-(d), along with those calculated from the *ab initio* trajectories. The corresponding values for the self-diffusion coefficients, obtained from the MSD *via* the Einstein relation given by Eq. (4.7) for both the NNP and for *ab initio* are in good agreement, as shown in Table 4.1. Note that the largest disagreement occurs for the minority components, Al in $Al_{25}Ni_{75}$ and Ni in $Al_{75}Ni_{25}$, with other components typically agreeing to within roughly $0.2 \times 10^{-9} m^2/s$, which can be seen as a reasonably good agreement.

The partial pair-correlation functions are displayed in Figure 4.1 (e), and it can be seen that the NNP curves show a nearly perfect match with the AIMD ones. Especially, it reproduces the quite strong affinity between Al and Ni revealed through the large amplitude of the first peak of the Al-Ni partial with respect to the Al-Al and Ni-Ni ones. This indicates a chemical short-range ordering [235] that has a maximum for $Al_{50}Ni_{50}$. Table SI of the supplementary information file (SI) shows that for this composition the first peak maximum of the Al-Ni partial is the maximum. All these results highlight the high quality of the training procedure in the whole range of compositions.

A comparison with experimental data is now done through the total and partial structure factors, the temperature and composition dependence self-diffusion coefficients as well as the viscosity. Figure 4.2(a) shows the structure factor from the HDNNP and compared to the recent neutron diffraction experiments of Belova *et al.* [249] and Maret *et al.* [229]. An excellent agreement is found for the peak positions as well as their amplitudes for all the considered compositions. The partials structure factors in the Faber-Ziman formulation [247] are shown in Figure 4.2(b) for the four different compositions. For $Al_{80}Ni_{20}$, the partials agree remarkably well with the data of Maret et al. [229]. Another interesting feature is that the amplitude of the pre-peak of the Ni-Ni partial decreases with increasing Ni composition and is seen to correlate tightly with that of the total structure factor. This suggests that the pre-peak of the total structure factor might originate from the Ni-Ni structural features. It was argued this pre-peak could be a signature of the chemical short-range order [250], and it was later found that it might be characteristic of a medium range ordering [251, 252].

The self-diffusion behaviour with respect to the composition is examined, seeking to match the experimental data points in [232]. At each composition the system is initiated in a 13500 atom fcc crystal, with atoms randomly assigned to each species, in the given ratio. The system is then melted at a high temperature of roughly 2000 K for 100 ps, then quenched to the final temperature over 10 ps, and equilibrated for 100 ps, all in a NPT ensemble.

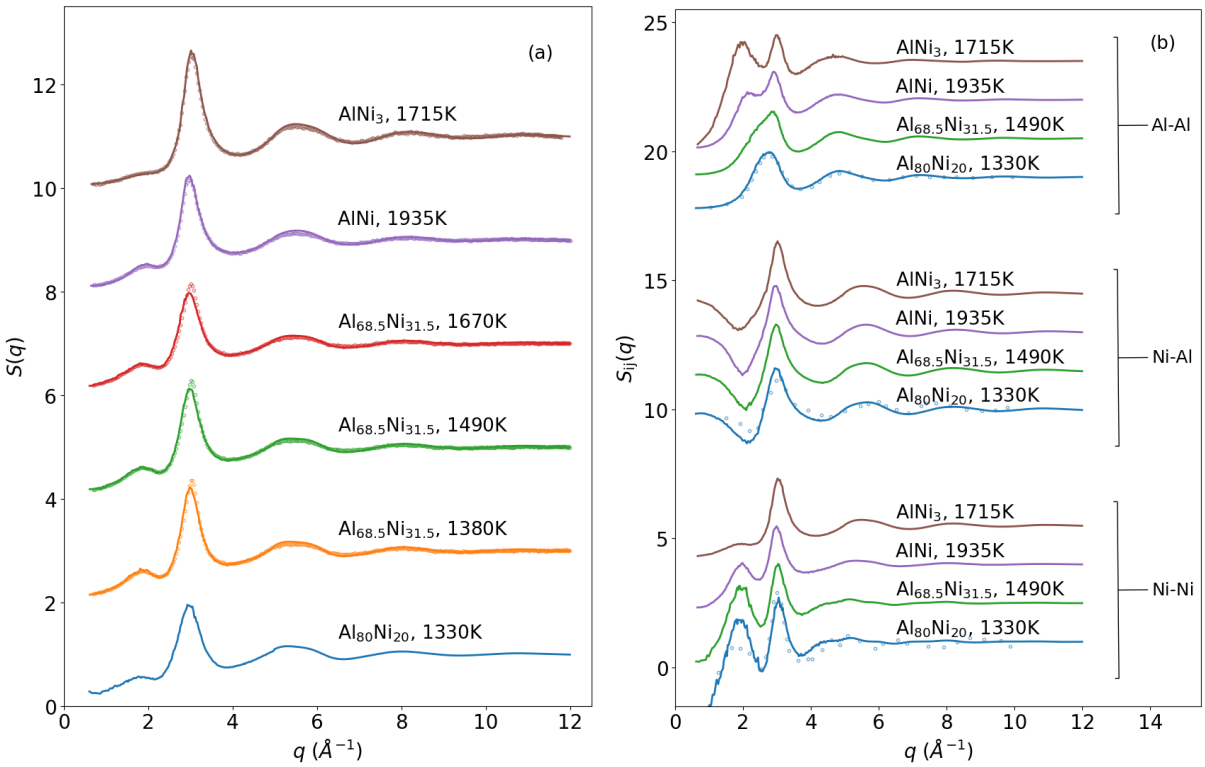

Figure 4.2: Total structure factor from neutron diffraction for liquid Al-Ni above the liquidus (a), and corresponding Faber-Ziman partial structure factors (b). Comparison between HDNNP (solid lines) and experimental total structure factor [249] (symbols) for $Al_{25}Ni_{75}$ at 1715 K, $Al_{50}Ni_{50}$ at 1935 K, $Al_{68.5}Ni_{31.5}$ at 1700, 1490, and 1380 K. and $Al_{80}Ni_{20}$ at 1935 K. The curves are shifted upwards by an amount of 2. The partial structure factors $Al_{80}Ni_{20}$ was compared to the experimental data of [229]. The different partials are shifted by an amount of 9 and subsequently shifted by 1.5 for the compositions.

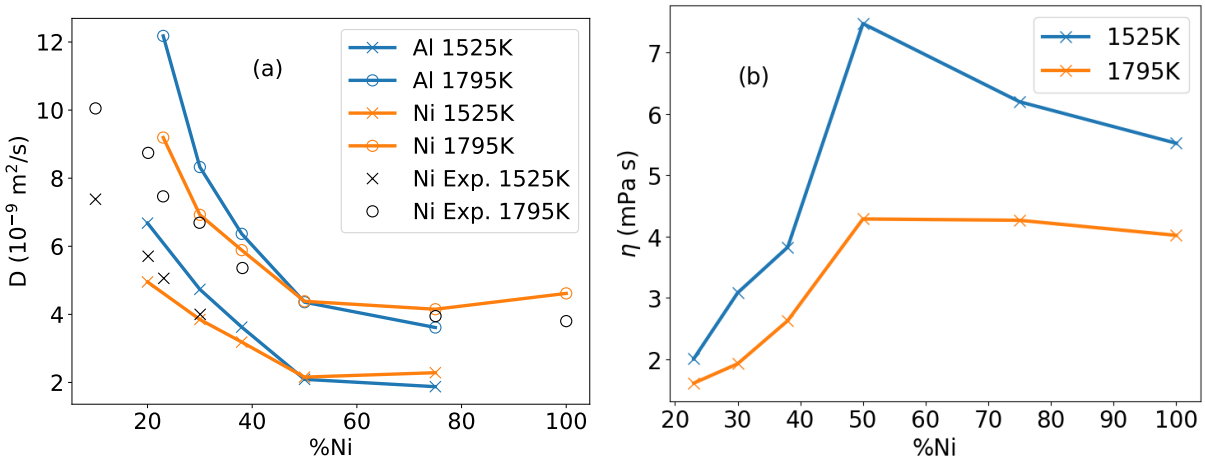

Figure 4.3: Composition dependence of self-diffusions (a), and viscosity (b), along 1795K and 1525K isotherms. Solid lines show results of this article, with dashed lines showing *ab initio* data of [235], and separate symbols showing experimental data from [232].

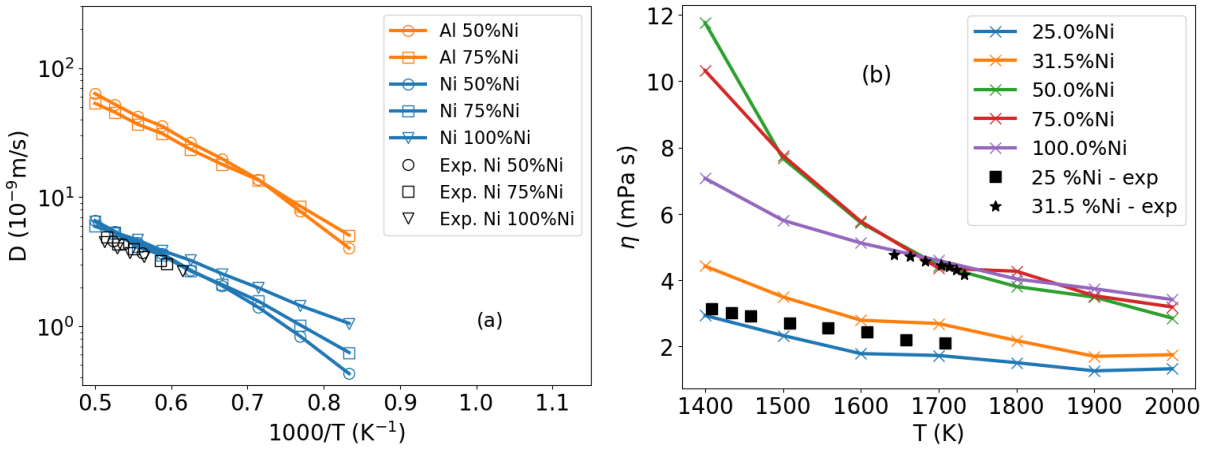

Figure 4.4: Composition evolution of self-diffusion coefficients (a) and viscosity (b) at ambient pressure for two selected temperatures. Separate symbols show experimental diffusion data from [253], and experimental viscosities from [254].

Measurements are then performed over 120 ps in a NVT ensemble. Figure 4.3(a) shows the resulting self-diffusion coefficients, calculated via the Einstein relations. The agreement with the experimental values, shown as separate symbols in the plot, is quite good. A progressive decoupling of the two self-diffusion coefficients sets in as the Ni content decreases above $x_{Ni} = 0.50$. A similar trend was observed in previous AIMD simulations [235]. The same simulation procedure is repeated, in a smaller system of 500 atoms to calculate the viscosity. The latter was calculated *via* the Green-Kubo relation given in Eq. (4.8), for which each simulation is repeated 10 times to obtain a better ensemble averaging in the calculation of the stress-stress autocorrelation function. Figure 4.3(b) displays its composition dependence for the two selected temperatures, which shows consistently an opposite trend to the self-diffusion, as expected. It worth mentioning that by decreasing the temperature, a clear maximum occurs at equiatomic composition. This can be understood by the larger Al-Ni affinity for this composition (see Table SI in the SI file) that promotes the compound formation.

A second set of simulations was performed to study the temperature dependence. After being initialized as for the previous set of simulations, the system was quenched in steps of 100 K. At each temperature point a 20 ps equilibration is performed, followed by a 60 ps measurement. The temperature dependence of the self-diffusion coefficients is shown in Figure 4.4(a). Again, a smaller system is simulated with the same procedure, and used to calculate the viscosities, averaging the stress-stress autocorrelation over 10 realizations before applying the Green-Kubo relations. The viscosities are shown in Figure 4.4(b). The simulations for the viscosities are also performed at two additional compositions, to compare with existing experimental values of [254], showing a good agreement for $Al_{75}Ni_{25}$, but an underestimation at $Al_{68.5}Ni_{31.5}$.

4.3.2 Homogeneous Nucleation

With our ML potential at hand, the early stages of crystal nucleation of Al-Ni alloy at equiatomic composition as well as pure Ni are investigated at an unprecedented accuracy. For this purpose, large scale simulations of 135000 atoms were performed. The system is initiated from an fcc solid solution assigning randomly the type of atoms to reproduce $Al_{50}Ni_{50}$ composition. The system is then melted at 2000K, and equilibrated for 100 ps, before being quenched to 1300K with a cooling rate of 13 K/ps, for an undercooling of $\Delta T \approx 650$ K, compared to the experimental liquidus temperature [255]. The simulation is then continued at this deep undercooling, up to 750 ps. All of these simulations were performed at ambient pressure, within an NPT ensemble to observe the onset of nucleation as can be seen in Figure 4.5, showing a sequence of snapshots of the simulation box at different times.

For all the snapshots, a common neighborhood analysis is performed to classify the local topology of atoms. To reduce the influence of thermal noise, before analyzing the configurations, a minimization is applied on each configuration so as to obtain the inherent structures. After an incubation period during which many pre-critical nuclei form and dissolve back into the liquid state, the majority are identified as bcc clusters (colored blue), with only sporadic clusters identified as fcc (green) and hcp (red). Atoms identified as neither of these structures are not shown. Based on visual inspection, we observe nuclei of sizes up to

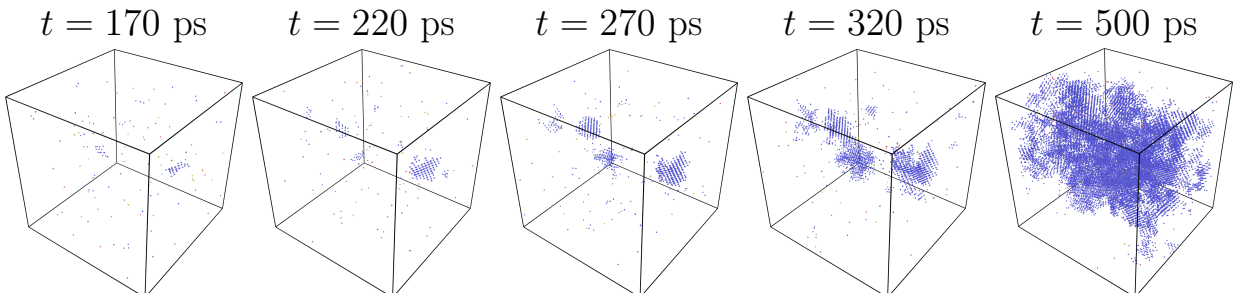

Figure 4.5: Snapshots of the AlNi system during nucleation, at 170, 220, 270, 320, and 500 ps. Particles are colored according to their CNA signature, blue for bcc, red for hcp, green for fcc, and unidentified atoms not shown.

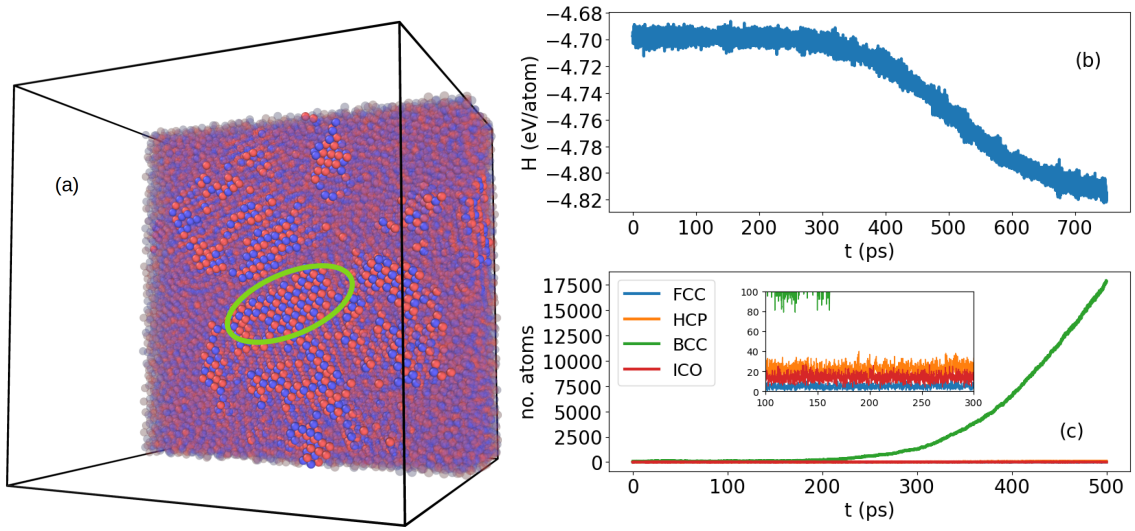

Figure 4.6: Slice of the simulation box for AlNi during nucleation at 500 ps showing the inner structure of the nuclei in the B2 Structure (a). Enthalpy as a function of time during nucleation (b). Abundance of local structures, identified with the CNA, in the AlNi system (c).

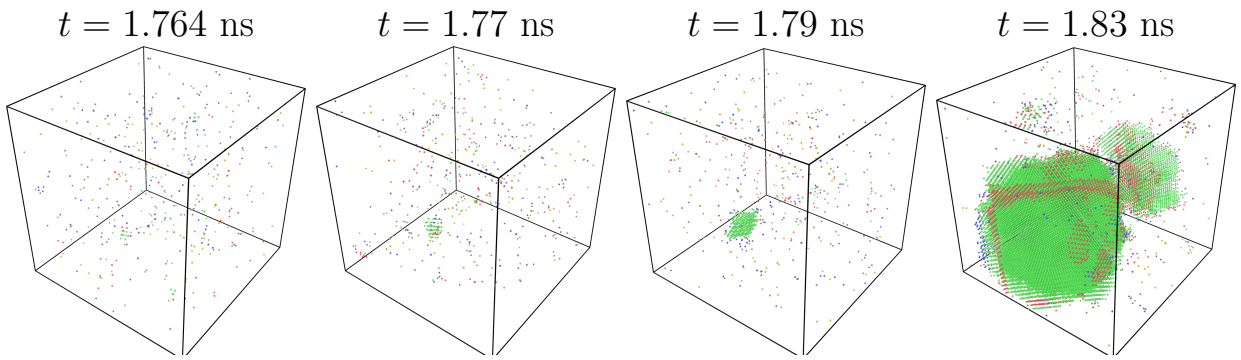

Figure 4.7: Snapshots of the pure Ni system during nucleation, at 1.76, 1.77, 1.79, and 1.83 ns. Particles are colored according to their CNA structure, blue for bcc, red for hcp, green for fcc, and unidentified atoms being translucent.

80 atoms which disappear back into the liquid, putting a lower bound on the critical size n^*. Conversely, the supercritical nuclei appears to arise from clusters to around 120 atoms representing an upper bound to n^*.

Nucleation starts around 170 ps with the growth of a first supercritical nuclei followed by a second one at 220 ps, which is also detected by a sharp drop in the time evolution of enthalpy drawn in Figure 4.6(b). The nuclei display directly a bcc-type from the beginning of the nucleation as quantified in Figure 4.6(c), while the fcc and hcp ordering remains always marginal. This indicates a single-step nucleation process directly from the undercooled liquid into the bcc phase. A closer analysis of the inner part of the nuclei as shown in Figure 4.6(a) indicates clearly a B2 structure. The supercritical nuclei in Figure 4.5 also shows an apparent non-spherical shape, in contrast to what is assumed in classical nucleation theory [3].

Finally, nucleation pathway of the pure Ni system is investigated. The simulation is performed in the NPT ensemble at ambient pressure with a box of the same size as for $Al_{50}Ni_{50}$. Nucleation starts around 1.77 ns after the initial quench from the liquid state at 2000 K to 1200 K, as shown in the snapshots in Figure 4.7. A CNA analysis on the inherent structures of the snapshots seems to indicate a single fcc nucleus, emerging directly from the melt through a single step process. Analysing the bond-order parameters yields an identical picture, as illustrated by the abundance of different structures shown in Figure 4.8. Here we observe an increase in fcc structures, with some hcp inclusions corresponding mainly to stacking faults in the fcc crystalline nucleus, that is deemed to disappear in a later stage of the crystallisation. The single-step process observed here is in contrast to the two-step process reported previously by Orihara *et al* [30] using an embedded atom model, but is analogous to the single-step nucleation we have observed for pure Al [16] using a HDNNP for that element.

The fact that $Al_{50}Ni_{50}$ nucleates in a B2 crystalline structure despite the fact that the pure elements counterparts, namely Al and NI, display fcc crystallisation could be understood from the Al-Ni chemical affinity that preexist in the liquid state and is the strongest at equiatomic composition, as pointed out in the preceding section. Such an affinity favors the

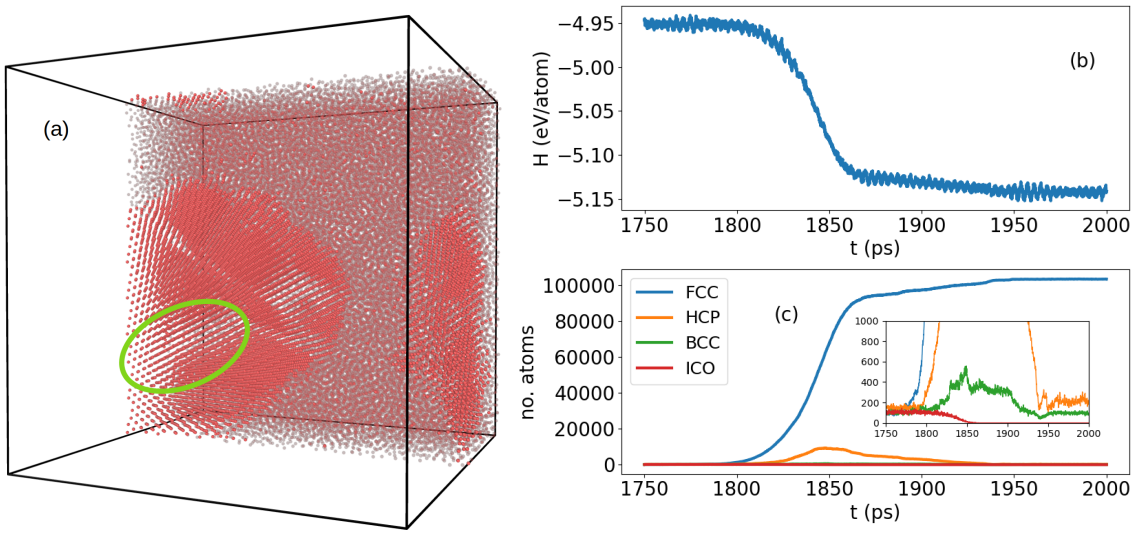

Figure 4.8: Slice of the simulation box during nucleation at 1.83 ns, showing the fcc structure of the nucleus (a). Enthalpy as a function of time during nucleation (b). Abundance of local structures, identified with the CNA, in the pure Ni system (c).

compound formation and the corresponding chemical ordering might take place prior to the topological one when the nucleation takes place as was shown in a previous contribution [98].

4.4 Conclusion

This work was devoted to the study of homogeneous nucleation of a binary Al-Ni alloy by means of molecular dynamics with a HDNNP trained on AIMD trajectories. A reasonable agreement is found between our simulation results and experimental data in the liquid state for the local structural properties as well as for the dynamics through self-diffusivity and viscosity. We take this as an indication of the reliability of the HDNNP in describing kinetic processes. The thus validated potential allowed us to perform large scale simulations at an unprecedented accuracy for this alloy system, close to the *ab initio* one. Simulation of a highly undercooled equimolar melt shows a single step homogeneous nucleation into a B2 phase. The resulting supercritical nuclei have an irregular shape, in disagreement with the assumptions of the classical theory, and the critical size is estimated to be between 80 and 120 atoms. In the pure Ni system with the same HDNNP, a single-step nucleation process towards the fcc phase is also seen, in contrast to previous results using EAM. This, at the very least, highlights the sensitive dependence of the nucleation pathway on the details of the underlying interatomic potential.

