Machine Learning for Atomistic Modeling:

Representations and Thermal Transport

vorgelegt von

M.Sc.
Marcel F. Langer
ORCID:0000-0002-1270-3016

an der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades
Doktor der Naturwissenschaften
– Dr. rer. nat. –
genehmigte Dissertation

Promotionsausschuss:

Vorsitzender	Prof. Dr. Wojciech Samek
Gutachter	Prof. Dr. Klaus-Robert Müller
Gutachter	Prof. Dr. Matthias Scheffler
Gutachter	Prof. Dr. Alexandre Tkatchenko
Gutachter	Dr. Matthias Rupp

Tag der wissenschaftlichen Aussprache: 7. Juli 2023

Berlin 2023

Abstract

Computer simulations of molecules and materials are an indispensable tool for physics, chemistry, and materials science. A wide range of methods are available for this task: On one end, first-principles electronic structure approaches, which numerically solve the Schrödinger equation, obtain high accuracy at high computational cost. On the other end, forcefields, simple analytical approximations, are fast to evaluate, but require parametrisation and an explicit model of desired physical interactions. Machine learning is increasingly used to bridge the gap between these two extremes, aiming to combine high accuracy with computational efficiency. To this end, regression models are trained on quantum-mechanical reference calculations and then used as surrogate model during simulations.

This thesis considers two topics related to such models: Representations of atomistic systems, and the application of machine-learning potentials to thermal transport simulations.

Efficient learning in this setting requires models, and therefore input features, that respect fundamental symmetries. We comprehensively review and discuss such representations and relations between them. For selected representations, we compare energy predictions for a range of datasets in numerical experiments controlled for data distribution, regression method, and hyperparameter optimisation.

The Green-Kubo method is a rigorous framework for thermal transport simulations in materials. It is based on equilibrium molecular dynamics simulations, requiring both an accurate description of the potential energy surface and careful consideration of convergence in simulation duration and size. In this context, machine-learning potentials trained with first-principles data promise the ability to reach convergence at a fraction of the computational cost. This thesis adapts the implementation of the Green-Kubo approach to the recently developed class of message-passing machine-learning potentials, which iteratively consider semi-local interactions beyond the initial interaction cutoff. We derive an adapted heat flux formulation for such potentials that can be implemented using automatic differentiation without compromising computational efficiency. The approach is validated by computing the thermal conductivity of zirconia, tin selenide, and silicon with message-passing neural networks.

Zusammenfassung

Computersimulationen von Molekülen und Materialien sind ein wichtiges Werkzeug für Chemie, Physik, und Materialwissenschaft. Es stehen dazu ein Spektrum an Methoden zur Verfügung: Auf der einen Seite stehen quantenmechanische Ansätze, die numerisch die Schrödingergleichung lösen und somit unter großem Rechenaufwand genaue Ergebnisse erzielen können. Auf der anderen Seite stehen einfache analytische Näherungen, sogenannte Kraftfelder, die sehr effizient sind, jedoch für neue Probleme parametrisiert werden müssen, und bei denen physikalische Wechselwirkungen explizit modelliert werden müssen. Methoden aus dem maschinellen Lernen werden zunehmend dazu verwendet, diese Extreme zusammenzuführen und Genauigkeit mit Effizienz zu vereinbaren. Dabei werden Regressionsmodelle auf Referenzrechnungen trainiert und dann als Ersatz für quantenmechanische Rechnungen in Simulationen verwendet.

Diese Arbeit setzt sich mit der Konstruktion und Anwendung solcher Modelle auseinander und erforscht dabei zwei verschiedene Themen: Die Darstellung von Molekülen und Materialien für maschinelles Lernen, und die Anwendung von tiefen neuronalen Netzwerken auf die Simulation von Wärmetransport.

Für das effiziente maschinelle Lernen in diesem Zusammenhang werden Modelle benötigt, bei denen grundlegende physikalische Symmetrien berücksichtigt werden. Von entscheidender Bedeutung ist dabei die angemessene Darstellung von atomistischen Systemen, auf deren Basis die Regression stattfindet. Wir fassen bestehende Methoden für die Konstruktion solcher Darstellungen zusammen, und zeigen dabei Zusammenhänge und gemeinsame Konstruktionsprinzipien auf. Ausgewählte Methoden werden in numerischen Experimenten auf verschiedenen Datensätzen verglichen. Dabei werden andere Faktoren wie die statistische Verteilung der Daten, die Regressionsmethode, oder die Optimierung von Hyper-Parametern gleich gehalten.

Die Green-Kubo-Methode ist ein Ansatz für die Simulation von thermischer Leitfähigkeit. Sie basiert auf Molekulardynamik, und benötigt daher sowohl ein genaues Modell der Potentialoberfläche als auch eine genaue Betrachtung der Konvergenz bezüglich der Simulationsgröße und -dauer. In diesem Zusammenhang versprechen Kraftfelder, die auf quantenmechanischen Referenzrechnungen trainiert worden sind, konvergierte Ergebnisse in einem Bruchteil der Rechenzeit zu liefern. In dieser Arbeit wird die Implementierung der Green-Kubo-Methode auf Kraftfelder erweitert die durch eine vor kurzem entwickelte Klasse von neuronalen Netzwerken, sogenannte Message-Passing-Netzwerke, beschrieben werden. In solchen Kraftfeldern können Wechselwirkungen über die lokalen Nachbarschaften von Atomen hinausreichen. Wir leiten eine passende Formulierung des Wärmestroms her welche mit automatischer Differenzierung effizient implementiert werden kann. Diese Methode wird dann für die Vorhersage der thermischen Leitfähigkeit von Zirconiumdioxid, Zinnselenid, und Silizium angewendet.

Acknowledgements

Like most things worth doing, this thesis was not a solitary effort.

First of all, I would like to thank Matthias Rupp, who replied to my first email asking about quantum mechanics and machine learning and has not stopped answering since. Equally big thanks belong to Matthias Scheffler and Klaus-Robert Müller, who welcomed me into their groups and supported and guided me in my work. I am very grateful for the opportunity to work in such excellent environments, and to benefit so greatly from their resources and opportunities. Particular thanks is also due to the Institute for Pure and Applied Mathematics, where many main ideas in this thesis were first developed.

This work would have been impossible without my co-authors, coworkers, and colleagues. The first is Florian Knoop, who taught me almost everything I know about thermal transport, and who never turned down a 'small beer'. Thanks also to Christian Carbogno, who taught me the rest. In no particular order, I would like to thank other colleagues who have answered my questions, indulged mad ideas, and have otherwise contributed to an overall fantastic working environment: Mihail Bogojeski, Stefan Chmiela, Lucas Deecke, Thorben Frank, Adrian Hill, Sebastian Kokott, Hagen-Henrik Kowalski, Marcin Krynski, Yair Litman, Dmitrii Maksimov (and friends), Niklas Schmitz, Daniel Speckhart, Chris Sutton, Thomas Purcell, the regulars of the `ase` bar, and many more who would not fit on the page.

Many thanks are also due to administrative staff, in particular Julia Pach, Hanna Krauter, Rayya Douedari, Andrea Gerdes, and Jana Peich, who have been an invaluable help in navigating bureaucracy and getting things done, often at the very last minute.

Outside of the office, I thank my friends in Berlin, and all the other places, physical and virtual, for making these last few years the best ones yet, and for making this city my home. Without your support, and a reminder of life outside work, this would have been impossible.

At this final chapter of my formal education, I would like to thank those that started it: Annegret Rehmet, who first taught me about learning, Peter Meggle, who showed me that computers and mathematics go hand in hand, and Benjamin Durst, who restored my faith in physics, and many others who helped along the way.

I would also like to acknowledge Keysound Recordings for providing music at a critical moment, and Sven Pfleiderer for donating a spare laptop at the perfect time.

Last, but very much not least, I would like to thank my parents, Heiderose Langer and Matthias Bier, without whom none of this would have been possible. Thank you for everything.

Berlin,
5th June 2023

Contents

for those who help

Chapter 1
Introduction

> But science, he said, had been the wild card, the twist. With … history itself become a slaughterhouse, science had started popping. Not all at once, no one big heroic thing, but there were cleaner, cheaper energy sources, more effective ways to get carbon out of the air, new drugs that did what antibiotics had done before, nanotechnology that was more than just car paint that healed itself … things that made people blink and sit up, but then the rest of it would just go on, deeper into the ditch.
>
> –William Gibson, *The Peripheral*

Computer simulations of atomistic systems are an indispensable tool for chemistry, physics, and materials science. Consequently, a significant fraction of the computational resources available to scientific inquiry today are dedicated to this task, as indicated by figure 1.

Since the earliest days of computing, simple but fast models have been employed to investigate the behaviour of microscopic systems [2–5]. Today, interatomic potentials, or forcefields (FFs), play an important role in the study of dynamical behaviour of materials, molecules, and proteins. In FFs, the quantum-mechanical Born-Oppenheimer (BO) potential energy surface (PES) on which the nuclei move is approximated in terms of interatomic potentials, often inspired by simple physical approximations, which are then parametrised to match reference calculations or experimental data [6–8].