While the focus was on just two compositions in the present work, the HDNNP is highly accurate for the melt dynamics also over the entire composition range, and thus the nucleation pathways also of other alloy compositions can be studied. We deem it important that the validation occurs against liquid-state transport coefficients: diffusion and interdiffusion are

important processes in nucleation and growth, while empirical potentials are often gauged against the equilibrium crystal structures.

Acknowledgments

We acknowledge the CINES and IDRIS under Project No. INP2227/72914, as well as CIMENT/GRICAD for computational resources. We acknowledge financial support under the French-German project PRCI ANR-DFG SOLIMAT (ANR-22-CE92-0079-01). This work has been partially supported by MIAI@Grenoble Alpes (ANR-19-P3IA-0003). JS acknowledges funding from the German Academic Exchange Service through DLR-DAAD fellowship grant number 509. We thank Fan Yang and Dirk Holland-Moritz for fruitful discussion on the experimental total structure factors.

4.5 Al-Ni Interdiffusion

Besides the study of nucleation in Al-Ni, simulations were also performed to study the interdiffusion in this system. Originally this was intended to be included in the paper presented in Chapter 4, but ultimately it was decided to focus that work on nucleation itself, with the interdiffusion warranting a more extensive study in the future. Still, some interesting preliminary results were obtained, and are thus presented here.

From the same simulations used to extract the self-diffusion, shown in Figure 4.3, we calculate the interdiffusion via the Green-Kubo relation (2.19). Figure 4.9 shows the isothermal composition dependence of the Onsager factor, thermodynamic factor, and the resulting interdiffusion, calculated from the simulations. These plots likewise show the *ab initio* results of [235], calculated with LDA. There is a fair agreement in the Onsager factor, with a similar trend with composition. This is consistent with the trend of the self-diffusion constants, and in contrast to EAM results reported in [233]. However, a significant difference to the previous *ab initio* results is observed in the thermodynamic factor, which is almost a factor 2 higher in the present simulations.

As the long-wavelength limit of S_{CC} enters through the thermodynamic factor, the process should be discussed, by which this quantity is extrapolated. To obtain this low-q limit, a second degree polynomial is fitted to the low-q points of the structure factor. Since the structure factor is symmetric in q, no linear term is included in the fit. A simple sensitivity analysis is performed, changing the highest and lowest q used in the fit, as well as the impact of fitting a higher order polynomial. Ultimately, a fit was chosen such that it was stable relative to changes in the number of datapoints fitted to. It should be noted that the change in low-q limit that would be necessary to lower Φ by a factor 2 is very clearly inconsistent with our calculated $S_{CC}(q)$. The thermodynamic factor calculated from our GGA AIMD simulations, shown also in Figure 4.9(b), are likewise much greater than the previous LDA results, while being in good agreement with the NNP results, indicating that this is a result of differences in the underlying density functional. As a final test we also calculate the

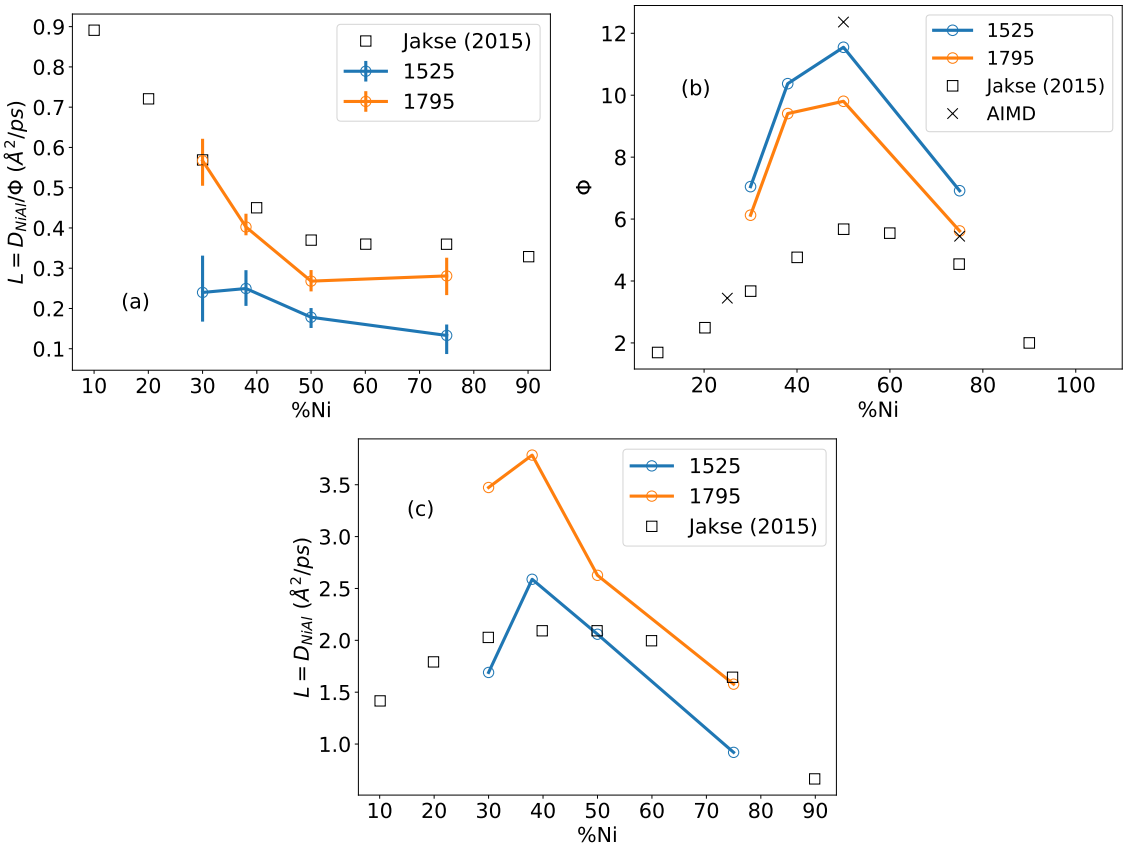

Figure 4.9: Composition dependence of the Onsager factor (a), thermodynamic factor (b), and interdiffusion (c), along ambient pressure isotherms. Solid lines show NNP results, with black symbols showing the LDA *ab initio* results of [235] for 1795K. Black crosses show results from our *ab initio*.

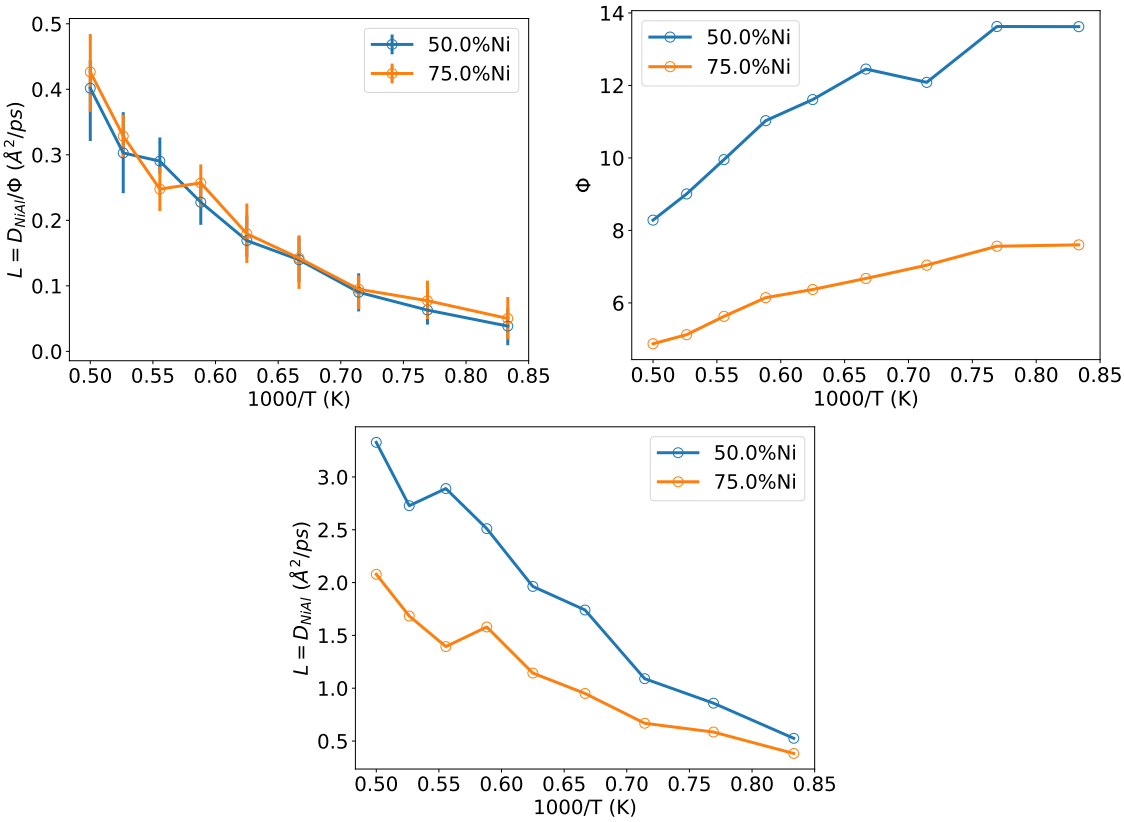

Figure 4.10: Ambient pressure temperature dependence of the Onsager factor (a), thermodynamic factor (b), and interdiffusion (c), at 50 and 75%$_{Ni}$.

thermodynamic factor via the Warren-Cowley parameters, following [235]. This indirect calculation provides significantly lower values for Φ in our case, much closer to the LDA results. Nevertheless, this is an approximate expression, which is not guaranteed to hold for the GGA functional in this case, although it should be mention that it has been shown to hold well for LDA [235]. As a result of the large differences in the thermodynamic factor, the interdiffusion deviates from the previous *ab initio* results using LDA.

The temperature dependence of the interdiffusion is shown in Figure 4.10, along with the Onsager and thermodynamic factors. As before, there is a fair bit of noise in the calculated values, arising due to the collective nature of the interdiffusion, and the difficulty of extrapolating to the low-q limit of $S_{CC}(q)$. Still, some observations can be made. The measured Onsager factors for the two compositions are essentially identical. This is reminiscent of the self-diffusion constant, where there is little to no composition dependence for Ni-rich compositions. In the thermodynamic factor there is, on the other hand, a noticeable difference in the thermodynamic factor between the two compositions, in turn giving rise to a larger interdiffusion in the equimolar system, with the difference decreasing with increasing undercooling.

Chapter 5

Adaptive Group Lasso for High-Dimensional Neural Network Potentials

The following chapter is based on a paper published as part of this doctoral work [18]. An early iteration of this work was also presented at the 2022 *ML for the Physical Sciences* workshop at NeurIPS [256]. The aim of this paper was to introduce the AGL method, described in Section 2.4.2 to MLIPs. It represents, to our knowledge, the first published effort to perform embedded feature selection in the context of HDNNPs. The work is my own, with contribution from my supervisors, as coauthors.

Feature Selection for High-Dimensional Neural Network Potentials with the Adaptive Group Lasso

Johannes Sandberg[123], Thomas Voigtmann[23], Emilie Devijver[4], Noel Jakse[1]

[1]Université Grenoble Alpes, CNRS, Grenoble INP, SIMaP, F-38000 Grenoble, France
[2]Institut für Materialphysik im Weltraum, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 51170
[3]Department of Physics, Heinrich-Heine-Universität Düsseldorf, Universitätsstraße 1, 40225 Düsseldorf, Germany
[4]Université Grenoble Alpes, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France
DOI: 10.1088/2632-2153/ad450e

Abstract

Neural network potentials are a powerful tool for atomistic simulations, allowing to accurately reproduce *ab initio* potential energy surfaces with computational performance approaching classical force fields. A central component of such potentials is the transformation of atomic positions into a set of atomic features in a most efficient and informative way. In this work, a feature selection method is introduced for high dimensional neural network potentials, based on the Adaptive Group Lasso (AGL) approach. It is shown that the use of an embedded method, taking into account the interplay between features and their action in the estimator, is necessary to optimize the number of features. The method's efficiency is tested on three different monoatomic systems, including Lennard-Jones as a simple test case, Aluminium as a system characterized by predominantly radial interactions, and Boron as representative of a system with strongly directional components in the interactions. The AGL is compared with unsupervised filter methods and found to perform consistently better in reducing the number of features needed to reproduce the reference simulation data at a similar level of accuracy as the starting feature set. In particular, our results show the importance of taking into account model predictions in feature selection for interatomic potentials.

5.1 Introduction

During the last decade, Machine Learning Interaction Potentials (MLIPs) have become a commonplace method for Molecular Dynamics simulations in material science and chemistry [126, 257], following a broader trend of data-driven approaches in material science [128, 258]. *Ab initio* simulations, using for instance Density Functional Theory (DFT) force calculations [4], have good accuracy and broad applicability, but suffer from poor scalability. Being trained to reproduce *ab initio* forces and energies, MLIPs were shown to combine many of the benefits of *ab initio* with the scalability and performance of classical force fields [118, 120, 259], thereby opening up new avenues of research into nucleation [100, 11, 260], structure-property relationship in alloys [7, 261], and amorphous solids [6, 262] to name a few.

A wide variety of MLIPs have been proposed, often relying on a local decomposition of the high dimensional potential energy into a sum of local contributions. Methods such as the spectral neighbor analysis potential [35] rely on a linear regression over a set of nonlinear descriptors of the local atomic environment. Nonlinear dependencies can be added by the use of kernel regression, as in the Gaussian approximation potential [10, 226], or by using Neural Networks (NN) as in the deep potential framework [39] and the high dimensional neural network potential [9]. More recently, methods based on graph neural networks have seen a lot of traction [228], including methods based on equivariant transformations [41]. Attempts have also been made to go beyond local interaction in what has been referred to as the third and fourth generations of machine learned potentials [34]. For most MLIPs, it is necessary to transform the bare atomic coordinates into a set of atomic descriptors [126] describing the local environment of each atom. The purpose of this transformation is to enable a local description, ensure invariance to local symmetry transformations, and to guarantee that the input to the Machine Learning (ML) model is of constant dimension, even as the number of atomic neighbors can change during a simulation.

Computing the descriptors is often the main time consuming part of applying a NN Potential (NNP), compared to the NN evaluation and backpropagation. As such, care is needed when designing the set of atomic features, and in particular one has to weight the need for a detailed description of the atomic environment against the additional computational cost of having a large feature space. There is also some evidence that larger feature sets can negatively impact generalization [76]. Feature selection [263] allows for a data driven way of designing such feature sets by identifying those features out of a larger collection that are the most relevant, and discarding redundant ones. The simplest approach to feature selection are filter methods. Such methods select features by looking only at the dataset, before training takes place, and are as such model independent. Imbalzano *et al.* [14] proposed three such methods for use with MLIPs. Two of these are based on minimizing the Pearson Correlation (PC), and maximizing the Euclidean distance, respectively between the selected features. The third one is based on the CUR decomposition [79], which can be regarded as an analogue of the singular value decomposition, constructing a low-dimensional representation of the data matrix but using only rows (columns) of the original matrix chosen such that the reconstruction error is minimized.

Filter methods can be contrasted with embedded methods, wherein the feature selection process is integrated into the training of a specific model. Such an embedded approach allows for explicitly taking into account model predictions, as well as interaction between different features [89]. A famous embedded method is the lasso [13], based on regularization using the $L1$ norm of the input parameters of a linear model. Lasso has previously been used to construct MLIPs for a variety of elements based on ridge regression [12, 264], and has been applied beyond MLIPs to predict directly material properties starting from large sets of material descriptors [265]. The latter led to the development of the SISSO method [266] in the framework of materials discovery, where features are subjected to an initial screening based on their correlation to the target property, before being further selected using the lasso, allowing for selection from more than billions of candidate material descriptors. However, as

it induces sparsity at the level of individual parameters, lasso is not applicable as a feature selection method for NNPs.

While much of the focus for feature selection was traditionally on linear regression, likely owing to the nonlinear nature of NNs, recent works tried to extend methods to the nonlinear case. Methods based on the Group Lasso (GL) has been applied to NNs as early as 2017 [83]. It was, however, shown that this direct application of GL to NNs cannot consistently discard truly irrelevant features, a problem that can be avoided by using an adaptive penalty for an Adaptive GL (AGL) approach [86]. Another recent method is LassoNet [89], adding bypass connections from each input variable to the NN output, applying a lasso penalty on the bypass weights and using them to constrain the maximum values of the input weights. This change in architecture, however, deviates from the simple networks used in most common NNP implementations, while also introducing an additional hyperparameter that in principle needs to be tuned. For these reasons the AGL might be more directly suitable for NNPs.

In this article, we introduce an approach of feature selection based on the AGL method applied to High Dimensional NNPs (HDNNPs), with the aim of showing that the use of a method that takes into account the interplay between features in the specific estimator allows for better selection of atomic fingerprints. This type of NNP model is known to work well for many systems, and has been well studied, making it a natural framework for our study. While more recent graph-based models avoid the need for feature selection, such deep models have been shown to suffer from potential stability issues [267, 268]. It should be noted though that methods for inducing sparsity might still have benefits [269] More importantly, message passing poses a problem for scalability of graph-based models, due to difficulties in parallelization [270]. This is especially relevant for situations requiring large scale simulations, in which feature selection is of particular interest. We consider three different systems: *Lennard-Jones* (LJ), serving as a simple and well known generic model whose analytic expression has no explicit angular dependence; *Aluminium* (Al), which serves as a relatively simple sp bonding metal; *Boron* (B), which is known to have a particularly complex structure with a high degree of directional covalent bonding [271, 272], in addition to radial interactions. Notably the B ground state is not fully understood [271], with a crystalline structure dominated by B_{12} icosahedra, with a pronounced short range order in the liquid [55]. Taken together, these three systems provide increasingly complex, and increasingly angularly dependent, interactions. We find that for Al the AGL method is competitive with filter methods. For the other systems it is explicitly shown by example how the filters can fail to select features that are necessary, while they are discovered by our method, illustrating the advantage of an embedded feature selection approach.

The remainder of the article is as follows. Section 5.2 provides background on our datasets, the HDNNP approach, the AGL method, and the computational tools used. Section 5.3 covers the results of training HDNNPs with AGL, comparing to the CUR and PC methods, as well as simulations used to test the effect of the reduced feature sets in production. Finally, section 5.4 provides the main conclusions and outlook of the paper.

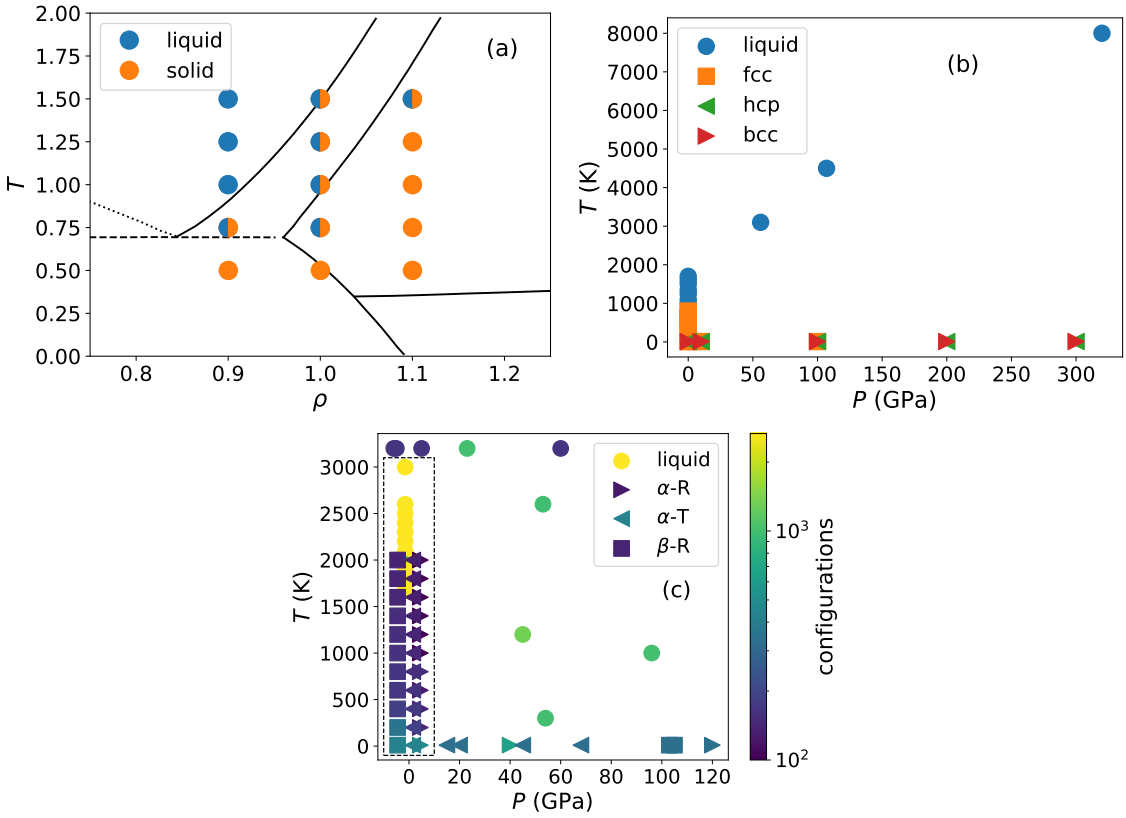

Figure 5.1: Thermodynamic points sampled in the construction of our datasets. Colors, symbols indicate respectively simulations started in different thermodynamic phases. (a) Temperature-density (T-ρ) phase diagram for LJ. For points with both colors, two separate simulation sets were included. (b) Temperature-pressure (T-P) phase diagram for Al. (c) Temperature-pressure (T-P) phase diagram for B. Points on the $P = 0$ line, inside the dashed rectangle, have been shifted horizontally for readability. Here, colors represent the number of states sampled.

5.2 Method

5.2.1 Datasets

A first step of training a HDNNP is to construct a dataset of reference structures. Figure 5.1 illustrates the location of the thermodynamic state points included in datasets used for the training of the three systems.

The dataset for LJ was extracted from a set of LAMMPS [133] simulations of 256 atoms at temperatures ranging from 0.5 to 1.5 (LJ units), and densities 0.9 to 1.1, in both solid (fcc) and liquid configurations. We use the standard LJ pair potential, given for interatomic distance $r < r_c$ by

$$V = 4\epsilon \left(\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right). \tag{5.1}$$

All the simulations are performed with parameters $\sigma = \epsilon = 1$, particle mass $m = 1$, and cutoff radius $r_c = 2.8$. Figure 5.1(a) shows the thermodynamic states included in the dataset. Each thermodynamic state was sampled 1333 times, with an interval of 0.3 time units (300 timesteps), for a total of 28000 configurations. Note that the coexistence lines in figure 5.1, reproduced from [273], are valid in the limit of infinite cutoff, and merely included as visual guide.

In the case of Al, our reference data is the same as in our previous article [100]. This dataset consists of 24300 configurations extracted from DFT-based *Ab Initio* Molecular Dynamics (AIMD) simulations performed in VASP [164] using the LDA functional [166] in an augmented plane wave framework with a cutoff of 241 eV. Configurations in the dataset cover fcc, bcc, and hcp crystalline states, and the liquid, at a variety of temperatures and pressures the details of which we refer to the original article [100]. Figure 5.1(b) shows the thermodynamic points sampled to construct the dataset. Liquid states, and fcc crystals at ambient pressure were sampled 1000 times each. The remaining crystal states were each sampled 100 times.

For B, we extract reference configurations from the AIMD trajectories used in [55], complemented with additional simulations for α-rhombohedral, α-tetragonal, and β-rhombohedral crystals at temperatures ranging from 10K to 2000K in steps of 200K, extracted from the *Materials Project* database [243]. Additional high-pressure simulations were also included, to probe the short-range interaction. Figure 5.1(c) shows the thermodynamic state of each simulation trajectory, with the number of configurations drawn from it. Each trajectory was sampled with an interval of 45 fs (30 timesteps), for a total of 45000 configurations. These simulations were performed using the Perdew Wang GGA functional [242] with a 300 eV augmented plane wave cutoff sampling only the Γ point, for consistency with [55].

In all cases, the simulations were performed in an NVT ensemble with a Nosé thermostat controlling the temperature, and pressure is controlled by fixing the volume of the simulation box. To ensure sampling of equilibrium states, each trajectory was preceded by an equilibration period ranging from 500 time units for LJ, and 100 to 200 ps for Al and B.

5.2.2 HDNNPs

The interaction between atoms in a material is frequently described in terms of a potential, depending in principle on the positions of all atoms in the many-particle system. This interaction is often short-sighted, and can be treated as sum of atomic contributions depending only on the local structure of each atom, within an appropriate cutoff radius r_c

$$E_{\text{total}} = \sum_{i=1}^{N_{\text{atoms}}} E_i. \tag{5.2}$$

A HDNNP [9, 125] is constructed from this decomposition by assigning a NNP to each species of atom, mapping between the local environment and the corresponding atomic energy contribution E_i. The input to the HDNNP are the atomic positions, which are transformed into a fingerprint vector for each atom, serving as input to the atomic NNP. Training then

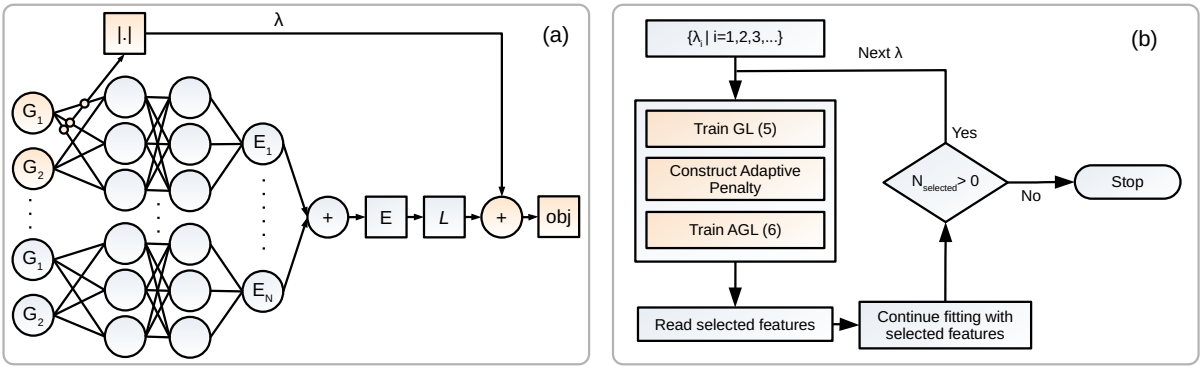

Figure 5.2: Schematic of feature selection for HDNNPs with AGL. (a) Illustration of a NNP, with GL penalty added to the first feature, used to construct the objective function. (b) Flowchart of the feature selection process.

consists of fitting the full HDNNP to the total potential energy obtained from *ab initio*. Often the derivative of the HDNNP is fitted to the *ab initio* forces as well, but for simplicity in focusing on the feature selection and following our previous work [100], we train only to the energies in this work.

There are many options in choosing atomic descriptors, with [126] offering a brief overview of some common types. In this work, we use the Behler-Parrinello symmetry functions (SF) [36], which is the conventional choice for HDNNPs. These consist of the radial G^2 and angular G^5 SFs defined by

$$G_i^2 = \sum_j e^{-\eta(R_{ij}-R_s)^2} f_c(R_{ij}) \tag{5.3}$$

$$G_i^5 = 2^{1-\zeta} \sum_{j,k} (1 + \Lambda \cos\theta_{ijk})^\zeta e^{-\eta(R_{ij}^2+R_{ik}^2+R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \ . \tag{5.4}$$

Here, R_{ij} is the distance between atoms i and j, θ_{ijk} is the angle between atoms j and k with respect to atom i, and $f_c(R_{ij})$ is defined as 0 for $R_{ij} > r_c$ and for $R_{ij} < r_c$ as a polynomial going smoothly to 0 at the neighborhood cutoff $R_{ij} = r_c$. The parameters η, ζ, Λ, and R_s allow for defining a set of features by assigning these parameters different values. Here the initial featuresets are generated by selecting parameter values on a grid, akin to the procedures described in [76, 14], with the aim of being sensitive to a range of interatomic radii and angles. The exact SF parameter values used can be found in the supplementary material [274].

5.2.3 Feature Selection

The main hindrance in applying feature selection methods based on the L1 norm to NNs is the fact that the L1 norm acts on individual weights. In a NN, several weights are associated with each feature, and so to do feature selection we need to penalize these weights as a group. The GL replaces the L1 norm with Euclidean norms over groups of parameters. As the

Euclidean norm of a parameter group vanishes if and only if all those parameters vanish, this allows for selecting or discarding groups of parameters simultaneously. To select features for NNs using GL we take the groups to be the input weights of feature i, $\omega^0_{i,[:]}$, with the corresponding Euclidean norm $|\omega^0_{i,[:]}|$. During training we then optimize the objective function

$$\text{obj}(W) = L(W) + \frac{\lambda}{N}\sum_{i=1}^{N}|w^0_{i,[:]}|, \tag{5.5}$$

with L being some loss function, in our case the Mean Square Error (MSE), W being the weights of the neural network, N being the number of inputs, and λ being a regularization parameter used to tune the relative strength of the feature selection. A challenge in performing this optimization is the fact that the second term in (5.5), called the penalty, is non-smooth. In [275] a smoothed approximation of (5.5) is used, but here the non-smooth optimization problem is instead solved directly using a proximal gradient descent algorithm, following [276]. Figure 5.2(a) illustrates the GL penalty acting on one of the input features to a schematic NNP.

The adaptive version of the algorithm [86] uses a separate regularization parameter for each individual weight group. This adapted penalty is constructed from an initial training run using the non-adaptive penalty. The training is then redone with the new penalty, optimizing

$$\text{obj}(W) = L(W) + \frac{\lambda}{N}\sum_{i=1}^{N}\frac{|w^0_{i,[:]}|}{|\hat{w}^0_{i,[:]}|} \tag{5.6}$$

with $\hat{w}^0_{i,[:]}$ being the values of $w^0_{i,[:]}$ obtained during the initial training run with the non-adaptive penalty. Depending on the value of λ, some features will have their weights go to zero during training, and can thus be discarded. This allows for selecting features by performing a search over this single parameter, following the workflow illustrated in figure 5.2(b).

5.2.4 Computational Tools

Training of HDNNPs were performed using our own code, with the SF calculations being performed using N2P2 [91]. For the CUR selection we use the code implementation from [277]. Simulations with the trained potentials were performed in LAMMPS [133] using the ml-hdnnp plugin provided by N2P2. As mentioned in section 5.2.1 we use VASP [164] for reference *ab initio* calculations. OVITO [178] was used for some post-processing, calculating the Radial Distribution Functions (RDFs).

5.3 Results and Discussion

5.3.1 Lennard Jones System

As a first test of our method we apply the AGL to the LJ system, where the exact interactions are perfectly known. In particular, they are perfectly spherically-symmetric pair interactions,

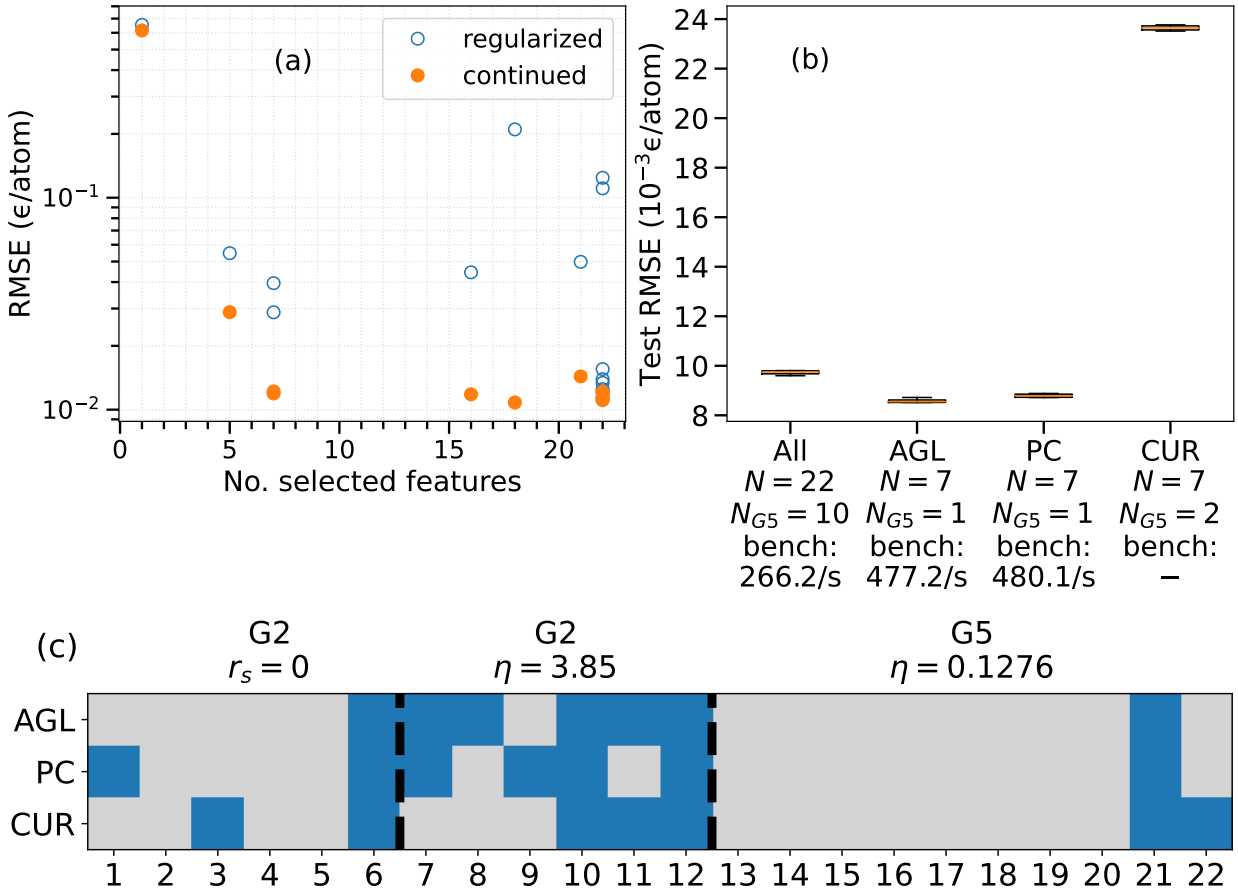

Figure 5.3: Selection process for LJ. (a) RMSE of models trained with different values of the regularization parameter λ, plotted against the number of selected features. Blue circles show the error at the end of training with the AGL penalty. Orange dots show the error after continuing training without penalty, with only selected features. (b) Box plot of test errors for different feature sets, with total number of features (N), number of angular features (N_{G^5}), and timesteps per second in a benchmark simulation. (c) Matrix plot of selected features. Rows correspond to different methods, with discarded features grayed out, and selected ones in blue. The features are grouped into centered G^2, shifted G^2, and G^5.

so that one might expect a feature-selection method to successfully discard features pertaining to angular directionality. The initial feature set contains 12 radial SFs, 6 of which are centered on $r_{ij} = 0$ with varying widths η, with the remaining 6 being centered on regularly spaced r_s having constant width. In addition to the radial SFs, 10 angular ones are included, using the same wide centered radial component, with varying angular width ζ in pairs of $+1$ and -1 for the Λ parameter. All the SFs use the same cutoff radius, set to the cutoff used in the reference LJ potential, $r_c = 2.8$. The NNP consists of two hidden layers with 10 neurons each.

For the feature selection, we apply the AGL method described in section 5.2.3 by defining a sequence of regularization parameters λ, training an initial model with the non-adaptive GL (5.5). This is then used to construct and retrain the model using the adaptive penalty given by (5.6). Each of these models has its weights randomly chosen at the beginning of the training, referred to as cold initialization, and is trained using the ADAMW optimizer [74] with learning rate set using a learning rate finder [68], and a small weight decay parameter $\gamma = 10^{-6}$ applied only to the internal weights so as to not interfere with the feature selection. The batch size was fixed at 256 configurations, and standard input normalization was used, shifting and scaling each feature to have mean 0 and standard deviation 1 over the training dataset. We let aside 10% of the training data as a hold-out validation set to monitor the model performance during training for early stopping. Crucially, for the sake of early stopping we do not monitor just the loss function, but the relevant objective function given by (5.5) or (5.6), ending training if it has not improved for 10 epochs by more than 10^{-7}. In the absence of early stopping, the training was capped at 1000 epochs for the non-adaptive part, and 10000 during the adaptive part.

During training with the adaptive penalty, the weights corresponding to some of the inputs will vanish. Following the training for each λ we identify these weights and freeze them before continuing training without the penalty. This is to avoid the bias that is otherwise known to occur for L1 regularized models [88]. Figure 5.3(a) shows the validation Root Mean Square Error (RMSE) for each model along this path, plotted against the number of selected features, both at the end of training with AGL (blue circles) and after continuing without the penalty (orange dots). Note that the regularization introduces a noticeable overestimation of the error associated to the selected feature sets, and so continuing the training is necessary to make an informed decision on which set of selected features to choose. In figure 5.3(a) one can observe an initial plateau in the lowest error reached during continued training when going from 22 selected features down to 7. We interpret this as the regime where the AGL method discards unnecessary features that lead to little decrease in performance. Going below 7 features, the model suffers a large increase in error, as the result of having to discard more and more important features.

Based on figure 5.3(a), we select the model with 7 features, of which 1 is of the angular type given by (5.4). The selected feature set is tested by training over four different random initializations, with the same training dataset, to ensure the features are not suited for just one part of the weight space. Unlike the models on the regularization path, in order to speed up convergence, these models were trained using the *cosine annealing with warm restarts*

learning rate schedule [69]. With this schedule the learning rate is annealed with a cosine from a large initial value to a small value (10^{-8}) over a number of weight updates, before resetting the learning rate to its initial value and repeating the process. Here the initial period of the scheduler is set to coincide with one epoch, and to double after each reset, ending training after a total of 12 resets (8190 epochs). We likewise test the starting feature set, as well as 7 features selected with the PC and CUR methods of [14]. The resulting test errors, evaluated on a held out test set, are presented in figure 5.3(b), together with the total number of features N and the number of angular features N_{G^5}. Additionally, we perform a benchmark simulation with each potential, consisting of 256 atoms simulated in an NVT ensemble for 6000 timesteps. These simulations ran on 48 2.7 GHz Intel Skylake cpu cores, and the average number of simulated timesteps per second of wall time is recorded and shown in figure 5.3(b). We note that the models trained on the features selected with CUR did not allow for a successful benchmark simulation on account of their large error, which will be discussed in more detail below.

It can be seen that there is a strong preference for radial SFs, as one would expect considering the lack of angular dependence in the reference LJ potential. Despite this, a single angular feature was selected by both the AGL and the PC filter. This is not unreasonable, since we train the LJ system with high-density configurations as reference data, where steric repulsion leads to the emergence of certain short-ranged angular order. The features selected with CUR greatly underperform those selected with the other methods, but we note that CUR performs much better for a larger number of features [14]. CUR selected two angular features, which could allow for a better reconstruction of the atomic environment overall by taking better into account the angles, but at the cost of a reduced radial resolution. As the CUR approach acts on the descriptors alone, it is largely incapable of knowing the lack of angular dependence of the energy in the ground truth. It should however be mentioned that this information could still be, to some extent, indirectly available through what configurations appear in the sampled MD trajectory used to construct the dataset.

To better illustrate the differences between the feature selection methods, we show in figure 5.3(c) a matrix representing the features selected by each method. The G^2 SFs selected by AGL and CUR are also plotted in figure 5.4, along with the Radial Distribution Function (RDF) extracted from one of the reference simulations. Of note is that CUR discarded three consecutive shifted radial SFs in a regime where the other methods kept at least one. This raises the question of whether adding one of these SFs to the CUR features would recover a good performance. In order to test this, we create two new sets by adding to the CUR features one of the shifted radial SFs selected by AGL but discarded by CUR, marked 7 and 8 in figure 5.4. Adding feature number 8 reduced the test RMSE to $18.4 \times 10^{-3}\epsilon$/atom, which is a modest improvement, but still nowhere near the performance of the other sets. Instead, adding feature number 7 lowers the test RMSE to $9.40 \times 10^{-3}\epsilon$/atom, a clear indication that this is indeed a vitally important feature for this system that the CUR method failed to detect. With this feature added, the resulting model also allowed for stable simulations to be performed.

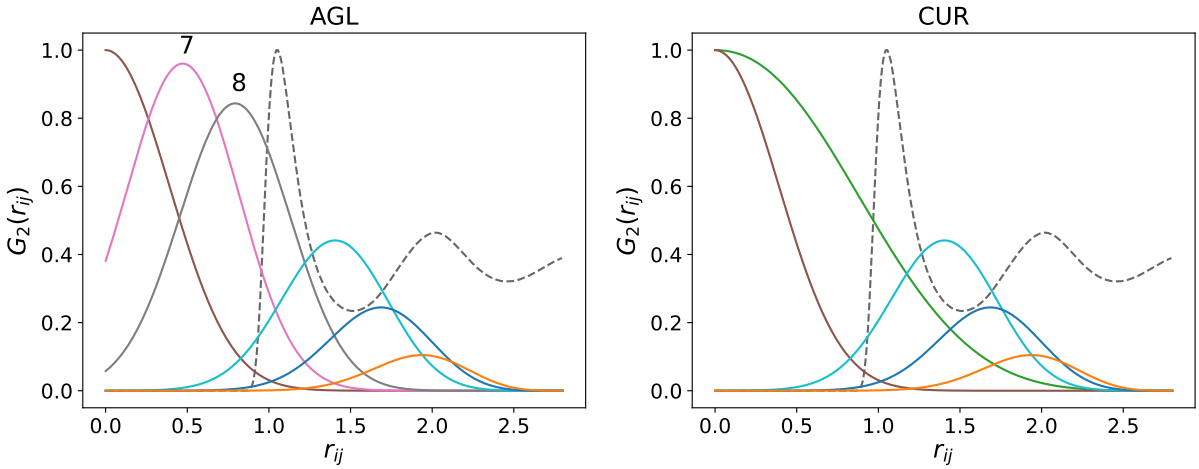

Figure 5.4: Radial symmetry functions (G_2) for LJ, visualized for a single atom pair, selected by AGL (left), and CUR (right). RDF of reference system included for comparison (dashed line). Features number 7 and 8 from figure 5.3(c) are marked.

5.3.2 Aluminium

To test the method in a more practical setting, we turn to the case of Al. The SF parameters and network architecture is chosen as in [100]. We proceed as for LJ, training a sequence of models on increasing values of λ, using cold initialization, continuing the training after selecting the features. The resulting validation errors are plotted against the number of selected features in figure 5.5(a). We find 10 features to be a good compromise between few features and low error. The set is again evaluated by training a set of four models on the selected features, with different initialization, likewise for the staring features and features selected with CUR and PC. The test errors are shown in figure 5.5(b), along with the number of angular features selected, and number of timesteps per second in a benchmark simulation identical to the one for LJ. We see a significant increase in computational speed for the feature-selected potential, at a relatively small increase in error. For this system, CUR and PC seem to perform equivalently. In particular the CUR features perform much better than in the LJ case, presumably because it is asked to select more features and so the method is not forced to compromise on the radial resolution. The features selected with AGL, on average, outperform those chosen by the filters, although there is not a large difference in this case, especially considering the deviations.

A point should be made regarding the nonlinear scaling of the benchmark performance in figure 5.5, with respect to the number of features. This is a direct consequence of the angular G^5 features involving a double sum over neighbor atoms, as opposed to the single sum of the radial G^2 features. In addition, depending on the SF parameters, some factors appear in the calculation of several different features [91], allowing for optimizations that further complicate the scaling.

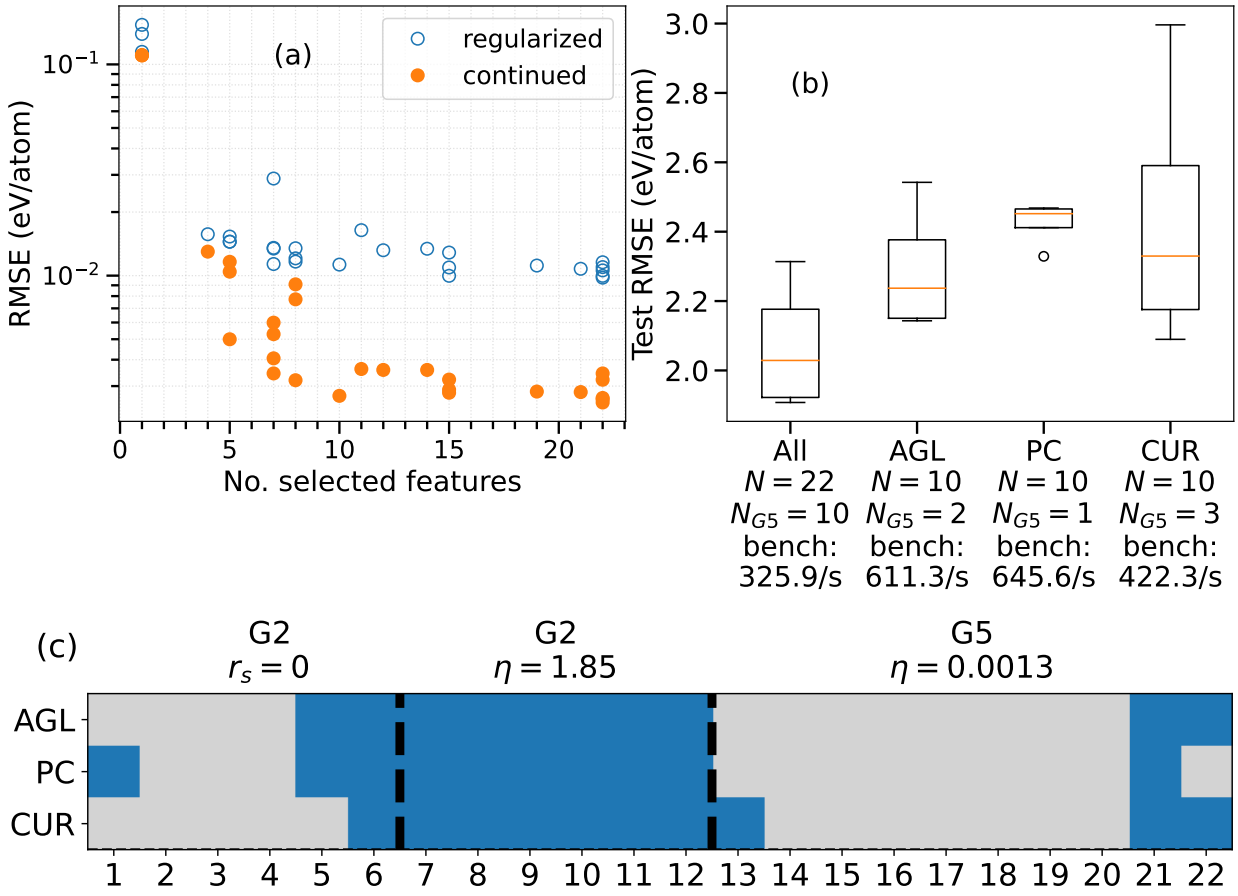

Figure 5.5: Selection process for Al. (a) RMSE of models trained with different values of the regularization parameter λ, plotted against the number of selected features. Blue circles show the error at the end of training with the AGL penalty. Orange dots show the error after continuing training without penalty, with only selected features. (b) Box plot of test errors for different feature sets, with total number of features (N), number of angular features (N_{G^5}), and timesteps per second in a benchmark simulation. (c) Matrix plot of selected features. Rows correspond to different methods, with discarded features grayed out, and selected ones in blue. The features are grouped into centered G^2, shifted G^2, and G^5.