In parallel, the rapid increase of available computational power over the last decades has enabled first-principles quantum-mechanical calculations, for instance with density-functional theory (DFT) [9–11], for ever-larger systems. As a result, high-accuracy ab initio molecular dynamics (aiMD) [12] simulations for hundreds of atoms and tens of picoseconds are now performed routinely on modern high-performance computing (HPC) systems. Nevertheless, the computational cost of first-principles calculations restricts accessible size and length scales, limiting, for instance, the ability to converge thermodynamic observables. Therefore, computationally efficient models such as FFs are still required, but can now make use of increasingly large datasets of reference calculations for parametrisation.

Machine learning (ML) [13] offers an array of methods to approximate functions based on a set of training examples, extending the concept of FFs by doing away with simple functional forms and replacing them with high-dimensional regression models. At the expense of direct physical interpretability, and a lack of built-in physics and chemistry, which may affect and limit extrapolation, such machine-

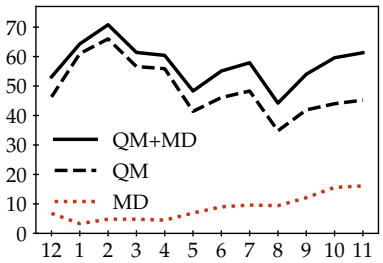


Figure 1: Monthly percentage of the ARCHER2 HPC system used for first-principles (QM) simulations and classical MD codes from December 2021 to November 2022 [1].

Note that an index of abbreviations is provided in the Glossary on page 163.

learning interatomic potentials (MLIPs) [14–20] offer the advantage of flexibility, and the ability, at least in principle, to systematically improve by adding more reference data, combining computational efficiency[1] with high accuracy.

In addition to learning a single PES, ML approaches have also been used to approximate general structure-property mappings across chemical space [21, 22]. Beyond such surrogate modeling approaches, ML can also be integrated more deeply into electronic structure calculations, for instance via learned exchange-correlation functionals [23, 24] or wave function methods based on neural networks (NNs) [25]. Many other applications of ML in computational quantum physics, chemistry, and biology have been considered [26–30].

[1] For instance, chapter 5 of this thesis considers MLIPs that are three to four orders of magnitude faster to evaluate than DFT for structures of similar size.

THE EFFECTIVE USE OF ML in the context of atomistic modeling can greatly benefit from an interdisciplinary approach.

While the underlying first-principles methods can, in principle, be treated as black-box processes that generate data, knowledge of their structure can be used to construct data- and compute-efficient models. For instance, as explored in chapter 3, the symmetries of the BO PES can be directly included in model architecture, avoiding the need to infer them from data.

In the other direction, the use of ML models for physical problems often requires an understanding of their architecture. Chapter 4, for instance, discusses message-passing neural networks (MPNNs) [31], models based on convolutional neural networks [32, 33], which originate from the domain of computer vision. When used as MLIPs, they constitute a class of interatomic potentials that exhibit *semi-local* interactions. Such interactions do not appear in standard formulations of other FFs, and adapted formulations of physical quantities are therefore required for their practical use.

Finally, these modeling tasks can take also advantage of the rapid development of computational methods in ML. The rapid proliferation of deep learning [34] has yielded an array of tools and techniques that can be repurposed for such tasks. In particular, the training and use of deep NNs required the development of specialised programming frameworks [35–37] that make use of highly parallel accelerator devices such as graphics processing units (GPUs) and tensor processing units (TPUs), and implement efficient automatic differentiation (AD) [38–40]. Many methods, and in particular MD, require derivatives of the PES. Here, AD can be used to simplify the implementation of FFs without compromising computional efficiency. Chapter 4 discusses how AD can be used to efficiently compute physical quantities of interest for MLIPs in a unified manner.

THIS THESIS DISCUSSES such connections between ML and atomistic modeling for two topics: The construction of *representations* of atomistic systems, i.e., engineering suitable features for ML, and the application of MLIPs based on MPNNs to thermal transport simulations with the Green-Kubo (GK) method.

Description of Chapters

- **Chapter 2 (Foundation)** provides a brief review of foundational topics, establishing the context and notation for the presented work.
- **Chapter 3 (Review and Benchmark of Representations of Molecules and Materials)** discusses representations of atomistic systems. Methods for incorporating physical symmetries, as well as requirements of representations, are discussed, and common techniques are identified. A review of available representations is conducted, and selected methods are benchmarked for energy predictions on a number of datasets in experiments controlled for hyperparameter optimisation, regression method, and dataset distribution. Tradeoffs between computational cost and accuracy are explored. We find that increased body-order leads to higher accuracy, but correspondingly higher computational cost, and that local representations outperform global ones.
- **Chapter 4 (Heat Flux for Semi-Local Machine-Learning Potentials)** considers the application of MLIPs based on MPNNs to thermal transport simulations with the GK method. Performing such simulations requires access to the *heat flux*, a derivative quantity that describes changes in energy density, and is known to be challenging to define and implement for many-body interatomic potentials. We tackle this problem with AD. As a prerequisite, this requires an understanding of the types of potentials described by MPNNs, which admit semi-local interactions beyond local atomic neighbourhoods. A unified perspective on such potentials, which we term graph-based machine-learning potentials (GLPs), is introduced, and used to define and implement the stress. Finally, an adapted version of the heat flux, the 'unfolded' heat flux, is derived, which applies to GLPs and can be implemented with AD without compromising asymptotic linear scaling of computational cost.
- **Chapter 5 (Thermal Conductivity with Message-Passing Neural Networks)** applies the methods developed in chapter 4 to computing the thermal conductivity of selected materials, first verifying the heat flux formulations, and then exploring the feasibility of GK calculations with MPNNs. First, thermal conductivity for zirconia is computed across temperatures (300 K to 1800 K) and phases (monoclinic, tetragonal) with a SchNet [41, 42] GLP trained on aiMD trajectories. Results are in good agreement with another MLIP, as well as experiments. However, the potential is observed to break down at elevated temperatures. Finally, materials with low thermal conductivity, tin selenide at 300 K, and high thermal conductivity, silicon at 400 K, are investigated with So3krates [43] GLPs. For silicon, simulation size convergence is identified as a key issue, which limits the presented approach to materials with low thermal conductivity.
- **Chapter 6 (Conclusion)** provides a summary of this thesis, discusses open questions and avenues for future work, and finally offers a broader perspective on future developments.

Main Contributions

- **Review and benchmark of representations**: We survey common techniques to incorporate physical symmetries into representations of atomistic systems, and provide an extensive review of available representations. A benchmark of three selected representations on three datasets is conducted, controlling for other factors, to isolate the impact of the representation on predictive accuracy.
- **Heat flux for message-passing machine-learning potentials**: A formulation of the heat flux suitable for interatomic potentials with semi-local interactions is derived, which includes MLIPs based on MPNNs. This formulation can be implemented efficiently with AD, and therefore enables the practical use of MPNNs for GK simulations for the first time.
- **Application of message-passing neural networks to predictions of thermal conductivity**: We apply two different MPNN architectures to thermal transport simulations in three different materials, verifying the heat flux formulation introduced in this thesis. Challenges for future applications are outlined, and convergence is investigated in detail.
- **Understanding graph-based interatomic potentials**: We discuss how interatomic potentials that use atomic neighbourhoods and are based on atom-pair vectors can be viewed as acting on a graph representation of atomistic systems. In this unified perspective, which includes finite-distance terms of classic FFs, common MLIPs, and MLIPs based on MPNNs, local and semi-local interactions can be distinguished, and we discuss how common definitions of derivative quantities such as pairwise forces and atomic stress are affected by this distinction. This framework allows a unified AD-based implementation of forces, stress and heat flux, which we provide, and yields the conceptual basis for the discussion of the heat flux.

Publications

Much of the presented work has been submitted to be published. The manuscripts are listed below:

"Representations of molecules and materials for interpolation of quantum-mechanical simulations via machine learning,"
by Marcel F. Langer, Alex Goeßmann, and Matthias Rupp
in npj Computational Materials **8**, 41 (2022)
doi:10.1038/s41524-022-00721-x
Referenced as [44].

"Heat flux for semilocal machine-learning potentials,"
by Marcel F. Langer, Florian Knoop, Christian Carbogno, Matthias Scheffler, and Matthias Rupp
in Physical Review B *in press*
arXiv:2303.14434
Referenced as [45].

"Stress and heat flux via automatic differentiation,"
by Marcel F. Langer, J. Thorben Frank, and Florian Knoop
in revision
arXiv:2305.01401
Referenced as [46].

I thank my co-authors for graciously allowing me to include text and figures from these works in this thesis.

Data and Code Availability

All code and data used to produce the results presented in this thesis, as well as the document itself, are available at or referenced in doi:10.5281/zenodo.7963152 and https://github.com/sirmarcel/doctor-archive.

Data sources for individual chapters are additionally highlighted in their respective introductions. A brief overview of software developed for this work is given in appendix E.

Additional information, for instance future versions of this thesis, can be found at https://marcel.science/doctor.

Readers are heartily encouraged to contact mail@marcel.science with any questions or comments.

This is the official accepted version of this thesis, designated v2.0, compiled on 31st August 2023, and deposited under doi:10.14279/depositonce-18647.

Chapter 2
Foundations

> You got to build your ***foundations***!
>
> – overheard in Manchester pub

This chapter sets the stage for the work presented in chapters 3 to 5 by providing an overview of foundational concepts and introducing relevant terminology and notation.