The feature sets are visualized in figure 5.5(c). We observe, somewhat different from the LJ case, a great overlap between the methods, and presumably the one or two features that differ between each set are not enough to cause a significant difference in the test error. In particular we notice that each model selected each shifted radial SF. Feature number 6 in figure 5.5(c), being also selected by each model, is identical to the shifted ones, but centered on $r_{ij} = 0$. Taken together these features can be argued to cover the entire range of interatomic distances up to the cutoff radius, allowing for a rough representation of the RDF. This preference for shifted radial SFs has also been indicated elsewhere in the literature [76].

Like in the case of LJ, there is here a preference towards radial features, with only two angular ones being chosen. We suggest a physical explanation for this preference for radial features, noting the tendency of Al to adopt a close-packed short range order and to maximize the number of nearest neighbors, due to the weakly directional sp bonding type electronic structure.

While the 10 features selected are a sensible choice, based on the training errors reported in figure 5.5(a), the threshold is not rigorous. From the RMSE values obtained, a selection of 8 or even only 7 features could also be argued for. In going to 7 features, a noticeable increase in the test error was observed, providing only a modest improvement in benchmark performance primarily due to an additional discarded angular SF. Simultaneously, the CUR features show a significant reduction in performance, reminiscent of what was observed for LJ. In the present case, this was presumably due to the deselection of both features number 6 and 7 by CUR. Figures for these featuresets can be found in the supplementary material.

5.3.3 Boron

We turn now to boron as a stringent test system. Due to the complicated structure of boron, induced by strong covalent directional bonding [271, 272, 55], we expect this to be a significantly more difficult task, and to require a more complex set of features compared to Al and LJ. For our initial set of descriptors we use a set of 12 radial SFs, and 48 angular SFs, with a cutoff of 5.3 Å corresponding roughly to the outer edge of the third neighbor shell. This relatively wide cutoff was chosen in order to hopefully be able to more adequately take into account the medium-range structure known to appear in boron, primarily the open icosahedra and the bonds between them [55]. Furthermore, to allow for a potentially more complex mapping we use a larger network than for LJ and Al, with two layers of 25 hidden nodes each, providing a slight improvement in error compared to smaller network sizes.

As for the previous systems, figure 5.6(a) shows the validation RMSE as a function of the selected features. In this case the best-performing model, apart from the one with the full set of features, is for 16 features. We select these 16 features, and again train a set of four models to test, with the results shown in figure 5.6(b). In this case we not only selected a larger number of features, but the majority of features selected were of the angular type. Unlike in the previous cases, we also observe an inability of the filter methods to adequately select features for this system, with a significant increase in error for the sets selected with PC and CUR. In fact, we were unable to perform even a benchmark simulation using the models trained on the PC set, with the simulations becoming unstable. For the AGL set

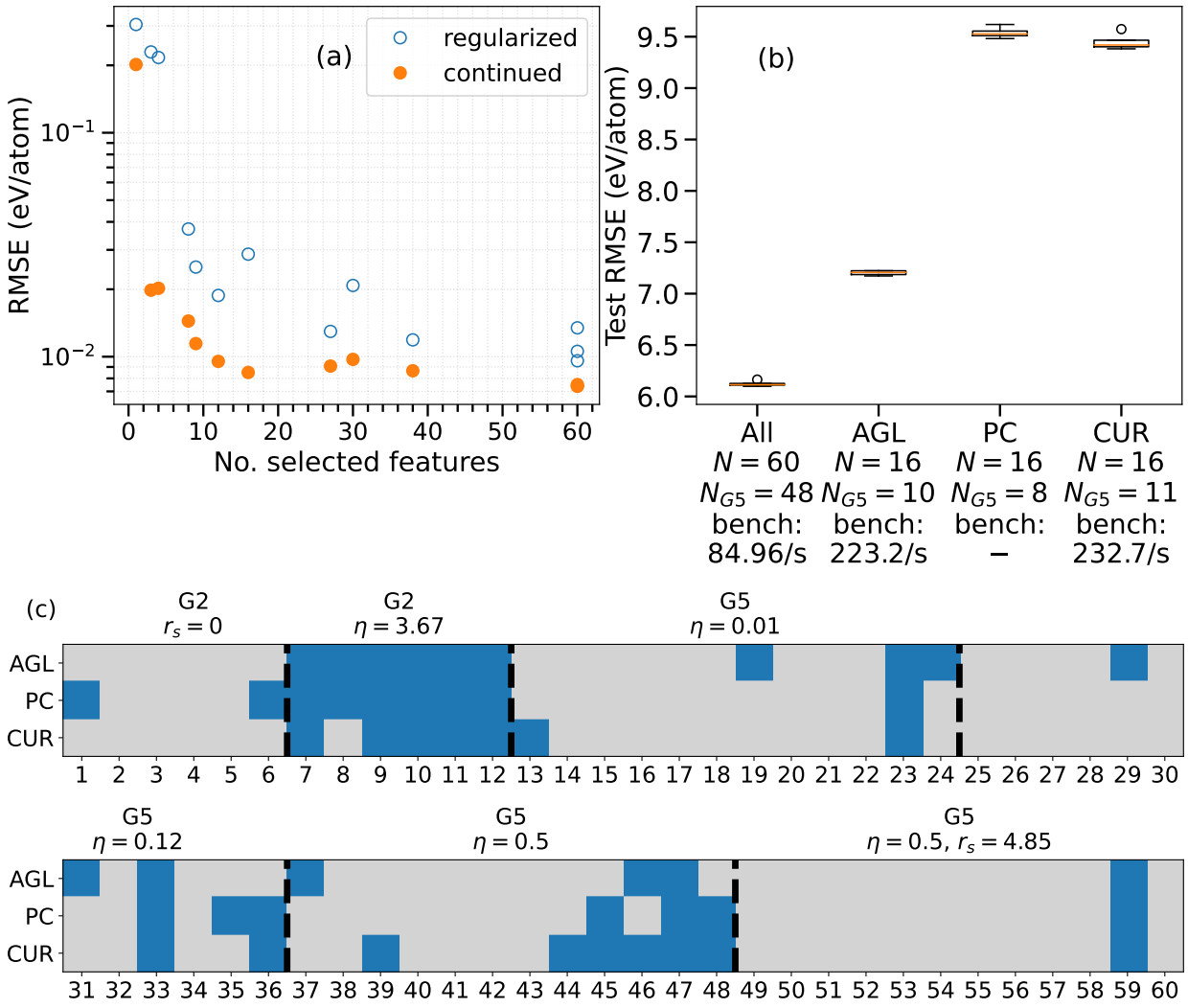

Figure 5.6: Selection process for B. (a) RMSE of models trained with different values of the regularization parameter λ, plotted against the number of selected features. Blue circles show the error at the end of training with the AGL penalty. Orange dots show the error after continuing training without penalty, with only selected features. (b) Box plot of test errors for different feature sets, with total number of features (N), number of angular features (N_{G^5}), and timesteps per second in a benchmark simulation. (c) Matrix plot of selected features. Rows correspond to different methods, with discarded features grayed out, and selected ones in blue.

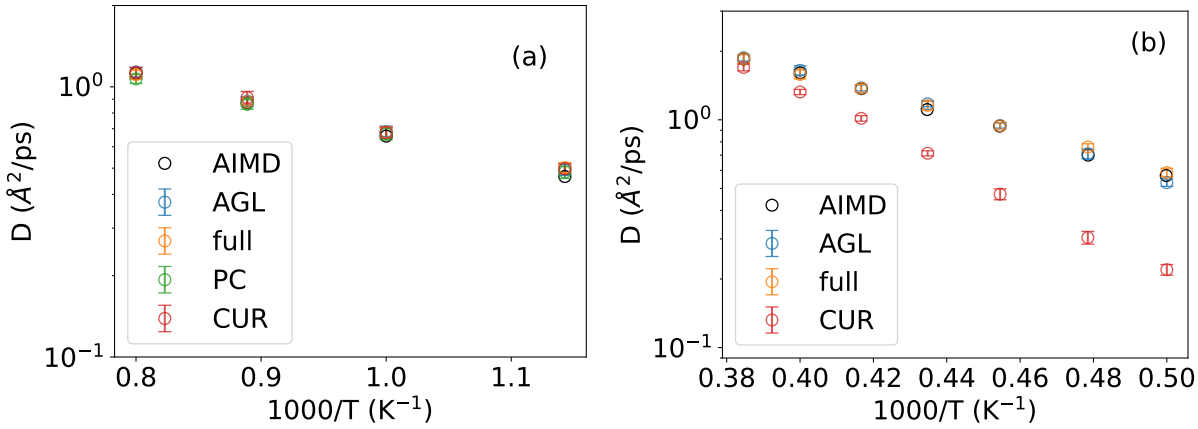

Figure 5.7: Arrhenius curves for the diffusion constant against temperature, with different feature sets, for (a) Aluminium and (b) Boron. Reference *ab initio* diffusion data from [101, 55]

there is a noticeable increase in the error compared to the full set of features, but this comes with a significant improvement in the computational performance of the potential. It should be noted that the error of even the baseline HDNNP trained with all the available features is noticeable, comparing to, for instance, the errors seen in figure 5.5 for Al. This very likely hints at the BP SFs not being well suited for the B system, a point which has been made previously in the literature [278], highlighting the need to explore different types of fingerprints for HDNNPs. Nevertheless, the success of the AGL method in even this setting shows promise that with more well suited features the method would still be usable.

5.3.4 Validation of the MLIP models

While looking at the RMSE of the models on a held-out set of test configuration is useful, the true test of the quality of a MLIP is in simulations and the accurate prediction of physical quantities. For each set of features we pick out the model with the best test error and perform an NVT simulation, aiming to obtain the diffusion constant for comparison to *ab initio*. We specifically focus on the real systems, Al and B, leaving the LJ case for the supplementary material. Each simulation uses a box of 256 atoms, in order to match the finite size effect in the reference systems. For Al we prepare the system in an fcc crystal configuration, and melt it at 1250 K. The resulting liquid is then repeatedly quenched in steps of 125 K, followed by 30 ps of equilibration, down to 875 K, relatively deep in the undercooled regime. At each temperature a measurement is then performed over 1 ns of simulation at constant temperature. For B the same procedure is followed, but starting from a 2600 K liquid configuration drawn from *ab initio*, and quenching in steps of 100 K down to 2000 K. In both cases, the diffusion is calculated from the mean square displacement, v*ia* the Einstein relations, and averaged over a set of 10 independent simulation runs. Figure 5.7 show the diffusion as a function of inverse temperature, for the different feature sets. In the case of Al (a), we see a good agreement across all temperatures, with none of the feature sets being obviously worse. This

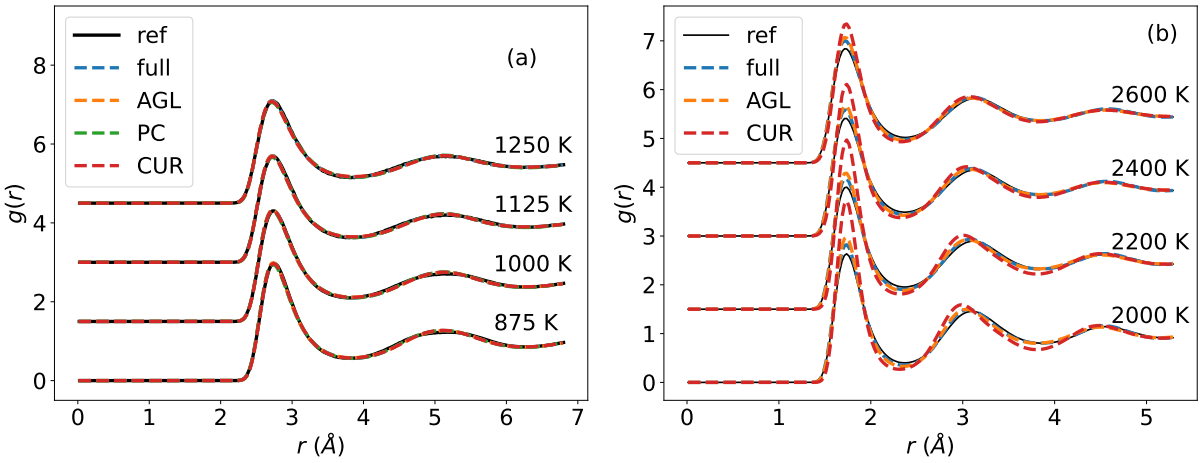

Figure 5.8: RDFs for different feature sets for Al (a) and B (b), at various temperatures. Plots are shifted upwards by 1.5 with increasing temperatures.

is not the case for B (b), where the full feature set and the features selected by AGL both agree well with *ab initio*, but the set selected by CUR show a significant deviation. For the PC set we were unable to perform a stable simulation for B, although we cannot rule out that it is possible to still train a functioning potential on these features; the CUR model has a comparable error, and in a previous iteration was also unstable.

From these simulations we also extract the RDF, shown in figure 5.8. One point that should be stressed here is that our aim is to evaluate the feature selection, rather than how well any of the models reproduce the AIMD reference system results. For the Al case we observe very little difference between the different NNP models, as both the initial large feature set and also the reduced sets following feature selection reproduce the AIMD results fairly well. The same holds for the LJ results, in the supplementary material. In the case of B, already the initial large feature set turns out to be not powerful enough to reproduce the boron RDF faithfully. But the feature selection by AGL does not deteriorate the agreement further, indicating that no significant performance is lost – the feature selection can be only as good as the initial starting point. This is also in contrast with the model trained on the CUR features, which is seen to greatly underperform the other two models, to an increasing extent at lower temperatures. The failure to reproduce the AIMD RDF emphasizes that boron is a challenging system for the training based on Behler-Parrinello SFs and potential energies as targets. Irrespective of this, the agreement with the AIMD MSD is very good also for the reduced feature set. We rationalize this as a result of the dynamics in boron being not predominantly determined by the radial structure encoded in the angle-averaged RDF. This additionally points to the possibility of the standard BP SFs being not well suited for this system. In the supplementary material we present a comparison of simulation results for a B model trained with N2P2, including forces in the training data. That model yielded even worse results than did the models presented here, indicating that this is also not a problem that can be solved just by introducing forces into the training.

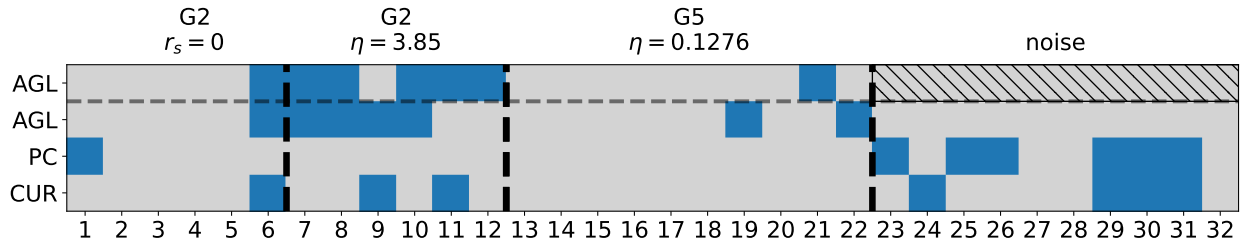

Figure 5.9: Matrix plot of selected features for LJ, with random noise features. Rows correspond to different methods, with discarded features grayed out, and selected ones in blue. Top row, above the dashed line, is from figure 5.3 (c), included for reference.

5.3.5 Confounding Features

Filter methods such as PC and CUR aim to reduce the number of features by looking for subsets that minimize the overlap between those features that are kept. However, this makes them potentially vulnerable to confounding features that are uncorrelated to the relevant input, but by themselves irrelevant. This requires the initial selection of features one starts with to be carefully chosen, in order to minimize irrelevant input. However, in a system with complex structure this might not be obvious to achieve. We demonstrate in the following, that AGL performs much better in the presence of irrelevant input.

For this purpose we return to the LJ system, modifying the starting featureset by adding 10 new features consisting of random noise drawn independently from a set of Gaussian distributions, with means and variances chosen to mimic those of the real features. We note that these fake features were sampled once for each atom and configuration, and as such the values do not vary between epochs. To ensure these fake features are nonnegative, like the real ones, we only work with their absolute values. While the situation considered here is a rather implausible one to occur in a practical setting, where features are unlikely to be truly uncorrelated to the potential energy, it could potentially have implications in situations where there is noise in the training dataset.

Having nothing to do with the real data generating process, these features are truly independent from the other features as well as the target energy. Ideally these features should be discarded, but as they are independent from the real features as well as each other, we expect that neither the PC nor CUR should be able to correctly discard them. This is indeed the case, as illustrated in figure 5.9, showing the features selected by AGL, PC, and CUR, as well as for comparison, the set selected by AGL in the absence of fakes. The PC method clearly did not succeed, as beyond the manually selected feature it only picked out fake features. With CUR we selected some real features, indicating that the method might be more robust compared to the PC in this regard, but still it selected more fakes than real features. In contrast to the filters, the AGL managed to discard the fakes, and select a set of features. And interesting observation is that the set selected by the AGL is slightly different to that selected in the absence of fakes. In fact, the error obtained on this set was 6.97×10^{-3}, below that of the set selected in absence of fakes. This is reminiscent of machine learning

methods where the deliberate addition of noise helps increasing the performance in training.

5.4 Conclusion and Outlook

We have applied the AGL as an embedded feature selection method for choosing atomic features in HDNNPs. This allows for selecting features as part of the training process, taking into account the action of the features in the resulting potential during the selection. In order to evaluate the method we have compared it to previously used unsupervised filter methods that take only into account the features themselves, aiming to minimize redundancy in the description of the local atomic environment. We find that for three test systems, ranging from a simple LJ system, to the highly complicated and directional boron system, that the AGL manages to perform as good as, or better than the other methods. This we consider the main outcome of this work. By utilizing a method that takes into account the NNP predictions, we can reduce the number of atomic features further than methods taking only into account the features themselves.

While we have applied our method to training on only energies, the next step would be to apply the method to the more common setting of fitting also forces during training. A natural question in this case is whether the inclusion of forces changes the features that are selected. It would also be a natural direction to use the method for different types of descriptors. Although the BP SFs are largely in use, and have seen plenty of success, since their introduction many other alternative descriptors have been developed. This is especially relevant considering the difficulty of even our full set of features to better reproduce the overall properties of boron, which could be an indication that the SFs are not ideally suited for this system. One can further consider multicomponent systems for which feature selection using AGL might potentially counteract the combinatorial increase in the number of features seen by traditional SF approaches. In view of recent concerns regarding the stability of MLIPs [267], it would also be interesting to study the extent to which input dimensionality affects the stability of models, and whether this can be alleviated by careful feature selection, or indeed regularization in general.

Code and Data Availability

The authors will make the data of this study available upon reasonable request. The code used in this article is available at https://github.com/JohannesSandberg/HDNNP-AGL

Acknowledgments

We acknowledge the CINES and IDRIS under Project No. INP2227/72914, as well as CIMENT/GRICAD for computational resources. We acknowledge financial support under the French-German project PRCI ANR-DFG SOLIMAT (ANR-22-CE92-0079-01). This work has been partially supported by MIAI@Grenoble Alpes (ANR-19-P3IA-0003). JS

acknowledges funding from the German Academic Exchange Service through DLR-DAAD fellowship grant number 509. We thank Gerhard Jung for suggesting tests with random features.

Chapter 6

Conclusions and Outlook

6.1 Conclusions

In this thesis I have trained high-dimensional neural network potentials, and performed simulations for metallic melts, specifically Al and binary Al-Ni. Two focus has been on using the potentials to study homogeneous nucleation and solidification, and on performing feature selection. I have performed brute-force simulations for both the pure Al system, and for binary Al-Ni, in order to study the homogeneous nucleation of the undercooled liquid. For pure Al a single-step nucleation process was observed for both high and low pressures. In the binary system, the primary focus was on the Ni-rich side of the phase diagram, where AlNi, and pure Ni was examined. For the equimolar AlNi, we observed a direct single-step nucleation into the stable BCC-like B2 phase, similar to previous results using classical potentials. The nucleation for pure Ni was more unexpected, where a single step nucleation into FCC was observed, in contrast to the two-step process previously reported with a classical EAM potential.

A method was developed, for performing active feature selection for high-dimensional neural network potentials, based on the adaptive group lasso. Testing this method on three increasingly complex systems, it was shown to outperform unsupervised filter methods in reducing the number of required descriptors, while maintaining acceptable model accuracy, highlighting the benefit of using a feature selection method that takes into account model predictions. By virtue of being an inherently supervised method, it was shown to be able to identify and deselect artificial confounding features, which unsupervised methods fail to detect. Supervised filter methods, such as the ones of Cersonsky *et al* [80], would, however, likely be able to handle such confounding features as well. The features selected by the method are more explainable, in the sense that, here, they agree with prior physical intuition for what types of interaction (radial or angular) should dominate in the different systems. A further outcome of this research was the failure of the atom centered symmetry functions to fully capture the interactions in one of the test systems, Boron, known for its complex bonding structure. This manifested in a reasonable agreement with *ab initio* on the diffusion coefficients, but a lesser agreement on the pair-correlation function, which would otherwise

be considered an "easier" property to reproduce.

During the course of this thesis work, the field of MLIPs has seen significant development. Especially active learning could have been relevant for the work presented herein, with our approach of performing, then sampling, a large number of AIMD trajectories being possibly quite inefficient. MLIPs are now seeing plenty of use also outside the context of MLIP development, and there has been a push to formalize the sharing of datasets and models through repositories such as the materials cloud, with national projects such as the french DIADEM also represents steps in this direction. With MLIPs becoming increasingly better at reproducing *ab initio*, experimental data integration should be considered a promising way to bring simulations closer to the real world, especially in light of conflicting results from different *ab initio* methods. Going forward, another important focus should be on more interpretable models, with feature selection being an important tool to that end. The combination of training on experimental data, with feature selection, and indeed going towards interpretable ML generally, could likewise open the door for some interesting possibilities of extracting physical knowledge. I will in the following sections go into some more details on future research direction suggested by this work.

6.2 Nucleation

One point of note, between the pure Al model of Chapter 3, and the binary Al-Ni model of Chapter 4, is that they are based on different DFT functionals. The pure Al moles was based on LDA, while PBE GGA was used for the binary system. This points to a further direction of research, namely a thorough comparison of the nucleation pathways of GGA and LDA. Such a study is made possible by MLIPs, as it is impossible to achieve the necessary simulation scales in direct AIMD. Comparing the nucleation pathways, and broader nucleation behavior, between the functionals is further motivated due to the differences between our results with HDNNP and previous EAM results for pure Ni. If changing the density functional is sufficient to cause a change between a single and multi step nucleation process, then this would constitute a rather significant result. This, ultimately, would raise the question of which nucleation path is the correct one, outside of simulation. A possible solution could potentially be given by the use of top-down training methods, discussed below in Section 6.6, providing a more direct link to physical reality through experiment.

6.3 Feature Selection

A major contribution of this work has been the introduction of the AGL method for selecting features in HDNNPs. Being an embedded method, this method holds up well to filter methods that have previously been used for HDNNPs. Nevertheless, there are some limitations to what have been done on this topic so far, and some points that require further attention in future research. Perhaps the greatest limitation of our work on the AGL has been the use of only energies during the fitting of our NNPs. Fitting forces is the *de facto* way to train NNPs,

as these are ultimately what drives the dynamics of the simulated system. As such, testing the method also in this setting is necessary for practical use. This should, in principle, not be a major difficulty, as only the calculation of the gradients of the loss function changes, with the regularization term being identical. However, in practice, some questions are likely to arise. One can immediately wonder whether or not the addition of forces will lead to different sets of features being selected. In principle, as the forces are derived from the energies, the selected features should be the same.

Another issue that arises with the introduction of forces is the additional computational cost. The AGL method is already quite computationally intensive in practice, mainly due to the need to search over the regularization parameter, training several models in the process. It should be mentioned, though, that such experimentation is common practice for tuning other hyperparameters. Nevertheless, extending the method to higher-order optimization methods would be an interesting direction forward. Optimizers such as the Kalman filter [279] and LBFGS [280] are commonly used for HDNNPs, which are often of such small sizes that second order methods are feasible. For instance, proximal Newton type methods, using an LBFGS approximation of the Hessian, could be used [281]. In addition to improving convergence, this could potentially lead to finding a better optimum.

While our aim with AGL has been to select descriptors for HDNNPs, there is a possible dual use for selecting internal nodes of the NNPs. This would allow for selecting network architectures, in addition to the features of the network, and could also be applied in descriptorless NNPs such as Deep Potentials and graph-NN-based models. Having such sparsification would also serve to regularize the NNP, countering the potential overfitting introduced by an overparameterized model. It is well known that NNPs are generally bad at extrapolation, and generalize poorly to unseen configurations. Feature selection, and indeed sparsification broadly, seems a promising approach for improving transferability on the basis of having a simpler model, especially as such improvements have been seen for linearized potentials trained with the lasso [15].

6.4 Boron Descriptors

In our work on feature selection, one of the test systems we used was Boron. This element was chosen on the grounds of it having one of the most complex bonding structures of any pure element, with a notoriously complex crystal structure consisting of B_{12} icosahedrae, and fused icosahedrae, with over 300 atoms per hexagonal unit cell [271]. Although our expectation that the AGL would select more angular features than radial ones was verified, one could argue that the use of more angular features than radial ones, pre-selection, might also influence this. This was, however, necessary in order to obtain a viable potential for this system. Further questions were, however, raised by our work on this system. Even the full featureset did not yield a very good potential, in terms of energy errors, which we attribute to the specific ACSF descriptors being unsuitable for this particular system. Indeed, this has been reported elsewhere in the literature [278]. However, the diffusion constants predicted by the HDNNP were in good agreement to the AIMD values, which is all the more surprising when taking

into account the relatively less well reproduced pair-correlation function. This also relates to the topic of interpretability. One of the motivations for using adaptive feature selection is the hope of being able to draw physical insight from the selected features. Ultimately, while we have made some somewhat *ad hoc* justifications for why certain groups of features were selected for, this has largely been based on prior physical knowledge of the systems in question (for Aluminium and Lennard Jones). Part of the issue is, again, attributable to the features used. This suggests further research, applying the AGL to other descriptor types, such as the bispectrum. In particular, using a descriptor type with a more direct physical interpretation might be of use. On account of its complex bonding, Boron also presents itself as an ideal model system for evaluating and comparing different descriptor types for MLIPs more broadly.

6.5 Multicomponent Feature Selection

The results in the previous chapter, on feature selection, has been exclusively for the single component case. While single component systems are relevant, most real materials are multicomponent. Multi component systems also clearly feature a much greater variation in structures, due to chemical ordering and the additional compositional degrees of freedom. It is thus of interest to extend the method to such a case, enough to warrant some discussion, even if our work on this topic is unfinished at the time of writing.

Applying the AGL approach to multicomponent HDNNPs is, on its surface, no different to the single component case. The weights corresponding to an individual feature are grouped in the same way as before, and their Euclidean norm is added as a regularization. The difference, however, is that now there are several NNPs, one for each species in the system. In the different systems shown in Section 5.3, the selected featuresets was obtained for different regularization strengths λ. Thus, it is not unreasonable to expect that the optimal λ can differ between the different NNPs in the same multicomponent HDNNP. This would be analogous to how the different features in a NN might require different values for λ, there solved by using an adaptive penalty.

A number of ways can be imaged for dealing with the possibly species-dependent regularization strength. The simplest one is to simply not worry about it at all, using the same regularization strength for each species. This could potentially be justified, assuming that the adaptive penalty is able to adjust the regularization strength *inter-network*, similarly to different features *intra-network*. A more thorough approach would be to do a grid search over combinations of species-dependent regularization parameters, but this would quickly become prohibitively expensive as the number of species increase. This could, however, be somewhat alleviated by assuming that the parameters must be similar in magnitude, or by using a more sophisticated parameter search method. For instance, one could perform a line search for each species, keeping the regularization strength of all other species fix to 0. This latter approach would scale linearly in number of species, compared to the exponentially scaling grid-search, but assumes that the species-dependent regularization strengths are independent, which is not necessarily the case.

6.6 Experimental Data Training

Most potentials are trained by fitting to *ab initio* energies and forces. Beside the approximation that goes into representing the quantum-mechanical interaction as a relatively simple functional form, on a much deeper lever this approach assumes that the underlying *ab initio* approach agrees with reality. However, the *ab initio* approaches used in practice, necessarily involve further approximations, in one form or another. In particular, in the context of Kohn Sham DFT, the choice of (approximate) xc potential is not unambiguous, and can have a strong impact on the values of observables. Seeing as different xc potentials constitute inherently different models of the interatomic interaction, they can potentially lead to quite radically different dynamics and structure. This can be a cause of issues, if different approximations are used to construct the dataset used to train a MLIP. Such an issue was in fact encountered during this thesis work when, due to an oversight, binary Al-Ni data created for the potential in Section 4 was combined with the pure Al data used for the potential in Section 3. The former used GGA, while the latter used LDA, consistently leading to potentials where there was an unphysical miscibility gap for Al-rich alloys. Ultimately, as the exact xc potential is unknown, which xc potential approximation to use is up to debate, and dependent on the system under study, and what properties are of interest.

Even if a MLIP is able to perfectly reproduce the PES for a given xc potential, it is itself approximate, and the only way to connect it to reality is by comparing results obtained from the potential, to experimental data. This, however, opens the question, can we not train on experimental data directly? Ultimately, if the aim is to achieve a good description of the real world, as opposed to the world of GGA or LDA, then bringing experimental data into the training process might allow for further improvements of MLIPs. This was one of the original aims of this thesis work as part of SOLIMAT, an international joint project between the DLR, HHU, and UGA, to combine state of the art experiment and simulations using MLIPs to understand structure-dynamics relationships in multicomponent liquid alloys, using Al-Fe-Si as a model system. While, in the end, this work has deviated from that original goal, I will here provide some thoughts on the topic, including laying out an idea on what could be a way to move forward. I will, in the process, also survey some of the recent progress elsewhere in the literature.

The way that some observables are calculated in MD has been discussed in Section 2.2. Specifically, they are often obtained as ensemble averages of some quantity, calculated over a MD trajectory. In principle, it is possible to differentiate through this entire trajectory, and by extension the final calculated quantity, with respect to the parameters of the interaction potential. This type of differentiable molecular dynamics is, in fact, not a new idea, with several tools already available [282, 283, 284]. However, often a large number of timesteps, or large systems, are necessary to obtain accurate values. This can potentially cause stability issues with vanishing or exploding gradients, as well as the difficulty of handling such large amounts of data. Beyond this, doing a MD simulation, and differentiating through it, each timestep is likely to cause an unacceptably long training time.

One promising approach is given by Differentiable Trajectory Reweighting, or DiffTRe [285]. In this approach, a Boltzmann reweighting scheme is used to avoid having to differentiate

through the full MD trajectory. This approach, originally proposed as a top-down training method, has later been combined with the more common bottom-up training approach of MLIPs on *ab initio* energies and forces [286]. Such a mixed approach is potentially preferable to the purely top-down approach, on account of the latter being underconstrained. For a finite set of experimental quantities, many different low-level potentials could fit those, with the *ab initio* providing a useful, physically motivated, way to distinguish between them. However, while useful for static properties like the RDF, static structure factor, pressure, etc., this reweighting scheme does not apply to transport properties which inherently require the evaluation of temporal correlations.

At its core, a given observable O can be considered a functional $O[U]$ of the interatomic potential U. For a given model potential, say HDNNP, defined by a finite set of variables W, this becomes a function of these parameters, $O(W)$. With this it would be possible to construct a regularized version of the loss function

$$L'(W|X, O_{\text{exp.}}) = L(W|X) + \gamma \left(O_{\text{exp.}} - O(W)\right)^2, \tag{6.1}$$

based on the experimental values O_{exp}, *ab initio* data X. The problem then, in this picture, is to calculate $O(W)$. One idea we have had for doing this is by training a auxiliary ML model on this functional. In essence, this would entail creating a number of potentials with different parameter values, possibly by taking a pretrained potential and varying the weights in a physically reasonable manner, calculating physical properties for each, and thus building a database of entries $(W_i, O(W_i))$. Then, one would train a ML model on this database, which would allow to estimate $O(W)$ and construct the regularization in (6.1).

Whether this method of learning an auxiliary functional could be feasible rests on some very important questions. How accurate must $O(W)$ be? This is a very important consideration, since it decides how much effort must go into constructing the database, and training this auxiliary model, which could potentially be quite time-consuming. Limiting oneself to a region in parameter space close to a pretrained potential, resting on the assumption that we are looking for a small correction to *ab initio*, could potentially make the auxiliary training problem more manageable. There are also some very broad concerns that determine the stability of any top-down, or partially top-down, method, namely the accuracy of experimental data, and the calculated observables from MD. Ultimately, whether such an approach would allow for the introduction of transport coefficients into the training process would need to be explored in future work. As an initial evaluation, a simple model like Lennard Jones could be used in lieu of a more complicated MLIP, allowing for faster construction of the potential to property database. This could then be done with either a real noble gas like Argon, or possibly even another Lennard Jones system, as "experimental" ground truth. If the approach is infeasible, even in this simplest possible setting, then clearly other approaches would need to be explored.

Bibliography

[1] Jonathan A Dantzig and Michel Rappaz. Solidification: -Revised & Expanded. EPFL press, 2016.

[2] Ken Kelton and Alan Lindsay Greer. Nucleation in condensed matter: applications in materials and biology. Elsevier, 2010.

[3] Gabriele C Sosso, Ji Chen, Stephen J Cox, Martin Fitzner, Philipp Pedevilla, Andrea Zen, and Angelos Michaelides. Crystal nucleation in liquids: Open questions and future challenges in molecular dynamics simulations. Chemical reviews, 116(12):7078–7116, 2016.

[4] Jürgen Hafner. Ab-initio simulations of materials using vasp: Density-functional theory and beyond. Journal of computational chemistry, 29(13):2044–2078, 2008.

[5] Gabriele C Sosso, Giacomo Miceli, Sebastiano Caravati, Jörg Behler, and Marco Bernasconi. Neural network interatomic potential for the phase change material gete. Physical Review B—Condensed Matter and Materials Physics, 85(17):174103, 2012.

[6] Nongnuch Artrith, Alexander Urban, and Gerbrand Ceder. Constructing first-principles phase diagrams of amorphous Li_xSi using machine-learning-assisted sampling with an evolutionary algorithm. The Journal of chemical physics, 148(24), 2018.

[7] Abhinav CP Jain, Daniel Marchand, Albert Glensk, M Ceriotti, and WA Curtin. Machine learning for metallurgy iii: A neural network potential for Al-Mg-Si. Physical Review Materials, 5(5):053805, 2021.

[8] Jacek Goniakowski, Sarath Menon, Gaétan Laurens, and Julien Lam. Nonclassical nucleation of zinc oxide from a physically motivated machine-learning approach. The Journal of Physical Chemistry C, 126(40):17456–17469, 2022.

[9] Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. Physical review letters, 98(14):146401, 2007.

[10] Albert P Bartók, Mike C Payne, Risi Kondor, and Gábor Csányi. Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. Physical review letters, 104(13):136403, 2010.

[11] Pablo M Piaggi, Jack Weis, Athanassios Z Panagiotopoulos, Pablo G Debenedetti, and Roberto Car. Homogeneous ice nucleation in an ab initio machine-learning model of water. Proceedings of the National Academy of Sciences, 119(33):e2207294119, 2022.

[12] Atsuto Seko, Akira Takahashi, and Isao Tanaka. Sparse representation for a potential energy surface. Physical Review B, 90(2):024101, 2014.

[13] Robert Tibshirani. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society Series B: Statistical Methodology, 58(1):267–288, 1996.

[14] Giulio Imbalzano, Andrea Anelli, Daniele Giofré, Sinja Klees, Jörg Behler, and Michele Ceriotti. Automatic selection of atomic fingerprints and reference configurations for machine-learning potentials. The Journal of chemical physics, 148(24), 2018.

[15] Akshay Krishna Ammothum Kandy, Kevin Rossi, Alexis Raulin-Foissac, Gaétan Laurens, and Julien Lam. Comparing transferability in neural network approaches and linear models for machine-learning interaction potentials. Physical Review B, 107(17):174106, 2023.

[16] Noel Jakse, Johannes Sandberg, Leon F Granz, Anthony Saliou, Philippe Jarry, Emilie Devijver, Thomas Voigtmann, Jürgen Horbach, and Andreas Meyer. Machine learning interatomic potentials for aluminium: application to solidification phenomena. Journal of Physics: Condensed Matter, 35(3):035402, 2022.

[17] Johannes Sandberg, Thomas Voigtmann, Emilie Devijver, and Noel Jakse. Homogeneous nucleation of undercooled Al-Ni melts via a machine-learned interaction potential, 2024.

[18] Johannes Sandberg, Thomas Voigtmann, Emilie Devijver, and Noel Jakse. Feature selection for high-dimensional neural network potentials with the adaptive group lasso. Machine Learning: Science and Technology, 5(2):025043, 2024.

[19] Furio Ercolessi, M Parrinello, and E Tosatti. Simulation of gold in the glue model. Philosophical magazine A, 58(1):213–226, 1988.

[20] SM Foiles, MI Baskes, and Murray S Daw. Embedded-atom-method functions for the fcc metals cu, ag, au, ni, pd, pt, and their alloys. Physical review B, 33(12):7983, 1986.

[21] Me Born. Oppenheimer jr zur quantentheorie der molekeln. Annalen der physik, 84:457–484, 1927.

[22] Richard Phillips Feynman. Forces in molecules. Physical review, 56(4):340, 1939.

[23] Richard Car and Mark Parrinello. Unified approach for molecular dynamics and density-functional theory. Physical review letters, 55(22):2471, 1985.

[24] Vladimir Fock. Näherungsmethode zur lösung des quantenmechanischen mehrkörperproblems. Zeitschrift für Physik, 61:126–148, 1930.

[25] Pierre Hohenberg and Walter Kohn. Inhomogeneous electron gas. Physical review, 136(3B):B864, 1964.

[26] Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. Physical review, 140(4A):A1133, 1965.

[27] Jiří Čížek. On the correlation problem in atomic and molecular systems. calculation of wavefunction components in ursell-type expansion using quantum-field theoretical methods. The Journal of Chemical Physics, 45(11):4256–4266, 1966.

[28] Walter Kohn. Nobel lecture: Electronic structure of matter—wave functions and density functionals. Reviews of Modern Physics, 71(5):1253, 1999.

[29] David K Belashchenko. Computer simulation of liquid metals. Physics-Uspekhi, 56(12):1176, 2013.

[30] Shunsuke Orihara, Yasushi Shibuta, and Tetsuo Mohri. Molecular dynamics simulation of nucleation from undercooled melt of nickel–aluminum alloy and discussion on polymorphism in nucleation. Materials Transactions, 61(4):750–757, 2020.

[31] Sébastien Becker, Emilie Devijver, Rémi Molinier, and Noël Jakse. Glass-forming ability of elemental zirconium. Physical Review B, 102(10):104205, 2020.

[32] Arthur L Samuel. Some studies in machine learning using the game of checkers. IBM Journal of research and development, 3(3):210–229, 1959.

[33] Thomas B Blank, Steven D Brown, August W Calhoun, and Douglas J Doren. Neural network models of potential energy surfaces. The Journal of chemical physics, 103(10):4129–4137, 1995.

[34] Jörg Behler. Four generations of high-dimensional neural network potentials. Chemical Reviews, 121(16):10037–10072, 2021.

[35] Aidan P Thompson, Laura P Swiler, Christian R Trott, Stephen M Foiles, and Garritt J Tucker. Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials. Journal of Computational Physics, 285:316–330, 2015.

[36] Jörg Behler. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. The Journal of chemical physics, 134(7), 2011.

[37] Albert P Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. Physical Review B—Condensed Matter and Materials Physics, 87(18):184115, 2013.

[38] Ralf Drautz. Atomic cluster expansion for accurate and transferable interatomic potentials. Physical Review B, 99(1):014104, 2019.

[39] Linfeng Zhang, Jiequn Han, Han Wang, Roberto Car, and Weinan E. Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics. Physical review letters, 120(14):143001, 2018.

[40] Patrick Reiser, Marlen Neubert, André Eberhard, Luca Torresi, Chen Zhou, Chen Shao, Houssam Metni, Clint van Hoesel, Henrik Schopmans, Timo Sommer, et al. Graph neural networks for materials science and chemistry. Communications Materials, 3(1):93, 2022.

[41] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. Nature communications, 13(1):2453, 2022.

[42] Albert Musaelian, Simon Batzner, Anders Johansson, Lixin Sun, Cameron J Owen, Mordechai Kornbluth, and Boris Kozinsky. Learning local equivariant representations for large-scale atomistic dynamics. Nature Communications, 14(1):579, 2023.

[43] Jonathan Vandermause, Steven B Torrisi, Simon Batzner, Yu Xie, Lixin Sun, Alexie M Kolpak, and Boris Kozinsky. On-the-fly active learning of interpretable bayesian force fields for atomistic rare events. npj Computational Materials, 6(1):20, 2020.

[44] Loup Verlet. Computer" experiments" on classical fluids. i. thermodynamical properties of lennard-jones molecules. Physical review, 159(1):98, 1967.

[45] MBBJM Tuckerman, Bruce J Berne, and Glenn J Martyna. Reversible multiple time scale molecular dynamics. The Journal of chemical physics, 97(3):1990–2001, 1992.

[46] Michael P Allen and Dominic J Tildesley. Computer simulation of liquids. Oxford university press, 2017.

[47] Jean-Pierre Hansen and Ian Ranald McDonald. Theory of simple liquids: with applications to soft matter. Academic press, 2013.

[48] AB Bhatia and D Eo Thornton. Structural aspects of the electrical resistivity of binary alloys. Physical Review B, 2(8):3004, 1970.

[49] S Amore, Jürgen Horbach, and Ivan Egry. Is there a relation between excess volume and miscibility in binary liquid mixtures? The Journal of chemical physics, 134(4), 2011.

[50] Berk Hess. Determining the shear viscosity of model liquids from molecular dynamics simulations. The Journal of chemical physics, 116(1):209–217, 2002.

[51] Dario Alfe and Michael J Gillan. First-principles calculation of transport coefficients. Physical review letters, 81(23):5161, 1998.

[52] Daniel Faken and Hannes Jónsson. Systematic analysis of local atomic structure combined with 3d computer graphics. Computational Materials Science, 2(2):279–286, 1994.

[53] Sébastien Becker, Emilie Devijver, Rémi Molinier, and Noël Jakse. Unsupervised topological learning approach of crystal nucleation. Scientific Reports, 12(1):3195, 2022.

[54] Alexander Stukowski. Structure identification methods for atomistic simulations of crystalline materials. Modelling and Simulation in Materials Science and Engineering, 20(4):045021, 2012.

[55] N Jakse and A Pasturel. Interplay between the structure and dynamics in liquid and undercooled boron: An ab initio molecular dynamics simulation study. The Journal of Chemical Physics, 141(23), 2014.

[56] J Dana Honeycutt and Hans C Andersen. Molecular dynamics study of melting and freezing of small lennard-jones clusters. Journal of Physical Chemistry, 91(19):4950–4963, 1987.

[57] Paul J Steinhardt, David R Nelson, and Marco Ronchetti. Bond-orientational order in liquids and glasses. Physical Review B, 28(2):784, 1983.

[58] Wolfgang Lechner and Christoph Dellago. Accurate determination of crystal structures based on averaged local bond order parameters. The Journal of chemical physics, 129(11), 2008.

[59] John P Perdew and Yue Wang. Accurate and simple analytic representation of the electron-gas correlation energy. Physical review B, 45(23):13244, 1992.

[60] John P Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. Physical review letters, 77(18):3865, 1996.

[61] Ryo Nagai, Ryosuke Akashi, Shu Sasaki, and Shinji Tsuneyuki. Neural-network kohn-sham exchange-correlation potential and its out-of-training transferability. The Journal of chemical physics, 148(24), 2018.

[62] João Paulo Almeida de Mendonça, Lorenzo Antonio Mariano, Emilie Devijver, Noel Jakse, and Roberta Poloni. Artificial neural network-based density functional approach for adiabatic energy differences in transition metal complexes. Journal of Chemical Theory and Computation, 19(21):7555–7566, 2023.

[63] Jürg Hutter. Car–parrinello molecular dynamics. Wiley Interdisciplinary Reviews: Computational Molecular Science, 2(4):604–612, 2012.

[64] Haoyu S Yu, Shaohong L Li, and Donald G Truhlar. Perspective: Kohn-sham density functional theory descending a staircase. The Journal of chemical physics, 145(13), 2016.

[65] Ken-Ichi Funahashi. On the approximate realization of continuous mappings by neural networks. Neural networks, 2(3):183–192, 1989.

[66] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. Neural networks, 2(5):359–366, 1989.

[67] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. nature, 323(6088):533–536, 1986.

[68] Leslie N Smith. Cyclical learning rates for training neural networks. In 2017 IEEE winter conference on applications of computer vision (WACV), pages 464–472. IEEE, 2017.

[69] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983, 2016.

[70] D.Randall Wilson and Tony R. Martinez. The general inefficiency of batch training for gradient descent learning. Neural Networks, 16(10):1429–1451, 2003.

[71] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. The elements of statistical learning: data mining, inference, and prediction, 2017.

[72] Stephen Hanson and Lorien Pratt. Comparing biases for minimal network construction with back-propagation. Advances in neural information processing systems, 1, 1988.

[73] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. arXiv e-prints, pages arXiv–1412, 2014.

[74] I Loshchilov. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101, 2017.

[75] Roberto Todeschini and Viviana Consonni. Handbook of molecular descriptors. John Wiley & Sons, 2008.

[76] Michael Gastegger, Ludwig Schwiedrzik, Marius Bittermann, Florian Berzsenyi, and Philipp Marquetand. wacsf—weighted atom-centered symmetry functions as descriptors in machine learning potentials. The Journal of chemical physics, 148(24), 2018.