In section 2.1, we begin by discussing quantum mechanics, and the particular task of solving the electronic Schrödinger equation. Computational approaches to this task are briefly discussed, with a particular focus on DFT. We conclude with the introduction of the BO PES. The dynamics of nuclei moving on this PES can be modelled with MD, the focus of section 2.2. Approaches to modeling systems at different thermodynamic conditions are introduced, and the usage of approximate PES such as FFs and MLIPs is motivated. Section 2.3 then explores how bulk properties can be obtained from simulations with a finite number of independent atoms through periodicity. Section 2.4 introduces the concepts and terminology of ML applied to atomistic modeling, as well as providing an overview of the techniques and types of applications relevant for this thesis. Finally, section 2.5 introduces the GK method for the calculation of thermal conductivity.

Section 2.1
Quantum Mechanics

> *"At the moment," the mathematics announced, "I'm solving Schrödinger's equation on a grid of ten spatial and four temporal dimensions. No one else can do that."*
>
> – M. John Harrison, *Light*

This section reviews select concepts of quantum mechanics (QM), and the approximations employed for this thesis. First, we briefly introduce the mathematical formulation of QM, and the particular problem of describing an atomistic system of nuclei and electrons. Then, strategies for solving this computationally challenging problem, even when reduced to electrons, are discussed. Armed with an approximate solution to the electronic Schrödinger equation, we then introduce the Born-Oppenheimer (BO) potential energy surface (PES).

Full introductions to quantum mechanics can be found in many textbooks, for instance Dirac [47] and Napolitano and Sakurai [48]. An excellent overview of quantum chemistry can be found in reference [49], and a more detailed account of DFT in reference [50].

2.1.1 Quantum Mechanics and Quantum Chemistry

To date, QM provides the most accurate and well-tested description of the behaviour of matter at the atomic scale. Its formal machinery, developed from the 1920s onwards, is based on abstract state vectors, or *kets*, $|\phi\rangle$ in a Hilbert space that describes the overall state space of the quantum-mechanical system.

Linear operators $\hat{O}$ act on kets in that space; self-adjoint operators[1] correspond to physical observables. The dynamics of a quantum system are governed by the time-dependent *Schrödinger equation*

$$\hat{H}\,|\phi;t\rangle = i\hbar\frac{\partial}{\partial t}\,|\phi;t\rangle\ , \tag{2.1.1}$$

where $\hat{H}$, the *Hamiltonian*, evaluates the total energy. If $\hat{H}$ is not dependent on time, which is the case considered in this thesis, the time-evolution of any initial state $|\phi;t_0\rangle$ is given by its expansion in the eigenstates of $\hat{H}$.[2] It is therefore sufficient to solve the time-independent Schrödinger equation, which poses an eigenvalue problem

$$\hat{H}\,|\phi\rangle = E\,|\phi\rangle\ , \tag{2.1.2}$$

where E is the total energy of the quantum system.

For the systems of concern in this thesis, which are composed of N atomic nuclei[3] with charges Z_i and masses m_i, as well as n electrons,

[1] In a finite-dimensional complex vector space, this is equivalent to the matrix representation of the operator being Hermitian, i.e., equivalent to its conjugate transpose.

[2] More precisely, letting a label eigenstates of an operator that commutes with $\hat{H}$, with corresponding energy eigenvalues E_a, the state at time t is given by $\sum_a \exp\left(-iE_a(t-t_0)/\hbar\right)|a\rangle\,\langle a|\phi;t_0\rangle$: The dynamics are fully determined by the solution of the time-independent Schrödinger equation. A full exposition of quantum dynamics can be found in reference [48, ch. 2].

[3] We do not consider the internal structure of nuclei.

which all share the mass m^e, the Hamiltonian in atomic units is

$$\hat{H} = \hat{T}_\mathrm{el} + \hat{T}_\mathrm{nuc} + \hat{H}_\mathrm{el\text{-}el} + \hat{H}_\mathrm{nuc\text{-}nuc} + \hat{H}_\mathrm{el\text{-}nuc} \tag{2.1.3}$$

$$= -\frac{1}{2}\sum_{p=1}^{n} \hat{\boldsymbol{p}}_p^\mathrm{e} - \frac{1}{2}\sum_{i=1}^{N} \frac{m^\mathrm{e}}{m_i} \hat{\boldsymbol{p}}_i \tag{2.1.4}$$

$$+ \sum_{p=1}^{n}\sum_{q<p} \frac{1}{|\hat{\boldsymbol{r}}_p^\mathrm{e} - \hat{\boldsymbol{r}}_q^\mathrm{e}|} + \sum_{i=1}^{N}\sum_{j<i} \frac{Z_i Z_j}{|\hat{\boldsymbol{r}}_i - \hat{\boldsymbol{r}}_j|} - \sum_{p=1}^{n}\sum_{i=1}^{N} \frac{Z_i}{|\hat{\boldsymbol{r}}_i - \hat{\boldsymbol{r}}_p^\mathrm{e}|}\,, \tag{2.1.5}$$

with $\hat{\boldsymbol{p}}$ denoting momentum and $\hat{\boldsymbol{r}}$ position operators, and a superscript e indicating electronic quantities. Solving this problem is the main task of quantum chemistry.

We note that this Hamiltonian is non-relativistic, and contains no spin-dependent terms. However, relativistic effects cannot be simply discarded; approximations are employed in practice [51–53].

In the context of this section, spin enters through the spin-statistics theorem [54], which places additional constraints on the solutions: Wave functions, i.e., eigenkets in position representation, must be anti-symmetric.

As a first step towards solving this problem, we separate the electronic states from the nuclear ones, motivated by the appearance of the mass ratio $m^\mathrm{e}/m_i \approx 10^{-4} - 10^{-6}$ in $\hat{T}_\mathrm{nuc}$. This can be achieved by expanding the full solution of $\hat{H}$ in terms of solutions $|\phi_a\rangle$ of the *electronic* Hamiltonian

$$\hat{H}_\mathrm{el} = \hat{T}_\mathrm{el} + \hat{H}_\mathrm{el\text{-}el} + \hat{H}_\mathrm{el\text{-}nuc}\,, \tag{2.1.6}$$

describing the electrons in an external potential $\hat{H}_\mathrm{el\text{-}nuc}$ which parametrically depends on atomic positions. Neglecting terms of the order $(m^\mathrm{e}/m_i)^{1/4}$, which describe the coupling of electronic states due to the nuclei, leads to a set of separate nuclear Schrödinger equations enumerated by the electronic states a with energy E_a:

$$(\hat{T}_\mathrm{nuc} + \hat{H}_\mathrm{nuc\text{-}nuc} + E_a)\,|\chi\rangle = E\,|\chi\rangle\ . \tag{2.1.7}$$

This approximation is known as the Born-Oppenheimer (BO) approximation [55]. Intuitively, we have made the assumption that the response of the electrons to movement of the nuclei is instantaneous.

We now further assume that the electronic ground state (a=0) is sufficiently separated from the first excited state (a=1)[4] such that the electronic system remains in the ground state as atomic positions change. We can therefore identify a single potential energy surface, the BO PES, on which the nuclei move. It will be discussed further in section 2.1.3.

[4] At 300 K, thermal energy is $k_\mathrm{B} \cdot 300\,\mathrm{K} \approx 26\,\mathrm{meV}$, smaller than typical bandgaps for the materials investigated in this work (semiconductors and insulators) [56, ch. 28].

We now turn our attention to the task of obtaining the ground-state solution $|\phi_0\rangle$ of the electronic Schrödinger equation. Despite the simplification of neglecting nuclear degrees of freedom, analytical solutions are unknown for any but the simplest systems. Therefore, we must rely on numerical methods for practical uses.[5]

[5] Such methods are typically called *ab initio* or *first-principles* methods.

Let us briefly consider the difficulty of this task. For an exact solution, we must find the lowest eigenvalue of the electronic Hamiltonian in an appropriate basis, constructed from a complete basis for the individual electron states. Equivalently, we could attempt to directly solve the Schrödinger equation as a partial differential equation in the position basis, constructing a many-body wave function. We could also exploit the Rayleigh-Ritz variational principle,[6] which states that the ground state minimises the energy, i.e.,

$$|\phi_0\rangle = \arg\min_{|\Psi\rangle} \frac{\langle\Psi|\hat{H}_\mathrm{el}|\Psi\rangle}{\langle\Psi|\Psi\rangle}\,. \tag{2.1.8}$$

A complete introduction to Slater determinants and approaches to construct many-body states is given in Szabo and Ostlund [57]. For the remainder of this section, we largely follow the exposition by Foulkes [58].

[6] The variational principle is a standard tool in QM, introduced, for instance in [48, ch. 5.4] or [57, ch. 1.3]. A historical overview can be found in reference [59].

$|\Psi\rangle$ is a trial wavefunction.

In all cases, we face the additional constraint that electrons, being spin-1/2 fermions, cannot occupy the same state due to the Pauli exclusion principle [54]. Therefore, a naive ansatz requires at least $k \geq n$ states $|a = 1...k\rangle$ for each electron,[7] leading to an overall Hilbert space of dimension at least n^n.

Some relief is brought by the realisation that we do not require a description of the individual states of each electron, which are indistinguishable – it is sufficient to consider the $\binom{k}{n}$ ways to distribute n electrons into k available states, or *orbitals*. We term each such selection $\boldsymbol{a}$, a one-hot vector[8] of dimension k indicating which n states are occupied.[9] From this, one can then construct a suitably anti-symmetric state from a linear combination of states $|a_1, a_2, ..., a_k\rangle$ using Slater determinants, which form a complete basis of the anti-symmetric full configuration-interaction (FCI) subspace. We can also note that $\hat{H}_{\text{el}}$ only contains one- and two-electron operators, so we can avoid instantiating the $k^n \times k^n$ Hamiltonian, and instead focus on its matrix elements in the anti-symmetric subspace.

[7] More precisely, since the Hamiltonian does not include spin, $n/2$ would be sufficient, as each state can be occupied by two opposite-spin electrons.

We also note that the single-electron states must be obtained from a single-electron basis first. The Hartree-Fock approach, introduced below, is one way to accomplish this.

[8] Entries are either 0 or 1.

[9] This can be naturally expressed in terms of second quantisation, i.e., the introduction of number states and ladder operators. In the interest of brevity, we do not pursue this here.

If we further restrict ourselves to using a single Slater determinant, in other words, aim to find the single $\boldsymbol{a}$ and corresponding set of orbitals that minimise the energy, we arrive at the Hartree-Fock (HF) method [60, 61]. Working it out in detail is beyond the scope of this thesis; the end result are a set of single-particle equations in an effective potential obtained by averaging over all electrons. The overall model must be solved iteratively in a self-consistent field (SCF) iteration.

Beyond HF, many more sophisticated methods can be constructed, often using its single-electron basis functions as starting point. While such full configuration-interaction (FCI) methods in practice scale exponentially with the number of electrons,[10] it can be used for small systems, or serve as the basis for other methods. Coupled cluster methods, for instance, operate in terms of creation/annihilation operators – adding/removing electrons from a HF ground state – becoming equivalent to FCI if all excitations are considered. However, the expansion can often be truncated at single, double, or triple excitations with reasonable accuracy, yielding a significant reduction in cost compared to FCI. However, computational cost still scales approximately as $O(n^{6-7})$, making it difficult to apply to larger systems.

[10] For instance, using $k = 2n$ single-electron states yields $\binom{2n}{n} \approx 2^{2n}/\sqrt{n}$ for large n via Stirling's approximation.

Alternative approaches that attempt to circumvent the eigenvalue problem of directly solving the Schrödinger equation exist: Variational quantum Monte Carlo approaches stochastically evaluate the right-hand side in equation (2.1.8), and then minimise over some parametrised ansatz for $|\Psi\rangle$. Recently, NN ansatzes have emerged as a promising direction for this type of method [25]. Diffusion Monte Carlo methods take a different approach, noting that $\exp(-\tau\hat{H})$ acting one a ket $|\Psi\rangle$ will yield the ground state in the limit of $\tau \to \infty$. One can then re-interpret the time-evolution as a stochastic diffusion-and-branching process of classical particles, which can be simulated numerically. However, the fermionic nature of electrons causes additional difficulty, making further approximations necessary.

2.1.2 Density-Functional Theory

We have seen that the many-body nature of the electronic state, combined with the anti-symmetry constraint, causes great difficulty in the search for the ground state. This motivates the search for a method that avoids dealing with the many-body wave function altogether.

THIS METHOD IS DFT, which descends from the Hohenberg-Kohn theorems [9], stating that *(a)* there is a one-to-one correspondence between the ground state electron density[11] $n_0(\boldsymbol{r})$, the 'external' potential appearing as $\hat{H}_{\text{el-nuc}}$ in equation (2.1.6), and the ground state itself, and *(b)* that the ground state density minimises the energy when phrased as a functional of the density, $E[n]$. This reduces the dimensionality of the problem considerably: Instead of considering a wave function with $3n$ coordinates, we can focus on a scalar field in three dimensions!

However, our problem is not yet solved. Consider the energy functional for a given density n

$$E[n] = \langle \Psi[n] | \hat{T}_{\text{el}} + \hat{H}_{\text{el-el}} | \Psi[n] \rangle + E_{\text{ext}}[n] \tag{2.1.9}$$

where the latter part corresponds to the 'external' potential due to $\hat{H}_{\text{el-nuc}}$, and the former to the 'universal' terms that are independent of external circumstances. While E_{ext} can be straightforwardly phrased in terms of the density,[12] the universal functional retains the full difficulty of the many-body problem of interacting electrons, as it is unknown how to phrase it directly in terms of the density [62, ch. 6.4].

Relief is provided by the Kohn-Sham scheme [10],[13] which introduces an auxiliary system of *non-interacting* electrons[14] with the same ground-state density n. The functional is then split further

$$E[n] = T_{\text{aux}}[n] + E_{\text{aux, es}}[n] + E_{\text{ext}}[n] + E_{\text{xc}}[n]\,, \tag{2.1.10}$$

with T_{aux} describing the kinetic energy of the auxiliary electrons, $E_{\text{aux, es}}$ the electrostatic energy of their charge distribution interacting with itself, and E_{xc} absorbs all the remaining contributions. Minimising this functional with respect to the states of the auxiliary electrons yields a set of single-particle Schrödinger equations, which in turn depend on the density, and must therefore be solved self-consistently until some convergence criterion is reached, similar to the SCF iterations in the HF method.

WITH THIS PROCEDURE, we have now further isolated the difficulty of the many-body problem in the exchange-correlation functional $E_{\text{xc}}[n]$. Treating it exactly would be equivalent to solving the full problem, so approximate exchange-correlation functionals must be used. All known such approximations are uncontrolled; their exact impact on accuracy is unknown and they cannot be systematically improved. Their validity must therefore be checked empirically, comparing to experiment or other, more computationally demanding, methods. Nevertheless, in practice, even simple approximations are often found to

[11] $n_0(\boldsymbol{r}) := \langle \phi_0 | \sum_{p=1}^{n} \delta(\boldsymbol{r} - \hat{\boldsymbol{r}}_p^{\text{e}}) | \phi_0 \rangle$. Computing the density requires evaluating the $3(n-1)$ dimensional integral $n_0(\boldsymbol{r}) = n \int \mathrm{d}^3\boldsymbol{r}_2 \ldots \mathrm{d}^3\boldsymbol{r}_n |\phi_0(\boldsymbol{r}, \boldsymbol{r}_2, \ldots \boldsymbol{r}_n)|^2$; the anti-symmetry of the wave function allows the permutation of arguments such that $\boldsymbol{r}$ can be shifted to the first position.

[12] In the present case, it is simply the electrostatic energy of a charge density in the potential of the nuclei, $-\int \mathrm{d}^3\boldsymbol{r}\, n(\boldsymbol{r}) \sum_i Z_i / |\boldsymbol{r}_i - \boldsymbol{r}|$.

[13] From this point on, DFT will typically refer to Kohn-Sham DFT.

[14] The state of the Kohn-Sham system is also given by a Slater determinant. Interestingly, one can show that for a complete basis, a single such determinant is sufficient to reproduce the exact density. This appears to hold with reasonable accuracy for finite basis sets as well [63].

be sufficient for many cases of interest, yielding a method of overall $O(n^3)$ scaling.[15]

The most straightforward approximation for E_{xc} is the local-density approximation (LDA), which simply takes the density at each spatial point into account. In that case, the general functional of the density at *every* spatial point at once is replaced by an operation acting on the density at each point *separately*, which is then integrated over the simulation domain to yield E_{xc}.[16]

Beyond LDA, improvements can be achieved by adding information of the local *gradient* of the density as well, leading to the generalised gradient approximation (GGA) class of approaches, of which the Perdew, Burke, and Ernzerhof (PBE) [69] functional is one. PBE, and its re-parametrisation for solids, PBEsol [70], will be used in latter parts of this thesis.

More sophisticated exchange-correlation functionals can also be constructed. For instance, hybrid functionals include a portion of HF information. Efforts to integrate ML approaches into DFT have been made [71, 72]. For instance, the DeepMind 21 (DM21) functional [23] predicts the strength of different elements of hybrid functionals (the *enhancement factors*) based on local features.

[15] Variations of DFT with different scaling behaviour exist, for instance linear-scaling formulations [64–66], which exploit locality (see also reference [62, ch. 18]), or DFT with hybrid or double-hybrid funtionals [67], which scale more steeply than cubically.

[16] The mapping between electron density $n(\boldsymbol{r})$ and and energy density $e(\boldsymbol{r})$ is often based on the energy density of the homogeneous electron gas, which can be obtained via diffusion Monte Carlo approaches [68].

2.1.3 Born-Oppenheimer Potential Energy Surface

Let us now assume that we have obtained an approximate ground-state solution of the electronic problem, which we term $|\phi_0\rangle = |\phi[n_0]\rangle$. Its energy eigenvalue is $E_0 = \langle\phi_0|\hat{H}_{\text{el}}|\phi_0\rangle$, which parametrically depends on the positions $\mathcal{R} = \{\, \boldsymbol{r}_i \mid i = 1...N \,\}$ and nuclear charges $\mathcal{Z} = \{\, Z_i \mid i = 1...N \,\}$. Substituting into equation (2.1.7), the Hamiltonian for the nuclei is therefore

$$-\frac{1}{2}\sum_{i=1}^{N}\frac{m^{\text{e}}}{m_i}\hat{\boldsymbol{p}}_i + \sum_{i=1}^{N}\sum_{j<i}\frac{Z_i Z_j}{|\hat{\boldsymbol{r}}_i - \hat{\boldsymbol{r}}_j|} + E_0[\mathcal{R},\mathcal{Z}]\,. \tag{2.1.11}$$

In the classical limit, this Hamiltonian yields the Hamiltonian function in equation (2.2.1), from which the equations of motion for the nuclei can be obtained: They move on a potential energy surface defined by the potential energy

$$U(\mathcal{R},\mathcal{Z}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{\substack{j=1\\ j\neq i}}^{N}\frac{Z_i Z_j}{|\boldsymbol{r}_i - \boldsymbol{r}_j|} + E_0[\mathcal{R},\mathcal{Z}]\,. \tag{2.1.12}$$

The negative derivatives of U with respect to $\boldsymbol{r}_i \in \mathcal{R}$ then correspond to the forces acting on each atom, determining their dynamical behaviour. We will discuss this further in section 2.2.