[77] Joe D Morrow, John LA Gardner, and Volker L Deringer. How to validate machine-learned interatomic potentials. The Journal of chemical physics, 158(12), 2023.

[78] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. Computers & electrical engineering, 40(1):16–28, 2014.

[79] Michael W Mahoney and Petros Drineas. CUR matrix decompositions for improved data analysis. Proceedings of the National Academy of Sciences, 106(3):697–702, 2009.

[80] Rose K Cersonsky, Benjamin A Helfrecht, Edgar A Engel, Sergei Kliavinek, and Michele Ceriotti. Improving sample and feature selection with principal covariates regression. Machine Learning: Science and Technology, 2(3):035038, jul 2021.

[81] Hansheng Wang and Chenlei Leng. A note on adaptive group lasso. Computational statistics & data analysis, 52(12):5277–5286, 2008.

[82] Neal Parikh and Stephen Boyd. Proximal algorithms. Foundations and Trends in Optimization, 1(3):127–239, 2014.

[83] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society Series B: Statistical Methodology, 68(1):49–67, 2006.

[84] Jean Feng and Noah Simon. Sparse-input neural networks for high-dimensional nonparametric regression and classification. arXiv preprint arXiv:1711.07592, 2017.

[85] Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. Journal of computational and graphical statistics, 22(2):231–245, 2013.

[86] Vu C Dinh and Lam S Ho. Consistent feature selection for analytic deep neural networks. Advances in Neural Information Processing Systems, 33:2420–2431, 2020.

[87] Hui Zou. The adaptive lasso and its oracle properties. Journal of the American statistical association, 101(476):1418–1429, 2006.

[88] Jason D Lee, Dennis L Sun, Yuekai Sun, and Jonathan E Taylor. Exact post-selection inference, with application to the lasso. The Annals of Statistics, 44(3):907, 2016.

[89] Ismael Lemhadri, Feng Ruan, Louis Abraham, and Robert Tibshirani. Lassonet: A neural network with feature sparsity. Journal of Machine Learning Research, 22(127):1–29, 2021.

[90] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[91] Andreas Singraber, Jörg Behler, and Christoph Dellago. Library-based lammps implementation of high-dimensional neural network potentials. Journal of chemical theory and computation, 15(3):1827–1840, 2019.

[92] Martin Volmer and A Weber. Keimbildung in übersättigten gebilden. Zeitschrift für physikalische Chemie, 119(1):277–301, 1926.

[93] L Farkas. The velocity of nucleus formation in supersaturated vapors. Z. phys. Chem, 125:236–242, 1927.

[94] Richard Becker and Werner Döring. Kinetische behandlung der keimbildung in übersättigten dämpfen. Annalen der physik, 416(8):719–752, 1935.

[95] Pablo G Debenedetti, Yi-Yeoun Kim, Fiona C Meldrum, and Hajime Tanaka. Special topic preface: Nucleation—current understanding approaching 150 years after gibbs. The Journal of Chemical Physics, 160(10), 2024.

[96] Pieter Rein Ten Wolde, Maria J Ruiz-Montero, and Daan Frenkel. Numerical evidence for bcc ordering at the surface of a critical fcc nucleus. Physical review letters, 75(14):2714, 1995.

[97] Pieter Rein ten Wolde, Maria J Ruiz-Montero, and Daan Frenkel. Numerical calculation of the rate of crystal nucleation in a lennard-jones system at moderate undercooling. The Journal of chemical physics, 104(24):9932–9947, 1996.

[98] Sébastien Becker, Emilie Devijver, Rémi Molinier, and Noël Jakse. Crystal nucleation in Al-Ni alloys: an unsupervised chemical and topological learning approach. arXiv preprint arXiv:2210.01894, 2022.

[99] Sh Alexander and J McTague. Should all crystals be bcc? landau theory of solidification and crystal nucleation. Physical Review Letters, 41(10):702, 1978.

[100] Noel Jakse, Johannes Sandberg, Leon F Granz, Anthony Saliou, Philippe Jarry, Emilie Devijver, Thomas Voigtmann, Jürgen Horbach, and Andreas Meyer. Machine learning interatomic potentials for aluminium: application to solidification phenomena. Journal of Physics: Condensed Matter, 51(3):035402, 2022.

[101] Noel Jakse and Alain Pasturel. Liquid aluminum: Atomic diffusion and viscosity from ab initio molecular dynamics. Scientific reports, 3(1):3135, 2013.

[102] N Jakse and A Pasturel. Dynamic properties of liquid and undercooled aluminum. Journal of Physics: Condensed Matter, 25(28):285103, 2013.

[103] Noël Jakse and Taras Bryk. Pressure evolution of transverse collective excitations in liquid al along the melting line. The Journal of Chemical Physics, 151(3), 2019.

[104] Franz Demmel, Louis Hennet, and Noel Jakse. The intimate relationship between structural relaxation and the energy landscape of monatomic liquid metals. Scientific Reports, 11(1):11815, 2021.

[105] Justin S Smith, Benjamin Nebgen, Nithin Mathew, Jie Chen, Nicholas Lubbers, Leonid Burakovsky, Sergei Tretiak, Hai Ah Nam, Timothy Germann, Saryu Fensin, et al. Automated discovery of a robust interatomic potential for aluminum. Nature communications, 12(1):1257, 2021.

[106] Joseph R Davis. Alloying: understanding the basics. ASM international, 2001.

[107] C Patrick Royall and Stephen R Williams. The role of local structure in dynamical arrest. Physics Reports, 560:1–75, 2015.

[108] Noel Jakse and A Pasturel. Liquid-liquid phase transformation in silicon: Evidence<? format?> from first-principles molecular dynamics simulations. Physical review letters, 99(20):205702, 2007.

[109] Luigi Bonati and Michele Parrinello. Silicon liquid structure and crystal nucleation from ab initio deep metadynamics. Physical review letters, 121(26):265701, 2018.

[110] Mike C Payne, Michael P Teter, Douglas C Allan, TA Arias, and ad JD Joannopoulos. Iterative minimization techniques for ab initio total-energy calculations: molecular dynamics and conjugate gradients. Reviews of modern physics, 64(4):1045, 1992.

[111] Kieron Burke. Perspective on density functional theory. The Journal of chemical physics, 136(15), 2012.

[112] A Pasturel and N Jakse. Role of five-fold symmetry in undercooled Al-Cu binary alloys. Journal of Applied Physics, 123(14), 2018.

[113] Jūrgen Hafner. From hamiltonians to phase diagrams. MRS Online Proceedings Library (OPL), 63:73, 1985.

[114] NW Ashcroft. Electron-ion pseudopotentials in metals. Physics Letters, 23(1):48–50, 1966.

[115] John M Wills and Walter A Harrison. Interionic interactions in transition metals. Physical Review B, 28(8):4363, 1983.

[116] John A Moriarty. Analytic representation of multi-ion interatomic potentials in transition metals. Physical Review B, 42(3):1609, 1990.

[117] N Jakse and JL Bretonnet. Structure and thermodynamics of liquid transition metals: integral-equation study of Fe, Co and Ni. Journal of Physics: Condensed Matter, 7(20):3803, 1995.

[118] Murray S Daw and Michael I Baskes. Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. Physical Review B, 29(12):6443, 1984.

[119] Murray S Daw, Stephen M Foiles, and Michael I Baskes. The embedded-atom method: a review of theory and applications. Materials Science Reports, 9(7-8):251–310, 1993.

[120] Michael I Baskes. Modified embedded-atom potentials for cubic materials and impurities. Physical review B, 46(5):2727, 1992.

[121] HS Huang, LQ Ai, ACT Van Duin, M Chen, and YJ Lü. Reaxff reactive force field for molecular dynamics simulations of liquid cu and zr metals. The Journal of Chemical Physics, 151(9), 2019.

[122] David G Pettifor. Bonding and structure of molecules and solids. Oxford university press, 1995.

[123] Hongxiang Zong, Ghanshyam Pilania, Xiangdong Ding, Graeme J Ackland, and Turab Lookman. Developing an interatomic potential for martensitic phase transformations in zirconium by machine learning. npj Computational Materials, 4(1):48, 2018.

[124] Alexandra M Goryaeva, Clovis Lapointe, Chendi Dai, Julien Dérès, Jean-Bernard Maillet, and Mihai-Cosmin Marinica. Reinforcing materials modelling by encoding the structures of defects in crystalline solids into distortion scores. Nature communications, 11(1):4691, 2020.

[125] Jörg Behler. Constructing high-dimensional neural network potentials: a tutorial review. International Journal of Quantum Chemistry, 115(16):1032–1050, 2015.

[126] Jörg Behler. Perspective: Machine learning potentials for atomistic simulations. The Journal of chemical physics, 145(17), 2016.

[127] Rampi Ramprasad, Rohit Batra, Ghanshyam Pilania, Arun Mannodi-Kanakkithodi, and Chiho Kim. Machine learning in materials informatics: recent applications and prospects. npj Computational Materials, 3(1):54, 2017.

[128] Jonathan Schmidt, Mário RG Marques, Silvana Botti, and Miguel AL Marques. Recent advances and applications of machine learning in solid-state materials science. npj computational materials, 5(1):83, 2019.

[129] Mário RG Marques, Jakob Wolff, Conrad Steigemann, and Miguel AL Marques. Neural network force fields for simple metals and semiconductors: construction and application to the calculation of phonons and melting temperatures. Physical Chemistry Chemical Physics, 21(12):6506–6516, 2019.

[130] Alexandra M Goryaeva, Jean-Bernard Maillet, and Mihai-Cosmin Marinica. Towards better efficiency of interatomic linear machine learning potentials. Computational Materials Science, 166:200–209, 2019.

[131] Tim Mueller, Alberto Hernandez, and Chuhong Wang. Machine learning for interatomic potential models. The Journal of chemical physics, 152(5), 2020.

[132] Andreas Singraber, Tobias Morawietz, Jörg Behler, and Christoph Dellago. Parallel multistream training of high-dimensional neural network potentials. Journal of chemical theory and computation, 15(5):3075–3092, 2019.

[133] Aidan P Thompson, H Metin Aktulga, Richard Berger, Dan S Bolintineanu, W Michael Brown, Paul S Crozier, Pieter J In't Veld, Axel Kohlmeyer, Stan G Moore, Trung Dac Nguyen, et al. Lammps-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. Computer Physics Communications, 271:108171, 2022.

[134] Ask Hjorth Larsen, Jens Jørgen Mortensen, Jakob Blomqvist, Ivano E Castelli, Rune Christensen, Marcin Dułak, Jesper Friis, Michael N Groves, Bjørk Hammer, Cory Hargus, et al. The atomic simulation environment—a python library for working with atoms. Journal of Physics: Condensed Matter, 29(27):273002, 2017.

[135] Zhenwei Li, James R Kermode, and Alessandro De Vita. Molecular dynamics with on-the-fly machine learning of quantum-mechanical forces. Physical review letters, 114(9):096405, 2015.

[136] Ryosuke Jinnouchi, Ferenc Karsai, and Georg Kresse. On-the-fly machine learning force field generation: Application to melting points. Physical Review B, 100(1):014105, 2019.

[137] Albert P Bartók and Gábor Csányi. G aussian approximation potentials: A brief tutorial introduction. International Journal of Quantum Chemistry, 115(16):1051–1057, 2015.

[138] Venkatesh Botu, Rohit Batra, James Chapman, and Rampi Ramprasad. Machine learning force fields: construction, validation, and outlook. The Journal of Physical Chemistry C, 121(1):511–522, 2017.

[139] Tobias Morawietz, Andreas Singraber, Christoph Dellago, and Jörg Behler. How van der waals interactions determine the unique properties of water. Proceedings of the National Academy of Sciences, 113(30):8368–8373, 2016.

[140] Gabriele C Sosso, Giacomo Miceli, Sebastiano Caravati, Federico Giberti, Jörg Behler, and Marco Bernasconi. Fast crystallization of the phase change compound gete by large-scale molecular dynamics simulations. The journal of physical chemistry letters, 4(24):4241–4246, 2013.

[141] Chao Zhang, Yang Sun, Hai-Di Wang, Feng Zhang, Tong-Qi Wen, Kai-Ming Ho, and Cai-Zhuang Wang. Crystallization of the P3Sn4 phase upon cooling P2Sn5 liquid by molecular dynamics simulation using a machine learning interatomic potential. The Journal of Physical Chemistry C, 125(5):3127–3133, 2021.

[142] Alain Pasturel and Noel Jakse. Atomic-scale structural signature of dynamic heterogeneities in metallic liquids. npj Computational Materials, 3(1):33, 2017.

[143] John Russo and Hajime Tanaka. Crystal nucleation as the ordering of multiple order parameters. The Journal of Chemical Physics, 145(21), 2016.

[144] Ivan Kruglov, Oleg Sergeev, Alexey Yanilkin, and Artem R Oganov. Energy-free machine learning force field for aluminum. Scientific reports, 7(1):8512, 2017.

[145] Anton S Bochkarev, Ambroise van Roekeghem, Stefano Mossa, and Natalio Mingo. Anharmonic thermodynamics of vacancies using a neural network potential. Physical Review Materials, 3(9):093803, 2019.

[146] DJ Wales. Energy Landscapes (Cambridge Molecular Science). Cambridge, UK: Cambridge University Press, 2003.

[147] Louis A Girifalco and Victor G Weizer. Application of the morse potential function to cubic metals. Physical Review, 114(3):687, 1959.

[148] Karsten Wedel Jacobsen, JK Norskov, and Martti J Puska. Interatomic interactions in the effective-medium theory. Physical Review B, 35(14):7423, 1987.

[149] J Mei and JW Davenport. Free-energy calculations and the melting point of al. Physical Review B, 46(1):21, 1992.

[150] Furio Ercolessi and James B Adams. Interatomic potentials from first-principles calculations: the force-matching method. Europhysics Letters, 26(8):583, 1994.

[151] Y Mishin, Diana Farkas, MJ Mehl, and DA Papaconstantopoulos. Interatomic potentials for monoatomic metals from experimental data and ab initio calculations. Physical Review B, 59(5):3393, 1999.

[152] Jess B Sturgeon and Brian B Laird. Adjusting the melting point of a model system via gibbs-duhem integration: Application to a model of aluminum. Physical Review B, 62(22):14720, 2000.

[153] Byeong-Joo Lee, Jae-Hyeok Shim, and MI Baskes. Semiempirical atomic potentials for the fcc metals cu, ag, au, ni, pd, pt, al, and pb based on first and second nearest-neighbor modified embedded atom method. Physical Review B, 68(14):144112, 2003.

[154] Xiang-Yang Liu, Furio Ercolessi, and James B Adams. Aluminium interatomic potential from density functional theory calculations with improved stacking fault energy. Modelling and Simulation in Materials Science and Engineering, 12(4):665, 2004.

[155] EB El Mendoub, R Albaki, I Charpentier, J-L Bretonnet, J-F Wax, and N Jakse. Molecular dynamics and integral equation study of the structure and thermodynamics of polyvalent liquid metals. Journal of non-crystalline solids, 353(32-40):3475–3479, 2007.

[156] MI Mendelev, MJ Kramer, Chandler A Becker, and M Asta. Analysis of semi-empirical interatomic potentials appropriate for simulation of crystalline and liquid al and cu. Philosophical Magazine, 88(12):1723–1750, 2008.

[157] JM Winey, Alison Kubota, and YM Gupta. A thermodynamic approach to determine accurate potentials for molecular dynamics simulations: thermoelastic response of aluminum. Modelling and Simulation in Materials Science and Engineering, 17(5):055004, 2009.

[158] VV Zhakhovskii, NA Inogamov, Yu V Petrov, SI Ashitkov, and K Nishihara. Molecular dynamics simulation of femtosecond ablation and spallation with different interatomic potentials. Applied Surface Science, 255(24):9592–9596, 2009.

[159] Kamal Choudhary, Tao Liang, Aleksandr Chernatynskiy, Simon R Phillpot, and Susan B Sinnott. Charge optimized many-body (comb) potential for Al2O3 materials, interfaces, and nanostructures. Journal of Physics: Condensed Matter, 27(30):305004, 2015.

[160] MI Pascuet and Julian Roberto Fernández. Atomic interaction of the meam type for the study of intermetallics in the al–u alloy. Journal of Nuclear Materials, 467:229–239, 2015.

[161] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. The elements of statistical learning: data mining, inference, and prediction, volume 2. Springer, 2009.

[162] The Python package can be found here : https://www.tensorflow.org.

[163] Oliver T Unke, Stefan Chmiela, Huziel E Sauceda, Michael Gastegger, Igor Poltavsky, Kristof T Schütt, Alexandre Tkatchenko, and Klaus-Robert Müller. Machine learning force fields. Chemical Reviews, 121(16):10142–10186, 2021.

[164] Georg Kresse and Jürgen Furthmüller. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. Computational materials science, 6(1):15–50, 1996.

[165] David M Ceperley and Berni J Alder. Ground state of the electron gas by a stochastic method. Physical review letters, 45(7):566, 1980.

[166] John P Perdew and Alex Zunger. Self-interaction correction to density-functional approximations for many-electron systems. Physical review B, 23(10):5048, 1981.

[167] Georg Kresse and Daniel Joubert. From ultrasoft pseudopotentials to the projector augmented-wave method. Physical review b, 59(3):1758, 1999.

[168] Hendrik J Monkhorst and James D Pack. Special points for brillouin-zone integrations. Physical review B, 13(12):5188, 1976.

[169] Peter E Blöchl, Ove Jepsen, and Ole Krogh Andersen. Improved tetrahedron method for brillouin-zone integrations. Physical Review B, 49(23):16223, 1994.

[170] Marc J Assael, Konstantinos Kakosimos, R Michael Banish, Jürgen Brillo, Ivan Egry, Robert Brooks, Peter N Quested, Kenneth C Mills, Akira Nagashima, Yuzuru Sato, et al. Reference data for the density and viscosity of liquid aluminum and liquid iron. Journal of physical and chemical reference data, 35(1):285–300, 2006.

[171] Dario Alfe. First-principles simulations of direct coexistence of solid and liquid aluminum. Physical Review B, 68(6):064423, 2003.

[172] F Demmel, D Szubrin, W-C Pilgrim, and C Morkel. Diffusion in liquid aluminium probed by quasielastic neutron scattering. Physical Review B—Condensed Matter and Materials Physics, 84(1):014307, 2011.

[173] F Kargl, H Weis, T Unruh, and A Meyer. Self diffusion in liquid aluminium. In Journal of Physics: Conference Series, volume 340, page 012077. IOP Publishing, 2012.

[174] Travis Sjostrom, Scott Crockett, and Sven Rudin. Multiphase aluminum equations of state via density functional theory. Physical Review B, 94(14):144101, 2016.

[175] John P Perdew, John A Chevary, Sy H Vosko, Koblar A Jackson, Mark R Pederson, Dig J Singh, and Carlos Fiolhais. Atoms, molecules, solids, and surfaces: Applications of the generalized gradient approximation for exchange and correlation. Physical review B, 46(11):6671, 1992.

[176] Anders S Christensen and O Anatole Von Lilienfeld. On the role of gradients for machine learning of molecular energies and forces. Machine Learning: Science and Technology, 1(4):045018, 2020.

[177] Daan Frenkel and Berend Smit. Understanding Molecular Simulation. Academic Press, San Diego, second edition edition, 2002.

[178] Alexander Stukowski. Visualization and analysis of atomistic simulation data with ovito–the open visualization tool. Modelling and simulation in materials science and engineering, 18(1):015012, 2009.

[179] Sarath Menon, Grisell Díaz Leines, and Jutta Rogal. pyscal: A python module for structural analysis of atomic environments. Journal of Open Source Software, 4(43):1824, 2019.

[180] NA Mauro, JC Bendert, AJ Vogt, JM Gewin, and KF Kelton. High energy x-ray scattering studies of the local order in liquid al. The Journal of chemical physics, 135(4), 2011.

[181] Kurt Binder and Walter Kob. Glassy materials and disordered solids: An introduction to their statistical mechanics. World scientific, 2011.

[182] Dieter M Herlach, Sven Binder, Peter Galenko, Jan Gegner, Dirk Holland-Moritz, Stefan Klein, Matthias Kolbe, and Thomas Volkmann. Containerless undercooled melts: ordering, nucleation, and dendrite growth. Metallurgical and Materials Transactions A, 46:4921–4936, 2015.

[183] J Orava and A áL Greer. Fast and slow crystal growth kinetics in glass-forming melts. The Journal of chemical physics, 140(21), 2014.

[184] Matthias Leitner, Thomas Leitner, Alexander Schmon, Kirmanj Aziz, and Gernot Pottlacher. Thermophysical properties of liquid aluminum. Metallurgical and Materials Transactions A, 48:3036–3045, 2017.

[185] Gene Simmons and Herbert Yang. Single Crystal Elastic Constants and Calculated Aggregate Properties (A Handbook). The MIT Press, Cambridge, second edition edition, 2009.

[186] DY Sun, M Asta, and JJ Hoyt. Kinetic coefficient of Ni solid-liquid interfaces from molecular-dynamics simulations. Physical Review B, 69(2):024108, 2004.

[187] James R Morris, CZ Wang, KM Ho, and Che Ting Chan. Melting line of aluminum from simulations of coexisting phases. Physical Review B, 49(5):3109, 1994.

[188] James R Morris and Xueyu Song. The melting lines of model systems calculated from coexistence simulations. The Journal of chemical physics, 116(21):9352–9358, 2002.

[189] Tatyana Zykova-Timan, Roberto E Rozas, Jürgen Horbach, and Kurt Binder. Computer simulation studies of finite-size broadening of solid–liquid interfaces: From hard spheres to nickel. Journal of Physics: Condensed Matter, 21(46):464102, 2009.

[190] Tatyana Zykova-Timan, Jürgen Horbach, and Kurt Binder. Monte carlo simulations of the solid-liquid transition in hard spheres and colloid-polymer mixtures. The Journal of chemical physics, 133(1), 2010.

[191] Philipp Kuhn and Jürgen Horbach. Molecular dynamics simulation of crystal growth in al 50 ni 50: The generation of defects. Physical Review B—Condensed Matter and Materials Physics, 87(1):014105, 2013.

[192] Ronald Benjamin and Jürgen Horbach. Crystal growth kinetics in lennard-jones and weeks-chandler-andersen systems along the solid-liquid coexistence line. The Journal of chemical physics, 143(1), 2015.

[193] RE Rozas, AD Demirağ, PG Toledo, and J Horbach. Thermophysical properties of liquid Ni around the melting temperature from molecular dynamics simulation. The Journal of Chemical Physics, 145(6), 2016.

[194] RE Rozas, Luis G MacDowell, PG Toledo, and J Horbach. Crystal growth of bcc titanium from the melt and interfacial properties: a molecular dynamics simulation study. The Journal of Chemical Physics, 154(18), 2021.

[195] Reinhard Boehler and Marvin Ross. Melting curve of aluminum in a diamond cell to 0.8 mbar: implications for iron. Earth and Planetary Science Letters, 153(3-4):223–227, 1997.

[196] A Hänström and P Lazor. High pressure melting and equation of state of aluminium. Journal of alloys and compounds, 305(1-2):209–215, 2000.

[197] JW Shaner, JM Brown, and RG McQueen. Melting of metals above 100 gpa. In Materials Research Society Symposia Proceedings, volume 22, 1984.

[198] J Bouchet, F Bottin, G Jomard, and G Zérah. Melting curve of aluminum up to 300 gpa obtained through ab initio molecular dynamics simulations. Physical Review B—Condensed Matter and Materials Physics, 80(9):094102, 2009.