For now, we must answer a more fundamental question: How do we obtain derivatives of U? Essentially, the answer is contained within equation (2.1.8): The ground state minimises the functional on the right-hand side, and therefore, perturbations, or *variations*, should not

See also reference [48, ch. 5.4]. The variational principle in equation (2.1.8) can alternatively be phrased as finding a state $|\phi_0\rangle$ where the functional on the right-hand side is robust under the variation $|\phi_0\rangle + \delta\,|\phi_0\rangle$.

change the ground state energy; the ground state is a stationary point of that functional. As a consequence, we can write [62, ch. 3.3]:

$$\frac{\mathrm{d}}{\mathrm{d}\lambda}E_0 = \cancel{\langle \frac{\partial\phi_0}{\partial\lambda}|\hat{H}_{\mathrm{el}}|\phi_0\rangle} + \cancel{\langle\phi_0|\hat{H}_{\mathrm{el}}|\frac{\partial\phi_0}{\partial\lambda}\rangle} + \langle\phi_0|\frac{\partial\hat{H}_{\mathrm{el}}}{\partial\lambda}|\phi_0\rangle \tag{2.1.13}$$

$$= \langle\phi_0|\frac{\partial\hat{H}_{\mathrm{el}}}{\partial\lambda}|\phi_0\rangle \tag{2.1.14}$$

where we use λ as any of the parameters entering $\hat{H}_{\mathrm{el}}$. This result is known as the Hellmann-Feynman theorem [73]. It allows us to compute forces for DFT by first computing the analytical derivative of the electron-nuclear Coulomb interaction, and evaluating it with a given density.[17] No derivatives of the density must be taken, and the terms which do not depend on nuclear coordinates do not contribute. However, we have not yet accounted for additional terms arising due to practical implementation details: For instance, if basis functions depend on atomic positions, so-called Pulay forces appear [75]. We also have skipped over complications arising from *approximate* solutions, which break the assumption of stationarity and therefore yield non-vanishing derivatives in the first two terms.

[17] It is interesting to note that this motivates the use of *learned* electronic densities to obtain forces [74].

AT THIS POINT, we have a practical method for obtaining the energy and forces[18] for a given atomistic system. Let us now consider some applications for this ability.

[18] The stress can also be computed, in principle, from the Hellmann-Feynman theorem. In practice, its implementation is rather involved [76].

Section 2.2
Molecular Dynamics

In the previous section, we concluded that, under appropriate assumptions, we can compute the classical[1] potential energy U of an atomistic system and obtain forces $\boldsymbol{F}_i = -\partial U/\partial \boldsymbol{r}_i$ acting on each atom via the Hellmann-Feynman theorem.

We are therefore in a position to apply the machinery of classical mechanics, having successfully isolated quantum mechanical behaviour inside the electronic energy E_0. This section briefly discusses how this leads to molecular dynamics (MD), the practice of simulating the movement of atoms on the BO PES by integrating their equations of motion. We begin with introducing these equations of motion, and then discuss how their numerical integration can be used to investigate thermodynamic properties. We conclude by discussing the use of approximate PES such as FFs and MLIPs.

[1] We note in passing that methods to treat nuclei in a quantum mechanical way exist. In principle, one has to solve the nuclear Schrödinger equation given by equation (2.1.11). In practice, some quantities can be extracted from modified MD approaches. For instance, the quantum mechanical partition function can be computed by executing multiple dynamics simulations in parallel, coupling each replica via a spring. This 'ring polymer' system, in the infinite-replica limit, can then be mapped onto the quantum system. Such a *path-integral* MD treatment can become necessary at low temperatures and for light atoms. [77]

We do not consider this further, but remark that MLIPs offer an intriguing way to manage the high computational cost of such approaches.

2.2.1 Equations of Motion

We can study dynamical behaviour of our system by formulating the classical Hamiltonian from equation (2.1.11) for a given set of positions $\mathcal{R}$ and momenta $\mathcal{P} \coloneqq \{\, \boldsymbol{p}_i \mid i = 1...N \,\}$, which we combine in a phase-space point $\Gamma \coloneqq (\mathcal{R}, \mathcal{P})$. To highlight time-dependence, we write $\Gamma^t =: (\mathcal{R}(t), \mathcal{P}(t)) \coloneqq (\{\, \boldsymbol{r}_i(t) \mid i = 1...N \,\}, \{\, \boldsymbol{p}_i(t) \mid i = 1...N \,\})$.

The classical Hamiltonian is then

$$\mathcal{H}(\Gamma^t) = \sum_{i=1}^{N} \frac{\boldsymbol{p}_i(t)^2}{2m_i} + U(\mathcal{R}(t))\,. \tag{2.2.1}$$

From this Hamiltonian, we can deduce the equations of motion for the phase-space coordinates, arriving at Newton's equations of motion.[2] In practice, as we do not have access to a closed-form solution for $\boldsymbol{F}_i$, these equations of motion must be solved numerically, typically by integrating them from a chosen starting configuration $\Gamma^{t=0}$.

In the present work, we employ the *velocity Verlet* integrator [5],[3] incurring error in the positions and velocities of $O(\Delta t^4)$ per timestep Δt and $O(\Delta t^2)$ over the course of the simulation. The scheme is symplectic[4] and time-reversible, and requires only a single evaluation of forces per timestep. This time-evolution describes an isolated system in the micro-canonical, or NVE ensemble, with fixed number of particles N, volume V, and energy E.

We now drop any explicit dependence on the atomic masses m_i and the charges, as they do not change during dynamics.

[2] The equations of motion are

$$\dot{\boldsymbol{r}}_i(t) = \frac{\partial \mathcal{H}}{\partial \boldsymbol{p}_i(t)} = \boldsymbol{p}_i(t)/m_i$$
$$\dot{\boldsymbol{p}}_i(t) = -\frac{\partial \mathcal{H}}{\partial \boldsymbol{r}_i(t)} = -\frac{\partial U(t)}{\partial \boldsymbol{r}_i(t)} = \boldsymbol{F}_i(t)\,.$$

[3] In particular, $t \to t + \Delta t$ entails [78]

$$\boldsymbol{p}_i(t + \frac{1}{2}\Delta t) = \boldsymbol{p}_i(t) + \frac{1}{2}\Delta t \boldsymbol{F}_i(t)$$
$$\boldsymbol{r}_i(t + \Delta t) = \boldsymbol{r}_i(t) + \Delta t \frac{\boldsymbol{p}_i(t + \frac{1}{2}\Delta t)}{m_i}$$
$$\boldsymbol{F}_i(t + \Delta t) = -\frac{\partial U(\mathcal{R}(t + \Delta t))}{\partial \boldsymbol{r}_i(t + \Delta t)}$$
$$\boldsymbol{p}_i(t + \Delta t) = \boldsymbol{p}_i(t + \frac{1}{2}\Delta t) + \frac{1}{2m_i}\Delta t \boldsymbol{F}_i(t + \Delta t)\,.$$

[4] Loosely speaking, this property ensures that while total energy can fluctuate slightly during the simulation, there exists an auxiliary Hamiltonian for which energy is conserved exactly.

We now consider simulations at other conditions. One such case is the canonical ensemble, NVT, i.e., simulations with a set temperature

rather than a fixed energy. One approach to this problem is to modify the equations of motion with an additional stochastic term.[5] Overall, these modified *Langevin* equations of motion [79] model the system in contact with a thermal bath;[6] it is subject to a *thermostat*.

These equations of motion can be numerically integrated, similar to the NVE case [80, 81]. Many other approaches exist, for instance the Nosé-Hoover family of methods [82–84], which introduce an auxiliary system, or stochastic velocity rescaling [85], which directly modifies the velocities. The latter method aims to overcome shortcomings in previous approaches, such as the lack of a conserved quantity in Langevin dynamics, or the requirement for chains of Nosé-Hoover thermostats to correct lack of ergodicity in some systems.

[5] In particular, we add

$$\dot{\boldsymbol{p}}_i(t) = \boldsymbol{F}_i(t) - \gamma \boldsymbol{p}_i(t) + \sqrt{2k_\mathrm{B}T\gamma m_i}\boldsymbol{\eta}(t)\,,$$

where $\boldsymbol{\eta}(t)$ is a stochastic noise term, k_B the Boltzmann constant, and γ the coupling strength.

[6] Operationally, they describe the force on each particle being augmented with a random 'kick' that injects energy into the system.

Finally, we consider constant pressure and temperature, the isothermal-isobaric NpT ensemble, additionally introducing a *barostat*, which ensures that the system reaches, or remains at, a certain pressure.[7] A simple barostat/thermostat was introduced by Berendsen et al. [86]. In principle, it modifies positions and velocities at each timestep to minimise the deviations relative to the target temperature and pressure.[8] The Berendsen approach yields efficient volume equilibration, but incorrect volume fluctuations and kinetic energy distribution, among other problems [87]. Nevertheless, it remains the only NpT method currently implemented in the `ase` [88] package, which `FHI-vibes` [89] relies on, and it is used in this thesis to generate training data, for which we find this approach to be sufficient.

In cases where such properties must be modeled accurately, alternative, more involved, barostats are used. Volume fluctuations can be modelled with the Bussi-Zykova-Parrinello method [90]. For general, anisotropic, changes in simulation cell shape, stochastic cell rescaling [91, 92], or Martyna-Tobias-Klein [93] and Raiteri-Gale-Bussi [94] approaches can be used.

[7] The pressure in this context is defined in terms of the stress $\sigma = \sigma_\mathrm{kin} + \sigma_\mathrm{pot}$. The former term is simply

$$\sigma_\mathrm{kin}^{\alpha\beta} = \sum_i -1/V\, p_i^\alpha p_i^\beta / m_i\,, \qquad (2.2.2)$$

the contribution of an ideal gas, while the latter is the strain derivative of U. The pressure is then simply $-1/3\,\mathrm{tr}(\sigma)$.

[8] More precisely,

$$\boldsymbol{r}_i(t) \to \boldsymbol{r}_i(t) \cdot \left(1 - \Delta t \lambda_\mathrm{p}(p - p(t))\right)$$
$$\boldsymbol{p}_i(t) \to \boldsymbol{p}_i(t) \cdot \sqrt{1 + \Delta t \lambda_\mathrm{T}(T/T(t) - 1)}\,,$$

with parameters λ. These modifications must then be added to the numerical integration of the equations of motion.

2.2.2 Thermodynamic Ensembles

With the considerations in the previous section, we have gained the ability to numerically simulate an atomic system consistent with external constraints such as a constant energy (NVE), constant temperature (NVT), or constant temperature and pressure (NpT) using MD. Let us now investigate how we can use this ability to compute thermodynamic ensemble averages.

For this, we must first define the notion of a thermodynamic ensemble [78, ch. 2]. For present purposes, it is sufficient to define it as a probability distribution $f(\Gamma)$ that assigns each possible configuration[9] of the system a probability. We consider the following ensembles:

NVE	micro-canonical	$p(\Gamma) \propto \delta(E(\Gamma) - E)$
NVT	canonical	$p(\Gamma) \propto \exp\left(-\beta E(\Gamma)\right)$
NpT	isothermal-isobaric	$p(\Gamma) \propto \exp\left(-\beta(E(\Gamma) + PV)\right)\,,$

where $\beta = 1/(k_\mathrm{B}T)$. In many cases, we wish to evaluate expectation

[9] In statistical mechanics, such a configuration is called a microstate.

The normalisation constants, the partition functions $\mathcal{Z}$, are obtained by integrating over all possible states.

k_B is the Boltzmann constant, converting from temperature to energy.

values of observables $O(\Gamma)$ of the form

$$\langle O \rangle = \int \mathrm{d}\Gamma \, O(\Gamma) p(\Gamma) \,. \tag{2.2.3}$$

MD provides a straightforward way by replacing the integral over Γ with an integral over simulation time, under the assumption of ergodicity.[10] We can then use our ability to numerically simulate the time evolution under a given ensemble, and compute

$$\langle O \rangle = \lim_{t_0 \to \infty} \frac{1}{t_0} \int_0^{t_0} \mathrm{d}\tau \, O(\Gamma^\tau) \,, \tag{2.2.4}$$

where Γ^τ denotes a phase-space point at time τ, after simulation of its time-evolution starting from $\Gamma^{\tau=0}$.

[10] A system is ergodic when the time average and ensemble average coincide. It is often difficult to prove rigorously for a given system; ergodicity is therefore often simply assumed.

2.2.3 Forcefields and Machine-Learning Potentials

So far, we have assumed that $U(\mathcal{R})$ is the BO PES, discussing ab initio molecular dynamics (aiMD) [95].[11] However, the severe computational cost of first-principles methods, even of DFT, limits the size and length scales accessible in aiMD simulations.[12]

Convergence of thermodynamic observables can, however, require large simulations over long timescales (see section 5.2). The size of the simulated system limits the scale of behaviour that can be modelled; for instance, collective excitations become limited in their wavelength by the size of the simulation cell. Some systems, such as biomolecules and proteins, are challenging to model explicitly with first-principles approaches due to their large number of atoms, and the additional requirement of treating solvents.

These considerations motivate the practice of approximating the BO PES with computationally cheaper expressions, so-called *interatomic potentials*[13] or FFs [7]. In fact, the original formulation and application of MD occurred in such a context [5, 96, 97], and it remains the most widely used approach in studies of the dynamics of many-atom systems today. We provide a brief overview of key ideas and methods in the following.

[11] We note in passing that multiple approaches to aiMD exist. In particular, one can perform a new calculation for every timestep, i.e., new SCF iterations in the case of DFT, often using the previous density as initial guess to aid convergence. This approach is sometimes called Born-Oppenheimer MD [95]. Alternatively, in Car-Parrinello MD [12], the electronic system is included in the dynamics simulation itself with an extended Lagrangian, and the electronic system is then kept at a low temperature, ensuring that it approaches, and then remains in, the ground state. In this thesis, aiMD is always performed with the former approach.

[12] For illustration, on a single node on the `raven` HPC system with 72 cores, using the computational settings of section 5.1, a single step of aiMD takes approximately four minutes for 96 atoms. Therefore, ≈ 360 timesteps can be computed in a day. At a typical timestep of 4 fs, 1 ns of simulation time would therefore require approximately two years. Such simulations can be performed with FFs in less than a day.

[13] We will often refer to them as 'potentials' for brevity.

THE PRINCIPLE OF SUCH APPROXIMATIONS is to replace the many-body function $U(\mathcal{R})$, which depends jointly on *all* atomic positions, with a parametrised analytical expression constructed from terms that are computed for subsets of $\mathcal{R}$, such as pairs, triplets, or higher-order combinations. Such a *body-order expansion* will appear again in section 3.2. The parameters of this FF are then optimised to match experimental values or the results of first-principles calculations.

The computational cost of evaluating the resulting expression can be controlled by two main mechanisms: The truncation of interactions based on pairwise distance, which enables linear scaling with system size N,[14] and specialised techniques that make use of the structure of specific interactions to enable favourable computational scaling. For instance, for pairwise Coulomb interactions, where the sum over a

[14] In this thesis, the limit $N \to \infty$ is always taken at constant density. In that case, the average number of neighbours of any atom is independent of the overall size of the system. We then further assume that the potential is defined as some function f acting on all neighbours of a given atom with computational cost scaling as $O(c(N_{\mathrm{nbh}}))$, with some cost function c. Anticipating the introduction of atomic potential energy contributions below, this function must be evaluated N times, so the overall scaling is $O(Nc(N_{\mathrm{nbh}}))$. As N_{nbh} is bounded, overall scaling is $O(N)$. We note that this argument relies on the assumption of a homogeneous system.

periodic system is conditionally convergent [98], techniques such as Ewald summation [99, 100] or fast multipole methods [101, 102] are used to both resolve this issue and enable efficient evaluation.

Additive Pairwise Potentials The first FF used for MD was a pairwise square well potential [96, 97].[15] Experiments with the Lennard-Jones [103] potential followed shortly after. It is defined as

$$U(\mathcal{R}) = \sum_{ij=1}^{N} 2\epsilon \left(\sigma^{12}/r_{ij}^{12} - \sigma^{6}/r_{ij}^{6} \right) , \tag{2.2.5}$$

with two adjustable parameters ϵ and σ; the double sum is often truncated at some cutoff radius[16] r_c. The Lennard-Jones potential serves as a prototypical example of *pairwise additive* potentials, which consist of sums of pairwise terms. Other examples include the Buckingham [104] and Morse [105] potential, as well as the pairwise Coulomb potential, which will be discussed further below. The Lennard-Jones potential is employed even today to approximate Van der Waals interactions, and is used in this thesis as a test case for which analytical derivatives are straightforward to obtain, see section 4.1.

Many-Body Potentials However, in such additive pairwise potentials, all pairs of atoms are treated on an equal footing: Atomic bonding is not modeled, leading to the failure of such *non-bonded* potentials to describe systems with covalent bonds, for instance silicon. For this reason, more complex *bonded* FFs were developed, starting with the embedded atom method (EAM) [106, 107], where pairwise interactions are augmented with a modification through the atomic density at i, and the Stillinger-Weber (SW) [108] potential, which added an angle-dependent triplet term in addition to a pairwise interaction.