[199] Wilhelm Ostwald. Studies on formation and transformation of solid materials. Z. Phys. Chem, 22:289–330, 1897.

[200] Avik Mahata, Mohsen Asle Zaeem, and Michael I Baskes. Understanding homogeneous nucleation in solidification of aluminum by molecular dynamics simulations. Modelling and Simulation in Materials Science and Engineering, 26(2):025007, 2018.

[201] LL Zhou, JM Pan, ZA Tian, L Lang, YF Mo, and KJ Dong. Atomic structure evolutions and liquid-solid phase transition mechanisms of liquid al during rapid cooling. RSC Adv, 11:39829–39837, 2021.

[202] Eric J Mittemeijer. Fundamentals of materials science, volume 8. Springer, 2010.

[203] Jihan Zhou, Yongsoo Yang, Yao Yang, Dennis S Kim, Andrew Yuan, Xuezeng Tian, Colin Ophus, Fan Sun, Andreas K Schmid, Michael Nathanson, et al. Observing crystal nucleation in four dimensions using atomic electron tomography. Nature, 570(7762):500–503, 2019.

[204] Tian Hui Zhang and Xiang Yang Liu. Experimental modelling of single-particle dynamic processes in crystallization by controlled colloidal assembly. Chemical Society Reviews, 43(7):2324–2347, 2014.

[205] M. Radu and T. Schilling. Solvent hydrodynamics speed up crystal nucleation in suspensions of hard spheres. epl, 105:26001, 2014.

[206] Stefan Auer and Daan Frenkel. Prediction of absolute crystal-nucleation rate in hard-sphere colloids. Nature, 409(6823):1020–1023, 2001.

[207] Takeshi Kawasaki and Hajime Tanaka. Formation of a crystal nucleus from liquid. Proceedings of the National Academy of Sciences, 107(32):14036–14041, 2010.

[208] Søren Toxvaerd. The role of local bond-order at crystallization in a simple supercooled liquid. The European Physical Journal B, 93:1–8, 2020.

[209] Chunguang Tang and Peter Harrowell. Anomalously slow crystal growth of the glass-forming alloy CuZr. Nature materials, 12(6):507–511, 2013.

[210] Yasushi Shibuta, Shinji Sakane, Eisuke Miyoshi, Shin Okita, Tomohiro Takaki, and Munekazu Ohno. Heterogeneity in homogeneous nucleation from billion-atom molecular dynamics simulation of solidification of pure metal. Nature communications, 8(1):10, 2017.

[211] Takuya Fujinaga, Yoshimi Watanabe, and Yasushi Shibuta. Nucleation dynamics in al solidification with al-ti refiners by molecular dynamics simulation. Computational Materials Science, 182:109763, 2020.

[212] Baoshuang Shang, Noël Jakse, Pengfei Guan, Weihua Wang, and Jean-louis Barrat. Influence of oscillatory shear on nucleation in metallic glasses: A molecular dynamics study. Acta Materialia, 246:118668, 2023.

[213] Kimberly Chenoweth, Adri CT Van Duin, and William A Goddard. Reaxff reactive force field for molecular dynamics simulations of hydrocarbon oxidation. The Journal of Physical Chemistry A, 112(5):1040–1053, 2008.

[214] HL Peng, Fan Yang, ST Liu, Dirk Holland-Moritz, Tobias Kordel, T Hansen, and Th Voigtmann. Chemical effect on the structural and dynamical properties in Zr-Ni-Al liquids. Physical Review B, 100(10):104202, 2019.

[215] Noël Jakse and Alain Pasturel. Local order of liquid and supercooled zirconium by ab initio molecular dynamics. Physical review letters, 91(19):195501, 2003.

[216] MMG Alemany, LJ Gallego, and David J González. Kohn-sham ab initio molecular dynamics study of liquid al near melting. Physical Review B—Condensed Matter and Materials Physics, 70(13):134206, 2004.

[217] Manuel MG Alemany, RC Longo, LJ Gallego, DJ González, LE González, Murilo L Tiago, and James R Chelikowsky. Ab initio molecular dynamics simulations of the static, dynamic, and electronic properties of liquid pb using real-space pseudopotentials. Physical Review B—Condensed Matter and Materials Physics, 76(21):214203, 2007.

[218] J Souto, MMG Alemany, LJ Gallego, LE González, and DJ González. Ab initio molecular dynamics study of the static, dynamic, and electronic properties of liquid bi near melting using real-space pseudopotentials. Physical Review B—Condensed Matter and Materials Physics, 81(13):134201, 2010.

[219] BG Del Rio, LE González, and DJ González. Ab initio study of several static and dynamic properties of bulk liquid ni near melting. The Journal of Chemical Physics, 146(3), 2017.

[220] Beatriz G Del Rio, Carlos Pascual, Luis E González, and David J González. Structure and dynamics of the liquid 3d transition metals near melting. an ab initio study. Journal of Physics: Condensed Matter, 32(21):214005, 2020.

[221] MMG Alemany, Jaime Souto-Casares, Luis E González, and David J González. Static structure, collective dynamics and transport coefficients in the liquid Li-Pb alloy. an ab initio molecular dynamics study. Journal of Molecular Liquids, 344:117775, 2021.

[222] Alaa Fahs, Philippe Jarry, and Noël Jakse. Structure-dynamics relationship in Al-Mg-Si liquid alloys. Physical Review B, 108(22):224202, 2023.

[223] Saichao Cao, Mingxu Xia, Noel Jakse, Long Zeng, Pengfei Yu, Yimeng Zhao, Wenquan Lu, and Jianguo Li. Influence of zr element on the atomic structure of Al-Cu alloy liquid. Scripta Materialia, 248:116143, 2024.

[224] Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. Physical review letters, 98(14):146401, 2007.

[225] Akira Takahashi, Atsuto Seko, and Isao Tanaka. Linearized machine-learning interatomic potentials for non-magnetic elemental metals: Limitation of pairwise descriptors and trend of predictive power. The Journal of chemical physics, 148(23), 2018.

[226] Volker L Deringer, Albert P Bartók, Noam Bernstein, David M Wilkins, Michele Ceriotti, and Gábor Csányi. Gaussian process regression for materials and molecules. Chemical Reviews, 121(16):10073–10141, 2021.

[227] Jörg Behler. Constructing high-dimensional neural network potentials: a tutorial review. International Journal of Quantum Chemistry, 115(16):1032–1050, 2015.

[228] Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet–a deep learning architecture for molecules and materials. The Journal of Chemical Physics, 148(24), 2018.

[229] M Maret, T Pomme, A Pasturel, and P Chieux. Structure of liquid Al80Ni20 alloy. Physical Review B, 42(3):1598, 1990.

[230] N Jakse, O Lebacq, and A Pasturel. Ab initio molecular-dynamics simulations of short-range order in liquid Al80Mn20 and Al80Ni20 alloys. Physical review letters, 93(20):207801, 2004.

[231] N Jakse, O Le Bacq, and A Pasturel. Chemical and icosahedral short-range orders in liquid and undercooled Al80Mn20 and Al80Ni20 alloys: A first-principles-based approach. The Journal of chemical physics, 123(10), 2005.

[232] SK Das, J Horbach, MM Koza, S Mavila Chatoth, and A Meyer. Influence of chemical short-range order on atomic diffusion in al–ni melts. Applied Physics Letters, 86(1), 2005.

[233] Philipp Kuhn, Jürgen Horbach, Florian Kargl, Andreas Meyer, and Th Voigtmann. Diffusion and interdiffusion in binary metallic melts. Physical Review B, 90(2):024309, 2014.

[234] N Jakse and AJAPL Pasturel. Dynamic properties and local order in liquid Al-Ni alloys. Applied Physics Letters, 105(13), 2014.

[235] N Jakse and A Pasturel. Relationship between structure and dynamics in liquid $Al_{1-x}Ni_x$ alloys. The Journal of Chemical Physics, 143(8), 2015.

[236] Y Mishin, MJ Mehl, and DA Papaconstantopoulos. Embedded-atom potential for B2-NiAl. Physical review B, 65(22):224114, 2002.

[237] GP Purja Pun and Y Mishin. Development of an interatomic potential for the Ni-Al system. Philosophical Magazine, 89(34-36):3245–3267, 2009.

[238] Georg Kresse and Jürgen Hafner. Ab initio molecular dynamics for liquid metals. Physical review B, 47(1):558, 1993.

[239] Shuichi Nosé. A molecular dynamics method for simulations in the canonical ensemble. Molecular Physics, 52(2):255–268, 1984.

[240] William G. Hoover. Canonical dynamics: Equilibrium phase-space distributions. Physical Review A, 31(3):1695–1697, 1985.

[241] P. E. Blöchl. Projector augmented-wave method. Physical Review B, 50(24):17953–17979, 1994.

[242] Yue Wang and John P Perdew. Correlation hole of the spin-polarized electron gas, with exact small-wave-vector and high-density scaling. Physical Review B, 44(24):13298, 1991.

[243] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, et al. Commentary: The materials project: A materials genome approach to accelerating materials innovation. APL materials, 1(1), 2013.

[244] Jörg Behler. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. The Journal of chemical physics, 134(7), 2011.

[245] Johannes Sandberg, Thomas Voigtmann, Emilie Devijver, and Noel Jakse. Homogeneous nucleation of undercooled Al-Ni melts via a machine-learned interaction potential. Materials Cloud Archive, 10.24435, 2024.

[246] Sébastien Le Roux and Valeri Petkov. Isaacs–interactive structure analysis of amorphous and crystalline systems. Journal of Applied Crystallography, 43(1):181–185, 2010.

[247] W. H. Young. Structural and thermodynamic properties of nfe liquid metals and binary alloys. Reports on Progress in Physics, 55:1769–1853, 1992.

[248] J. P. Hansen and I. R. McDonald. Theory of Simple Liquids. Elsevier, London, 3rd edition, 2006.

[249] Irina V Belova, Tanvir Ahmed, Ujjal Sarder, William Yi Wang, Rafal Kozubski, Zi-Kui Liu, Dirk Holland-Moritz, Andreas Meyer, and Graeme E Murch. Computer simulation of thermodynamic factors in Ni-Al and Cu-Ag liquid alloys. Computational Materials Science, 166:124–135, 2019.

[250] Jürgen Brillo, Alexeij Bytchkov, Ivan Egry, Louis Hennet, Gerhard Mathiak, Irina Pozdnyakova, DL Price, Dominique Thiaudiere, and Didier Zanghi. Local structure in liquid binary al–cu and al–ni alloys. Journal of Non-Crystalline Solids, 352(38-39):4008–4012, 2006.

[251] OS Roik, OV Samsonnikov, VP Kazimirov, VE Sokolskii, and SM Galushko. Medium-range order in al-based liquid binary alloys. Journal of Molecular Liquids, 151(1):42–49, 2010.

[252] N Jakse and A Pasturel. Interplay between structural and atomic transport properties of undercooled Al-Cu binary alloys. AIP Advances, 7(10), 2017.

[253] Sebastian Stüber, Dirk Holland-Moritz, Tobias Unruh, and Andreas Meyer. Ni self-diffusion in refractory Al-Ni melts. Physical Review B—Condensed Matter and Materials Physics, 81(2):024204, 2010.

[254] Iván Egry, Rob Brooks, Dirk Holland-Moritz, Rada Novakovic, Taishi Matsushita, Yuriy Plevachuk, Enrica Ricci, Seshadri Seetharaman, Vasyl Sklyarchuk, and Rainer Wunderlich. Thermophysical properties of liquid Al-Ni alloys. High Temperatures-High Pressures, 38(4):343–351, 2010.

[255] H Okamoto. Al-Ni (aluminum-nickel). Journal of phase equilibria, 14(2):257–259, 1993.

[256] Johannes E Sandberg, Emilie Devijver, Noel Jakse, and Thomas Voigtmann. Adaptive selection of atomic fingerprints for high-dimensional neural network potentials. ML4PS@NeurIPS, 2022.

[257] Emir Kocer, Tsz Wai Ko, and Jörg Behler. Neural network potentials: A concise overview of methods. Annual review of physical chemistry, 73:163–186, 2022.

[258] Kamal Choudhary, Brian DeCost, Chi Chen, Anubhav Jain, Francesca Tavazza, Ryan Cohn, Cheol Woo Park, Alok Choudhary, Ankit Agrawal, Simon JL Billinge, et al. Recent advances and applications of deep learning methods in materials science. npj Computational Materials, 8(1):59, 2022.

[259] Adri CT Van Duin, Siddharth Dasgupta, Francois Lorant, and William A Goddard. Reaxff: a reactive force field for hydrocarbons. The Journal of Physical Chemistry A, 105(41):9396–9409, 2001.

[260] Daniel Marchand, Abhinav Jain, Albert Glensk, and WA Curtin. Machine learning for metallurgy i. a neural-network potential for Al-Cu. Physical review materials, 4(10):103601, 2020.

[261] Daniel Marchand and WA Curtin. Machine learning for metallurgy iv: A neural network potential for Al-Cu-Mg and Al-Cu-Mg-Zn. Physical Review Materials, 6(5):053803, 2022.

[262] Wenwen Li, Yasunobu Ando, Emi Minamitani, and Satoshi Watanabe. Study of li atom diffusion in amorphous li3po4 with neural network potential. The Journal of chemical physics, 147(21), 2017.

[263] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. Computers & Electrical Engineering, 40(1):16–28, 2014.

[264] Atsuto Seko, Akira Takahashi, and Isao Tanaka. First-principles interatomic potentials for ten elemental metals via compressed sensing. Physical Review B, 92(5):054113, 2015.

[265] Luca M Ghiringhelli, Jan Vybiral, Emre Ahmetcik, Runhai Ouyang, Sergey V Levchenko, Claudia Draxl, and Matthias Scheffler. Learning physical descriptors for materials science by compressed sensing. New Journal of Physics, 19(2):023017, 2017.

[266] Runhai Ouyang, Stefano Curtarolo, Emre Ahmetcik, Matthias Scheffler, and Luca M Ghiringhelli. Sisso: A compressed-sensing method for identifying the best low-dimensional descriptor in an immensity of offered candidates. Physical Review Materials, 2(8):083802, 2018.

[267] Xiang Fu, Zhenghao Wu, Wujie Wang, Tian Xie, Sinan Keten, Rafael Gomez-Bombarelli, and Tommi Jaakkola. Forces are not enough: Benchmark and critical evaluation for machine learning force fields with molecular simulations. arXiv preprint arXiv:2210.07237, 2022.

[268] Sina Stocker, Johannes Gasteiger, Florian Becker, Stephan Günnemann, and Johannes T Margraf. How robust are modern graph neural network potentials in long and hot molecular dynamics simulations? Machine Learning: Science and Technology, 3(4):045010, 2022.

[269] Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. Advances in neural information processing systems, 29, 2016.

[270] Tsz Wai Ko and Shyue Ping Ong. Recent advances and outstanding challenges for machine learning interatomic potentials. Nature Computational Science, pages 1–3, 2023.

[271] Tadashi Ogitsu, Eric Schwegler, and Giulia Galli. β-rhombohedral boron: At the crossroads of the chemistry of boron and the physics of frustration. Chemical reviews, 113(5):3425–3449, 2013.

[272] Barbara Albert and Harald Hillebrecht. Boron: elementary challenge for experimenters and theoreticians. Angewandte Chemie International Edition, 48(46):8640–8668, 2009.

[273] Andrew J Schultz and David A Kofke. Comprehensive high-precision high-accuracy equation of state and coexistence properties for classical lennard-jones crystals and low-temperature fluid phases. The Journal of Chemical Physics, 149(20), 2018.

[274] Suplementary Material.

[275] Huaqing Zhang, Jian Wang, Zhanquan Sun, Jacek M Zurada, and Nikhil R Pal. Feature selection for neural networks using group lasso regularization. IEEE Transactions on Knowledge and Data Engineering, 32(4):659–673, 2019.

[276] Jean Feng and Noah Simon. Sparse-input neural networks for high-dimensional non-parametric regression and classification. arXiv preprint arXiv:1711.07592, 2017.

[277] Alexander Goscinski, Guillaume Fraux, Giulio Imbalzano, and Michele Ceriotti. The role of feature space in atomistic learning. Machine Learning: Science and Technology, 2(2):025028, 2021.

[278] Si-Da Huang, Cheng Shang, Pei-Lin Kang, and Zhi-Pan Liu. Atomic structure of boron resolved using machine learning and global sampling. Chemical science, 9(46):8644–8655, 2018.

[279] Thomas B Blank and Steven D Brown. Adaptive, global, extended kalman filters for training feedforward neural networks. Journal of chemometrics, 8(6):391–407, 1994.

[280] Albert S Berahas, Jorge Nocedal, and Martin Takác. A multi-batch l-bfgs method for machine learning. Advances in Neural Information Processing Systems, 29, 2016.

[281] Jason D Lee, Yuekai Sun, and Michael Saunders. Proximal newton-type methods for convex optimization. Advances in Neural Information Processing Systems, 25, 2012.

[282] Samuel Schoenholz and Ekin Dogus Cubuk. Jax md: a framework for differentiable physics. Advances in Neural Information Processing Systems, 33:11428–11441, 2020.

[283] Stefan Doerr, Maciej Majewski, Adrià Pérez, Andreas Kramer, Cecilia Clementi, Frank Noe, Toni Giorgino, and Gianni De Fabritiis. Torchmd: A deep learning framework for molecular simulations. Journal of chemical theory and computation, 17(4):2355–2363, 2021.

[284] Xinyan Wang, Jichen Li, Lan Yang, Feiyang Chen, Yingze Wang, Junhan Chang, Junmin Chen, Wei Feng, Linfeng Zhang, and Kuang Yu. Dmff: an open-source automatic differentiable platform for molecular force field development and molecular dynamics simulation. Journal of Chemical Theory and Computation, 19(17):5897–5909, 2023.

[285] Stephan Thaler and Julija Zavadlav. Learning neural network potentials from experimental data via differentiable trajectory reweighting. Nature communications, 12(1):6884, 2021.

[286] Sebastien Röcken and Julija Zavadlav. Accurate machine learning force fields via experimental and simulation data fusion. npj Computational Materials, 10(1):69, 2024.

Appendix A

Supplementary Material for Chapter 3

A.1 Dataset for the training of the HDNN potential

Tables A.1 and A.2 gather all thermodynamics states that have been simulated by AIMD to generates configuration for the training procedure of the HDNN potential. For each state, after an equilibration at a target temperature a phase space trajectory was produced from which a sample of configurations was randomly extracted to include in the data set. Additional AIMD at $T = 950$ K and $P = 0$ GPa, 1500 K and 2.5 GPa, 2000 K and 28 GPa, and 8000 K and 227 GPa were performed but were not included in the data set for the sake of testing the predictive ability of the HDNN potential. The database contains in total 24300, which amounts to 6 220 800 atoms.

A.2 EAM and MEAM pair-correlation functions

To get a sense of how our HDNNP compares to other widely used potentials, we have run simulations using the Embedded Atom Model (EAM) [156] and Modified Embedded Atom Model (MEAM) [153] potentials both known as performing well in the liquid and solid states (see Ref. [200] and references therein). These simulations were performed identically to the ones used to obtain the pair-correlation functions for our HDNNP, and for the same set of temperatures and densities (see the main text). Figure A.1 shows g(r) obtained from these simulations, along with the corresponding AIMD ones. For the sake of clarity, the results of the HDNN potential are not shown since they match very closely the AIMD curves. It is worth mentioning that above 20 GPa we were not able to achieve MD simulations with the EAM potential. At ambient pressure, both EAM and MEAM reproduce well the AIMD simulations even if noticeable can bee seen. At larger pressures and temperature the agreement worsen showing that the EAM is not transferable while the MEAM still gives reasonable results.

he mean square error between the two were calculated, as shown in table A.3. In this table, are given the p-values obtained by performing a t-test statistics between the square errors $(g_{\mathrm{AIMD}}(r) - g_{\mathrm{NNP}}(r))^2$ and $(g_{\mathrm{AIMD}}(r) - g_{\mathrm{emp.}}(r))^2$, treating the error at different radii

Structure	T (K)	P (GPa)	Trajectory (ps)	Sample size
fcc	10	0	40	1000
fcc	300	0	40	1000
fcc	400	0	40	1000
fcc	500	0	40	1000
fcc	600	0	40	1000
fcc	700	0	40	1000
fcc	800	0	40	1000
fcc	10	1	10	100
fcc	10	10	10	100
fcc	10	100	10	100
hcp	10	0	10	100
hcp	10	10	10	100
hcp	10	100	10	100
hcp	10	200	10	100
hcp	10	300	10	100
bcc	10	0	10	100
bcc	10	10	10	100
bcc	10	100	10	100
bcc	10	200	10	100
bcc	10	300	10	100

Table A.1: Characteristics of the data set built from AIMD simulations. Are given the structure of the simulation (fcc, hcp, bcc, and liquid), the temperatiure T, the pressure P, the time span of the AIMD trajectory from which the configuration are sampled, the sample size, namely the number of configurations randomly extracted from the trajectory. The pressures less that 1 GPa were indicated as 0.

as independent.

Structure	T (K)	P (GPa)	Trajectory (ps)	Sample size
Liquid	500	0	40	1000
Liquid	600	0	40	1000
Liquid	650	0	40	1000
Liquid	700	0	40	1000
Liquid	750	0	40	1000
Liquid	800	0	40	1000
Liquid	1000	0	40	1000
Liquid	1100	0	40	1000
liquid	1250	0	40	1000
liquid	1350	0	40	1000
liquid	1500	0	40	1000
liquid	1600	0	40	1000
liquid	1700	0	40	1000
liquid	3100	56	40	1000
liquid	4500	107	40	1000
liquid	8000	320	40	1000

Table A.2: Characteristics of the data set built from AIMD simulations (ontinued). Same caption as Table A.1.

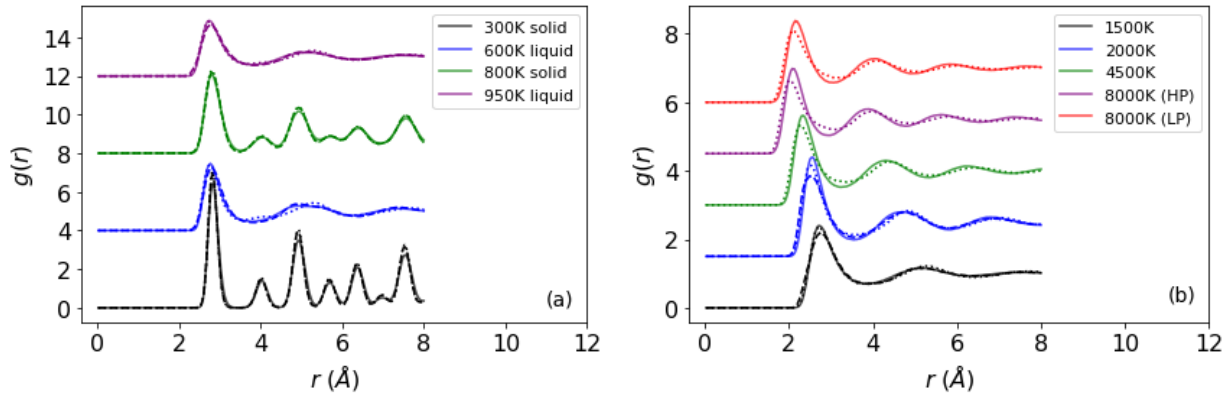

Figure A.1: Pair-correlation functions obtained using the Embedded Atom Model (dashed lines) and Modified Embedded Atom Model (dotted lines) potentials, with solid lines showing the corresponding g(r) obtained from AIMD simulations. Lines of same color correspond to the same temperature and system volume. The plots for $T = 600$, 800, and 950 K are shifted upwards by 4, 8, and 12. Likewise the plots for $T = 2000$, 4500, 8000 K (HP), and 8000 K (LP) are shifted by 1.5, 3, 4.5, and 6. The two 8000 K lines correspond to high (360 GPa) and low (250 GPa) pressure.