The Tersoff potential [109], introduced shortly after, is defined as

$$U = \sum_{ij=1}^{N} U_{ij} \tag{2.2.6}$$

$$U_{ij} = f_{\text{rep.}}(r_{ij}) - b_{ij} f_{\text{attr.}}(r_{ij}) \tag{2.2.7}$$

$$b_{ij} = (1 + \beta^n \zeta_{ij}^n)^{-1/(2n)} \tag{2.2.8}$$

$$\zeta_{ij} = \sum_{k \neq i,j} f_{\text{3-bdy}}(\boldsymbol{r}_{ij}, \boldsymbol{r}_{ik}) . \tag{2.2.9}$$

Here, b_{ij} is a *bond-order* function, $f_{\text{rep.}}$ and $f_{\text{attr.}}$ a repulsive and attractive pair potential, and $f_{\text{3-bdy}}$ a function that takes the angle between $\boldsymbol{r}_{ij}$ and $\boldsymbol{r}_{ik}$ as well as r_{ij} and r_{ik} into account. Other variables are free parameters. In this potential, the interaction between atoms i and j is modified by the overall environment of the pair, or bond, under consideration. This idea was extended in the Brenner [110] potential, modulating interactions based on the bonds of i and j. The general idea of modifying pairwise interactions based on atomic environments will occur again in section 4.1. Since potentials of this type can no longer be decomposed into a sum over pairwise contributions that only depend on $\boldsymbol{r}_{ij}$, they are also referred to as *many-body* potentials.

[15] Defined in reference [97]:

$$V(r) = \begin{cases} \infty & \text{for } r < r_a \\ \text{const.} & \text{for } r_a < r < r_b \\ 0 & \text{otherwise,} \end{cases}$$

with two radii r_a and r_b.

[16] In practice, a cutoff function is also added to ensure continuity. See appendix C.2 for the form used in this work. Alternatively, Ewald summation can be used to treat all-to-all interactions in periodic systems.

Atomic Potential Energy Contributions Even though FFs are naturally described in terms of sums over n-tuples of atoms, it is conceptually useful to recast them in terms of an additive ansatz,

$$U = \sum_{i=1}^{N} U_i \,, \tag{2.2.10}$$

simply collecting terms sharing an index. This gives rise to atomic potential energy contributions U_i, which are, however, not uniquely defined: Contributions can be distributed, or *partitioned*, in different ways without changing U by renaming indices.[17] At present, U_i can each depend on all positions $\mathcal{R}$; for finite-distance interactions,[18] which are studied in this thesis, it reduces to a function of a finite neighbourhood $\mathcal{N}(i)$, defined further in section 2.4.

[17] This freedom of choice will appear again as a gauge freedom of the energy density in section 2.5.

[18] This typically includes bonded terms, and some non-bonded ones. For instance, in this work, we use a truncated Lennard-Jones potential, which would otherwise be classified as non-bonded, but is included here.

Polarisable Potentials and Electrostatics The most straightforward way to model electrostatic interactions is a pairwise Coulomb interaction with fixed charges q located at each atom,

$$U_{ij} \propto \frac{q_i q_j}{|\boldsymbol{r}_{ij}|} \,. \tag{2.2.11}$$

However, such a model cannot account for charge transfer. Therefore, many different schemes for the adjustment of charges during dynamics have been proposed, some of which are surveyed in reference [111]. Beyond point charges located at atoms, descriptions of dipoles and higher-order multipoles can also be included in such *polarisable* FFs.

One commonly used class of methods is charge equilibration [112, 113]. The input parameters to such a scheme are then FF parameters, or can be predicted based on atomic environments [114].

Reactive Forcefields Once bonds are introduced, the modeling of bond breaking and formation, and therefore chemical reactions becomes feasible, leading to *reactive* FFs [8], defined as a combination of non-bonded and bonded terms with electrostatics.

Currently used reactive FFs, such as ReaxFF [115, 116], COMB [117, 118], MEAM [119] and many others surveyed in reference [8],[19] contain a multitude of different terms and corrections. For instance, ReaxFF is defined as [116]

$$U = U_{\text{bond}} + U_{\text{over}} + U_{\text{angle}} + U_{\text{torsion}} \tag{2.2.12}$$

$$+ U_{\text{vdW}} + U_{\text{Coulomb}} \,, \tag{2.2.13}$$

respectively describing bond-related interactions,[20] penalising over-coordination,[21] computing angular and torsional contributions, non-bonded Van der Waals interactions, pairwise Coulomb interactions, and system-specific terms. In the classification from above, the latter two—Van der Waals and Coulomb—terms are non-bonded interactions, the remainder are considered bonded.

[19] In the interest of brevity, we have not discussed FFs focusing on biomolecular simulations, such as AMBER and CHARMM. An overview is given in reference [6].

[20] Similar in concept to the Tersoff potential described above.

[21] Assigning higher energy to unphysical situations where, for instance, a carbon atom with more than four bonds.

Parametrisation Each term in a FF expression depends on a number of parameters, on the order of 100 in the case of ReaxFF. Parametrising

such a FF for a new material is therefore a non-trivial task, requiring global optimisation schemes and a well-curated database of reference results. In the case of ReaxFF, recent efforts have been made to make use of modern ML frameworks with support for AD, allowing the use of gradient descent methods for optimisation [120, 121]. Once parametrisation for a particular material, or class of materials, is complete, FFs often display good transferability and robustness, as many physical mechanisms are modeled explicitly. For instance, nuclear repulsion is typically included directly in the energy expression, avoiding undefined behaviour for configurations where interatomic distances are smaller than the ones seen during training.

Machine-Learning Potentials In parallel to the development of FFs, and in tandem with increasing computational capabilities and the continuing development of first-principles methods, an alternative approach to the design of FFs was developed: The use of general function interpolation, or *regression* (see section 2.4), techniques for the reconstruction of PES based on reference calculations.

Early machine-learning interatomic potential (MLIP) approaches focused on interpolating the PES in between a grid of first-principles calculations, based on the coordinates of the system at hand, and therefore limited to relatively small systems, in line with available first-principles data [122–125]. Representations (see chapter 3) for high-dimensional systems appeared a decade later [15–17], and constitute an active field of study today [18–20].

In such approaches, the explicit modeling of physical interactions, based on chemical insights, intuition, or approximations, which characterises FFs is replaced, or augmented, by fitting a flexible functional form directly to data. This brings advantages and disadvantages: On the one hand, such models can be highly expressive and model mechanisms present in the data, but not known at the time of MLIP construction, and can be systematically improved, at least in principle, by adding data. On the other hand, lacking explicit physical mechanisms can impact transferability and stability in situations not present in the training data. A recent discussion of stability in MD can be found in reference [126]; see also section 2.4.

Many concepts from FF development have been included in MLIPs. For instance, a majority of models[22] are constructed based on the additive ansatz of equation (2.2.10), which will be discussed further in section 2.4.2. Some MLIPs include interaction terms from FFs in their architecture; a paradigmatic example is SpookyNet [130], which was recently used for a large-scale biomolecular simulation [131].

[22] Other approaches are possible, for instance the direct prediction of forces [127–129] in the gradient domain.

While currently available MLIPs are typically less computationally efficient than FFs, efforts have been made to close the gap [132–134]; efforts to include recently developed mechanisms from MLIPs, such as message passing, into FFs have also been made [135]. Further convergence on shared methods is anticipated as a future development.

MLIPs will be discussed further in section 2.4.

Section 2.3
Periodic Systems

> *I will be bold enough to suggest this solution to the ancient problem: The Library is unlimited but periodic. If an eternal traveler should journey in any direction, he would find after untold centuries that the same volumes are repeated in the same disorder – which, repeated, becomes order: the Order. My solitude is cheered by that elegant hope.*
>
> – Jorge Luis Borges, *The Library of Babel*

In the last sections, we have discussed methods to model systems with relatively small numbers of atoms. Despite large-scale benchmarks, for instance with a highly parallel MLIP [133] or DFT on TPUs [136], routine calculations in the context of dynamics are limited in size: Millions of atoms are achievable with fast FFs, tens of thousands to thousands of atoms with MLIPs, hundreds with DFT, and tens or even fewer atoms in higher-level quantum chemistry methods. Objects even on the scale of millimetres, however, contain $\approx 10^{19}$ atoms.[1]

In order to make predictions about such 'bulk' systems regardless, we transition to an alternative description: We now consider the system at hand, which is large relative to the atomistic resolution of the methods we are investigating, as *infinite*. From this infinite bulk, we select a finite region as *simulation cell* and tile it periodically in space. With this step, we have formally reduced the many, or rather infinite, degrees of freedom contained in the bulk to the ones in the simulation cell.[2] Because of this reduction, which entails an approximation of the overall behaviour of the system, the selection of the simulation cell must be carefully checked for convergence in practice. The severity of this approximation depends on the type of system being investigated: Crystalline solids, which exhibit long-range order, and are, barring quantum effects, strictly periodic at 0 K, are naturally modeled in a periodic fashion [56]. Amorphous solids, on the other hand, which are locally ordered but disordered at the larger scale, are challenging to treat in this manner [137]. Bulk liquids and gases can be described periodically in many cases, provided that all relevant interactions can be modeled within the simulation cell [78].

[1] For illustration, elemental Si contains approximately 0.5×10^{23} atoms per cubic centimetre.

[2] To be precise, to the positions in the simulation cell, and the definition of the cell boundaries.

Having motivated the transition to a periodic system, we will now discuss the mathematical, as well as the computational, description of such systems in detail. We begin by introducing notation and

terminology, then discuss how periodic systems are used and implemented, and conclude with some remarks on uniqueness.