T (K)	300	600	800	950	
MSE (EAM)	0.0297	0.0041	0.0048	0.0032	
MSE (MEAM)	0.036	0.0065	0.0064	0.0018	
p (EAM)	5.49×10^{-6}	3.59×10^{-4}	4.48×10^{-6}	3.71×10^{-6}	
p (MEAM)	5.48×10^{-7}	2.20×10^{-13}	3.98×10^{-9}	3.78×10^{-10}	

$T(K)$	1500	2000	4500	8000 (H)	8000 (L)
MSE (EAM)	0.0028	0.012	N/A	N/A	N/A
MSE (MEAM)	0.00086	0.066	0.0091	0.015	0.0095
p (EAM)	4.24×10^{-5}	1.51×10^{-6}	N/A	N/A	N/A
p (MEAM)	5.96×10^{-7}	9.63×10^{-16}	1.04×10^{-11}	7.44×10^{-11}	5.92×10^{-11}

Table A.3: Mean-square error of the pair-correlation function $g(r)$ obtained *via* the EAM and MEAM potentials, and evaluated against the one obtained from AIMD, with corresponding p-values (noted here as p) as explained in the text.

Appendix B

Supplementary Material for Chapter 4

B.1 Dataset Composition

Figure B.1 shows the location in the x_{Ni}-T that were sampled during the construction of the training dataset, as described in the main text.

B.2 Chemical Affinity

Tables B.1 show the primary peak heights for different components of the pair correlation function, at select compositions, calculated using the HDNNP in the main text.

	1795 K			1525 K		
	Ni-Ni	Ni-Al	Al-Al	Ni-Ni	Ni-Al	Al-Al
$Ni_{30}Al_{70}$	1.356	3.436	2.115	1.371	3.750	2.024
NiAl	1.854	3.470	1.905	1.960	3.782	2.024
Ni_3Al	2.513	3.154	1.312	2.699	3.412	1.330

Table B.1: Heights of the first peak of the partial pair correlation functions, for compositions $Ni_{30}Al_{70}$, NiAl, and Ni_3Al, at temperatures 1795 K and 1525 K.

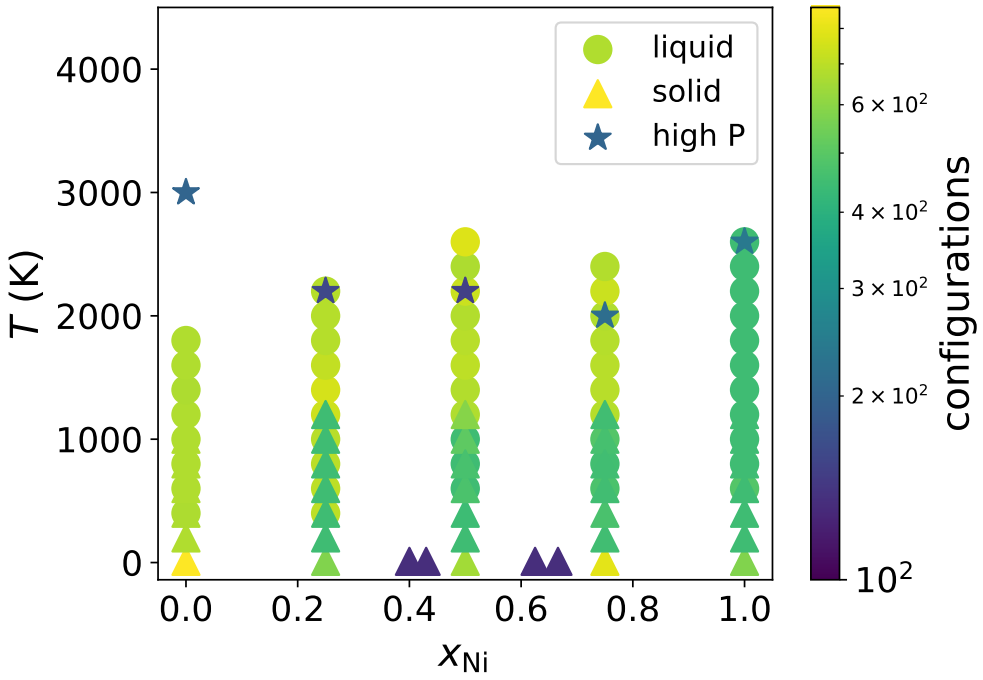

Figure B.1: Composition of the training dataset across the x_{Ni}-T phase diagram, with color indicating the number of configurations sampled at that point. Marker shape indicates whether the sampled AIMD trajectory were from the liquid, respectively solid, branch of simulations at ambient pressure, or from simulation of a high-pressure liquid.

Appendix C

Supplementary Material for Chapter 5

C.1 Symmetry Function Parameters

Tables C.1, C.2 and C.3 contain the symmetry function parameters used for the Lennard Jones system, Al, and B respectively. Definitions of the symmetry functions, and their parameters, are found in the main text.

Table C.1: Parameter values of symmetry functions used for the Lennard Jones system.

index	type	η	Λ	ζ	r_s	r_c	index	type	η	Λ	ζ	r_s	r_c
1	G^2	0.12755	–	–	0.0	2.8	13	G^5	0.12755	−1.0	1.0	0.0	2.8
2	G^2	0.24281	–	–	0.0	2.8	14	G^5	0.12755	1.0	1.0	0.0	2.8
3	G^2	0.46223	–	–	0.0	2.8	15	G^5	0.12755	−1.0	2.0	0.0	2.8
4	G^2	0.87993	–	–	0.0	2.8	16	G^5	0.12755	1.0	2.0	0.0	2.8
5	G^2	1.6751	–	–	0.0	2.8	17	G^5	0.12755	−1.0	4.0	0.0	2.8
6	G^2	3.1888	–	–	0.0	2.8	18	G^5	0.12755	1.0	4.0	0.0	2.8
7	G^2	3.858	–	–	0.5	2.8	19	G^5	0.12755	−1.0	8.0	0.0	2.8
8	G^2	3.858	–	–	0.86	2.8	20	G^5	0.12755	1.0	8.0	0.0	2.8
9	G^2	3.858	–	–	1.22	2.8	21	G^5	0.12755	−1.0	16.0	0.0	2.8
10	G^2	3.858	–	–	1.58	2.8	22	G^5	0.12755	1.0	16.0	0.0	2.8
11	G^2	3.858	–	–	1.94	2.8							
12	G^2	3.858	–	–	2.3	2.8							

C.2 Test Errors

Tables C.4, C.5 and C.6 show the average test errors of the models trained for each featureset discussed in the main text.

Table C.2: Parameter values of symmetry functions used for Al.

index	type	η	Λ	ζ	r_s	r_c	index	type	η	Λ	ζ	r_s	r_c
1	G^2	0.0013	–	–	0.0	6.8	13	G^5	0.0013	−1.0	1.0	0.0	6.8
2	G^2	0.04	–	–	0.0	6.8	14	G^5	0.0013	1.0	1.0	0.0	6.8
3	G^2	0.1	–	–	0.0	6.8	15	G^5	0.0013	−1.0	2.0	0.0	6.8
4	G^2	0.26	–	–	0.0	6.8	16	G^5	0.0013	1.0	2.0	0.0	6.8
5	G^2	0.66	–	–	0.0	6.8	17	G^5	0.0013	−1.0	4.0	0.0	6.8
6	G^2	1.85	–	–	0.0	6.8	18	G^5	0.0013	1.0	4.0	0.0	6.8
7	G^2	1.85	–	–	1.017	6.8	19	G^5	0.0013	−1.0	16.0	0.0	6.8
8	G^2	1.85	–	–	2.035	6.8	20	G^5	0.0013	1.0	16.0	0.0	6.8
9	G^2	1.85	–	–	3.052	6.8	21	G^5	0.0013	−1.0	64.0	0.0	6.8
10	G^2	1.85	–	–	4.07	6.8	22	G^5	0.0013	1.0	64.0	0.0	6.8
11	G^2	1.85	–	–	5.087	6.8							
12	G^2	1.85	–	–	6.105	6.8							

C.3 Smaller featuresets for Al

Figure C.1 shows the results of potentials for Al trained using 7 features selected by AGL, CUR, and PC respectively, as well as with the 7 features selected for LJ (reparametrized for Al).

C.4 Additional Simulation Results

We present in table C.7 the diffusion constants for the LJ system, Al, and B, with different featuresets. Figure C.2 shows the Radial Distribution Function (RDF) and Mean Square Displacement (MSD) for Lennard Jones at $T = 1.5\ k_B/\epsilon$, $\rho = 0.9\ \sigma^{-3}$.

C.5 Fitting with Forces

While the models presented in the main text are fitted only on energies, we have also trained models on forces for B. These models were used the full featureset, with the same architecture and symmetry functions as the other B models. Our best performing such model had a force rmse of $0.702eV/$, and an energy rmse of $71.6meV/atom$. Figure C.3 show the RDF and MSD of this model at 2600 K, along with corresponding for aimd and a model trained using only energies as elsewhere in the article. In the RDF the peak heights appear to be better represented, but at the same time there appears to be introduced a slight shift towards shorter distances. The second minima is likewise badly represented by the model trained on forces. For the MSD the difference is even clearer, with the force-trained model noticeably overestimating the diffusion.

Table C.3: Parameter values of symmetry functions used for B.

index	type	η	Λ	ζ	r_s	r_c	index	type	η	Λ	ζ	r_s	r_c
1	G^2	0.012356	–	–	0.0	5.3	31	G^5	0.12	−1.0	8.0	0.0	5.3
2	G^2	0.033587	–	–	0.0	5.3	32	G^5	0.12	1.0	8.0	0.0	5.3
3	G^2	0.091298	–	–	0.0	5.3	33	G^5	0.12	−1.0	16.0	0.0	5.3
4	G^2	0.24817	–	–	0.0	5.3	34	G^5	0.12	1.0	16.0	0.0	5.3
5	G^2	0.67461	–	–	0.0	5.3	35	G^5	0.12	−1.0	64.0	0.0	5.3
6	G^2	1.8338	–	–	0.0	5.3	36	G^5	0.12	1.0	64.0	0.0	5.3
7	G^2	3.6675	–	–	0.75714	5.3	37	G^5	0.5	−1.0	1.0	0.0	5.3
8	G^2	3.6675	–	–	1.5143	5.3	38	G^5	0.5	1.0	1.0	0.0	5.3
9	G^2	3.6675	–	–	2.2714	5.3	39	G^5	0.5	−1.0	2.0	0.0	5.3
10	G^2	3.6675	–	–	3.0286	5.3	40	G^5	0.5	1.0	2.0	0.0	5.3
11	G^2	3.6675	–	–	3.7857	5.3	41	G^5	0.5	−1.0	4.0	0.0	5.3
12	G^2	3.6675	–	–	4.5429	5.3	42	G^5	0.5	1.0	4.0	0.0	5.3
13	G^5	0.01	−1.0	1.0	0.0	5.3	43	G^5	0.5	−1.0	8.0	0.0	5.3
14	G^5	0.01	1.0	1.0	0.0	5.3	44	G^5	0.5	1.0	8.0	0.0	5.3
15	G^5	0.01	−1.0	2.0	0.0	5.3	45	G^5	0.5	−1.0	16.0	0.0	5.3
16	G^5	0.01	1.0	2.0	0.0	5.3	46	G^5	0.5	1.0	16.0	0.0	5.3
17	G^5	0.01	−1.0	4.0	0.0	5.3	47	G^5	0.5	−1.0	64.0	0.0	5.3
18	G^5	0.01	1.0	4.0	0.0	5.3	48	G^5	0.5	1.0	64.0	0.0	5.3
19	G^5	0.01	−1.0	8.0	0.0	5.3	49	G^5	0.5	−1.0	1.0	4.85	5.3
20	G^5	0.01	1.0	8.0	0.0	5.3	50	G^5	0.5	1.0	1.0	4.85	5.3
21	G^5	0.01	−1.0	16.0	0.0	5.3	51	G^5	0.5	−1.0	2.0	4.85	5.3
22	G^5	0.01	1.0	16.0	0.0	5.3	52	G^5	0.5	1.0	2.0	4.85	5.3
23	G^5	0.01	−1.0	64.0	0.0	5.3	53	G^5	0.5	−1.0	4.0	4.85	5.3
24	G^5	0.01	1.0	64.0	0.0	5.3	54	G^5	0.5	1.0	4.0	4.85	5.3
25	G^5	0.12	−1.0	1.0	0.0	5.3	55	G^5	0.5	−1.0	8.0	4.85	5.3
26	G^5	0.12	1.0	1.0	0.0	5.3	56	G^5	0.5	1.0	8.0	4.85	5.3
27	G^5	0.12	−1.0	2.0	0.0	5.3	57	G^5	0.5	−1.0	16.0	4.85	5.3
28	G^5	0.12	1.0	2.0	0.0	5.3	58	G^5	0.5	1.0	16.0	4.85	5.3
29	G^5	0.12	−1.0	4.0	0.0	5.3	59	G^5	0.5	−1.0	64.0	4.85	5.3
30	G^5	0.12	1.0	4.0	0.0	5.3	60	G^5	0.5	1.0	64.0	4.85	5.3

Table C.4: Total number of features (N), number of angular features (N_{G^5}), test error averaged over four models, and computational performance of selected features for LJ, for the original set of features and for the different feature selection methods discussed in the text.

Method	N	N_{G^5}	RMSE ($10^{-3}\epsilon$/atom)	Benchmark (timesteps/s)
-	22	10	9.73 ± 0.08	266.2
AGL	7	1	8.59 ± 0.08	477.2
PC	7	1	8.79 ± 0.07	480.1
CUR	7	2	23.6 ± 0.1	-

Table C.5: Total number of features (N), number of angular features (N_{G^5}), test error averaged over four models, and computational performance of selected features for Al, for the original set of features and for the different feature selection methods with 10 respectively 7 chosen features. The last line shows the results using the feature set chosen by AGL for the LJ training.

method	N	N_{G^5}	RMSE (meV/atom)	Benchmark (timesteps/s)
-	22	10	2.07 ± 0.17	325.9
AGL	10	2	2.29 ± 0.16	611.3
PC	10	1	2.43 ± 0.06	645.6
CUR	10	3	2.44 ± 0.35	422.3
AGL	7	1	2.78 ± 0.07	698.0
PC	7	1	3.06 ± 0.27	592.3
CUR	7	2	6.38 ± 0.10	587.6
AGL (LJ)	7	1	2.67 ± 0.05	701.2

Table C.6: Total number of features (N), number of angular features (N_{G^5}), test error averaged over four models, and computational performance of selected features for B, for different feature selection methods and original set of features. Benchmark performance also shown for *ab initio* calculation.

method	N	N_{G^5}	RMSE (meV/atom)	Benchmark (timesteps/s)
AIMD	-	-	-	0.111
-	60	48	6.12 ± 0.03	84.96
AGL	16	10	7.20 ± 0.02	223.2
PC	16	8	9.54 ± 0.05	-
CUR	16	11	9.45 ± 0.07	232.7

Table C.7: Comparison of diffusion constants D predicted with different sets of N features to *ab initio*, at $T = 1.5\ k_B/\epsilon$ for LJ, 1500 K for Al, and 2600 K for B.

	LJ		Al		B	
method	N	D (σ^2/τ)	N	D (Å^2/s)	N	D (Å^2/s)
AIMD	-	0.0625 ± 0.0054	-	1.6080	-	1.8176
-	22	0.0603 ± 0.0040	22	1.5483 ± 0.0845	60	1.8557 ± 0.0791
AGL	7	0.0631 ± 0.0029	10	1.6431 ± 0.0620	16	1.8732 ± 0.0614
PC	7	0.0633 ± 0.0039	10	1.5843 ± 0.0607	16	-
CUR	7	-	10	1.5813 ± 0.0920	16	1.6981 ± 0.0588

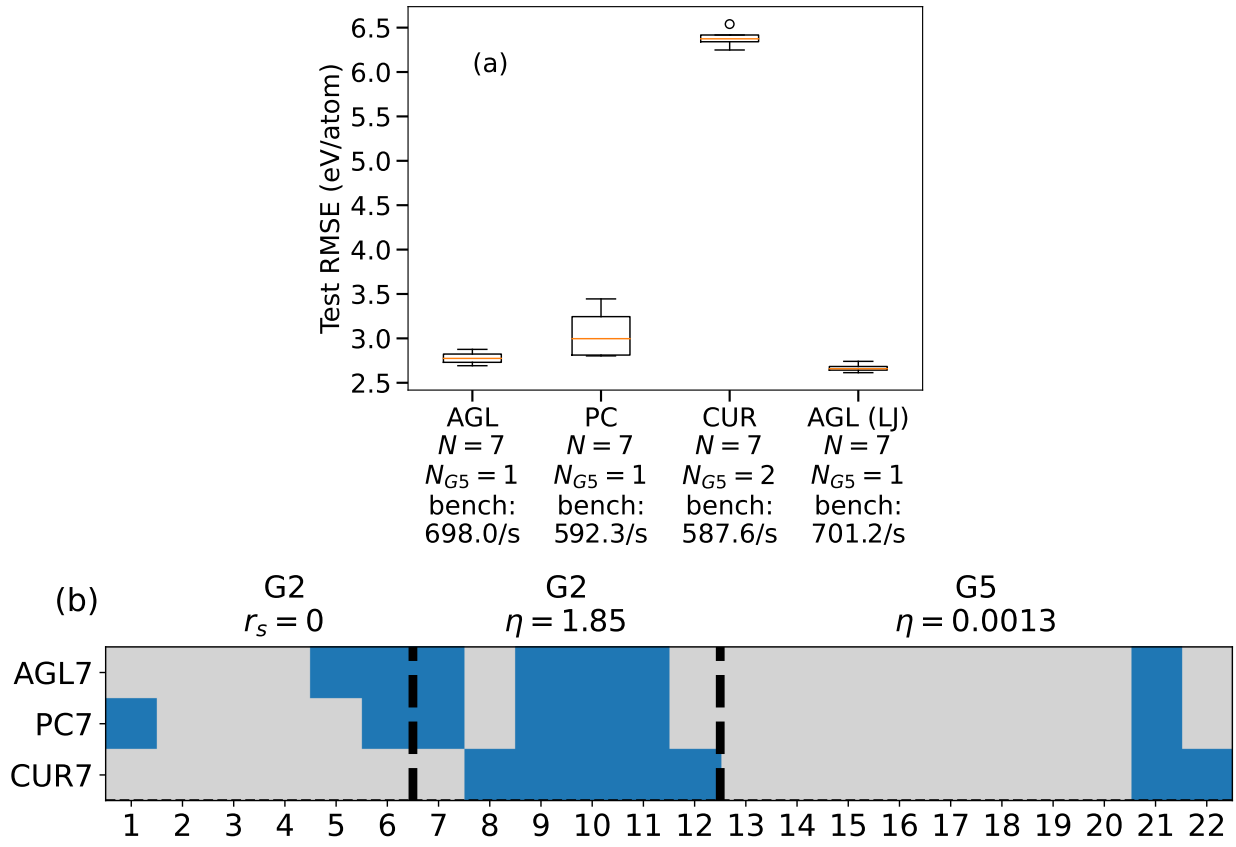

Figure C.1: (a) Box plot of test errors for different feature sets, with total number of features (N), number of angular features (N_{G^5}), and timesteps per second in a benchmark simulation. (b) Matrix plot of features selected for Al, with 7 features selected. Rows correspond to different methods, with discarded features grayed out, and selected ones in blue. The features are grouped into centered G^2, shifted G^2, and G^5.

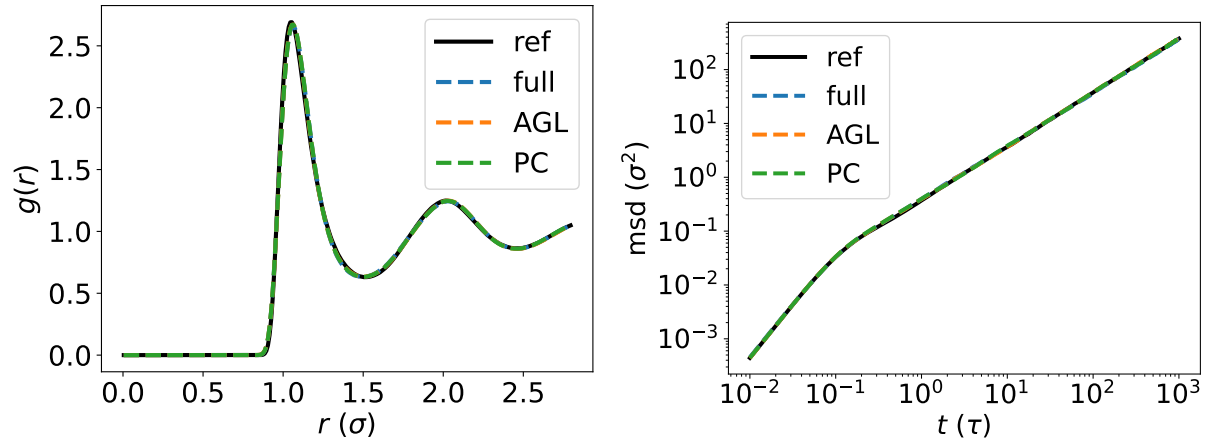

Figure C.2: Comparison of simulation results for HDNNPs with different featuresets against reference potential for Lennard Jones. Left, the radial distribution function. Right, the mean square displacement.

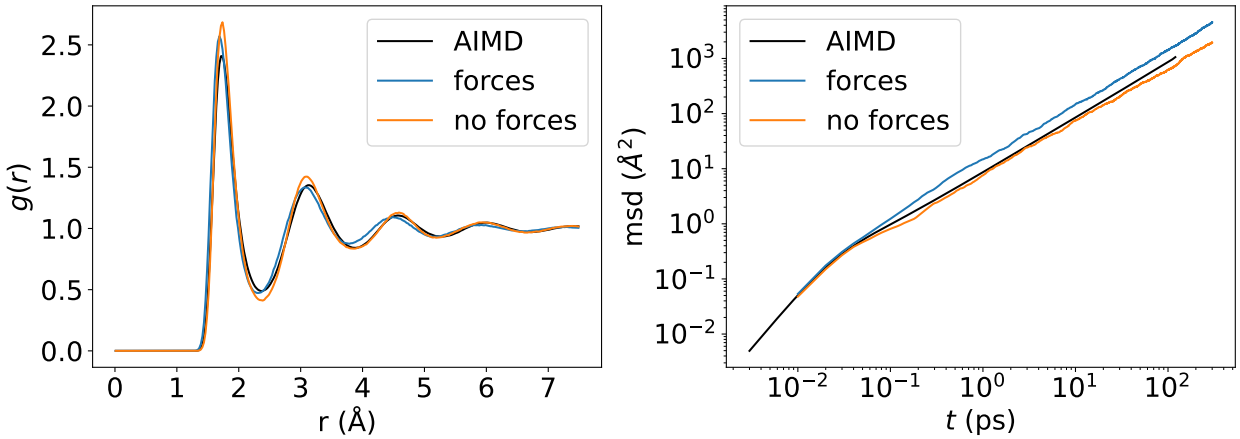

Figure C.3: Simulation results of B models, trained with and without fitting of forces, at 2600 K. *Left:* The radial distribution function. *Right:* The mean square displacement.