2.3.1 Terminology and Notation

We define three lattice vectors $\boldsymbol{b}_a$ and the positions in the simulation cell (s.c.) $\mathcal{R}_{\text{sc}}$. These lattice vectors span a parallelepiped[3] containing the N positions $\mathcal{R}_{\text{sc}}$, which is then periodically tiled to form the infinite bulk system $\mathcal{R}_{\text{all}}$, formed of replicas $\mathcal{R}_{\text{rep}}$ and the simulation cell itself. In summary:

$$
\begin{aligned}
\mathcal{B} &= \{\, \boldsymbol{b}_\alpha \mid \alpha = 1,2,3 \,\} && \text{lattice vectors} \\
\mathcal{R}_{\text{sc}} &= \{\, \boldsymbol{r}_i \mid i = 1...N \,\} && \text{positions in s.c.} \\
\boldsymbol{n} &= \{\, n_\alpha \mid \alpha = 1,2,3 \,\} \in \mathbf{Z}^3 && \text{offset vector} \\
\boldsymbol{r}_{i\boldsymbol{n}} &= \boldsymbol{r}_i + \textstyle\sum_{\alpha=1}^{3} n_\alpha \boldsymbol{b}_\alpha && \text{(replica) position} \\
\mathcal{R}_{\text{rep}} &= \left\{\, \boldsymbol{r}_{i\boldsymbol{n}} \;\middle|\; \boldsymbol{r}_i \in \mathcal{R}_{\text{sc}}, \boldsymbol{n} \in \mathbf{Z}^3, \boldsymbol{n} \neq \mathbf{0} \,\right\} && \text{replica positions} \\
\mathcal{R}_{\text{all}} &= \mathcal{R}_{\text{sc}} \cup \mathcal{R}_{\text{rep}} \\
&= \left\{\, \boldsymbol{r}_{i\boldsymbol{n}} \;\middle|\; \boldsymbol{r}_i \in \mathcal{R}_{\text{sc}}, \boldsymbol{n} \in \mathbf{Z}^3 \,\right\} && \text{all positions}\,.
\end{aligned}
$$

It is often necessary to move between replica positions and those in the simulation cell. Latin indices, for instance j, can refer to positions in both the simulation cell and replicas, and should be taken as shorthand for $j\boldsymbol{n}$, where $j \in \mathcal{R}_{\text{sc}}$ and $\boldsymbol{n}$ identifies the replica that j resides in.[4] If a double index of this form appears, the latin index always refers to a position in the simulation cell.

THE LATTICE VECTORS in $\mathcal{B}$ can be seen as basis vectors for a coordinate system, in which a position $\boldsymbol{r}$ can be written as $\boldsymbol{x}$ such that

$$
\boldsymbol{r} = \sum_\alpha x_\alpha \boldsymbol{b}_\alpha \,. \tag{2.3.1}
$$

Positions in $\mathcal{R}_{\text{sc}}$ are, by definition, given by $x_\alpha \in [0,1]$. The coordinates in this basis, $\boldsymbol{x}$, are called *fractional coordinates*. They can be obtained by computing the inverse of the column matrix obtained from $\mathcal{B}$: Letting $\boldsymbol{B} = [\, \boldsymbol{b}_\alpha \,|\, \alpha = 1,2,3 \,]$ then $\boldsymbol{x} = \boldsymbol{B}^{-1} \cdot \boldsymbol{r}$. The rows of $\boldsymbol{B}^{-1}$ are the *reciprocal lattice* vectors[5] $\mathcal{A} = \{\, \boldsymbol{a}_\alpha \mid \alpha = 1,2,3 \,\}$. Intuitively, the normalised reciprocal lattice vectors $\overline{\boldsymbol{a}}_\alpha$ are the surface normals[6] of the parallelepiped spanned by $\mathcal{B}$, indexed such that $\overline{\boldsymbol{a}}_\alpha$ is orthogonal to the surface spanned by the two $\boldsymbol{b}_\beta$ for which $\beta \neq \alpha$. Therefore, $\overline{\boldsymbol{a}}_\alpha \cdot \boldsymbol{b}_\alpha$ yields the distance between opposite faces of the parallelepiped.

Simulation cells where $\mathcal{B}$ are orthogonal are called *orthorhombic* cells. If all $\boldsymbol{b}$ additionally share the same length, the cell is cubic. The general case, when the orthorhombic cell is distorted into a parallelepiped is called non-orthorhombic or *triclinic*. We note that the coordinate system can always be re-oriented in a standard way that ensures that $\boldsymbol{b}_1$ is aligned with the x-axis and $\boldsymbol{b}_2$ lies in the x-y plane.[7]

THE SMALLEST POSSIBLE CHOICE of simulation cell that still generates the same bulk structure is the *primitive cell*. Many choices are possible,

[3] In principle, other shapes can also be chosen, as long as they can be seamlessly tiled in three dimensions.

We will often use the sets defined here to refer to collections of indices instead of positions. For instance $i \in \mathcal{R}_{\text{sc}}$ in a sum should be taken as indicating a sum over all $\boldsymbol{r}_i \in \mathcal{R}_{\text{sc}}$.

We introduce the following convention: In addition to enumerating cartesian directions, greek indices will also enumerate (lattice) basis vectors. To distinguish from cartesian directions, indices of basis vectors will be lowered.

[4] If $j \in \mathcal{R}_{\text{sc}}$, then $\boldsymbol{n} = \mathbf{0}$.

[5] Multiple definitions exist; in a physics context, an additional factor of 2π is often introduced to aid the discussion of wave vectors.

[6] We follow the convention that $\mathcal{B}$ is chosen such that the resulting coordinate system is right-handed. This ensures that the reciprocal lattice vectors always point into the parallelepiped if they are computed via the standard cross products.

[7] In the convention for $\boldsymbol{B}$ adopted here, this leads to an upper triangular form. The opposite convention, where the *rows* of $\boldsymbol{B}$ are the $\boldsymbol{b}$, yields a lower triangular form. Both conventions are routinely employed in software.

all with the same volume. At present, we always adopt a form that can be written as described above. An alternative construction is the Wigner-Seitz cell, defined as the set of points closest to the origin of a given cell.[8]

A finite number of periodic repetitions of a primitive cell, i.e. restricting $\boldsymbol{n}$ to a finite range, is called a *supercell*. Supercells are often used to initialise MD simulations for crystalline systems, where the primitive cell is given by the fully relaxed structure at 0 K. Once atoms in this *pristine* supercell have moved independently, for instance during a MD timestep, the supercell can no longer be reduced to a repetition of primitive cells, becoming the primitive cell of the resulting system. All cells that can be periodically tiled are called unit cells; we choose the alternative term simulation cell to emphasise that only the positions in that cell are treated explicitly.

[8] More precisely, the volume closest to the Bravais lattice point (see below).

CRYSTALLOGRAPHY IS CONCERNED WITH the systematic study of periodic systems. It describes periodic structures as consisting of a *Bravais lattice*, formed by taking linear combinations of vectors in $\mathcal{B}$ with integer coefficients. Each point on the Bravais lattice $\boldsymbol{R} = n_1\boldsymbol{b}_1 + n_2\boldsymbol{b}_2 + n_3\boldsymbol{b}_3$ is then decorated with a *basis*[9] of atoms, replicas of positions in the unit cell. Bravais lattices can be categorised by the symmetry operations that leave them unchanged,[10] yielding 14 different classes of Bravais lattices. Systematic enumeration of possible basis arrangements yields a total number of 230 distinct crystal structures.[11]

[9] We avoid usage of this term as it conflicts with the perspective that $\mathcal{B}$ yields basis vectors for a coordinate system.

[10] More precisely, we consider the *space group* of the lattice, which contains all operations that leave distances unchanged (*rigid operations*). This group in turn is composed of translations through lattice vectors and operations that leave one lattice point unchanged (*point group*). Combining the 7 possible point groups (or *crystal systems*) with translations yields 14 space groups.

[11] This enumeration proceeds by reducing the symmetry of the basis, starting from the maximally symmetric choice of a single atom. See reference [56, ch. 7].

For a detailed account of these issues, see [56, ch. 8].

2.3.2 Usage

Now that we have introduced the concept of periodicity, we consider how to apply it to solving problems in practice.

Schrödinger Equation In order to apply periodicity to the task of solving the electronic Schrödinger equation – or the Kohn-Sham equations – we must augment the periodic description with suitable boundary conditions. Periodicity alone, via the Bloch theorem [138],[12] yields the information that solutions must take the form $\phi(\boldsymbol{r}) = \exp(i\boldsymbol{k}\boldsymbol{r})u_{\boldsymbol{k}}(\boldsymbol{r})$ where $u_{\boldsymbol{k}}$ is periodic with the lattice,[13] but does not yet tell us how to select $\boldsymbol{k}$. This is achieved by introducing Born-von Karman boundary conditions, demanding that for some *fixed* $\{ n_\alpha \mid \alpha = 1, 2, 3 \}$ the wave function is periodic: $\phi(\boldsymbol{r}) = \phi(\boldsymbol{r} + n_\alpha \boldsymbol{b}_\alpha)$. This yields[14] the restriction that $\boldsymbol{k} = \sum_{\alpha=1}^{3} \boldsymbol{a}_\alpha m_\alpha / n_\alpha$ with $m_\alpha \in \mathbf{Z}$.

[12] Similar results can be shown for periodic Hamiltonians, or general differential equations, giving rise to Floquet theory [139].

[13] $u_{\boldsymbol{k}}(\boldsymbol{r} + \boldsymbol{R}) = u_{\boldsymbol{k}}(\boldsymbol{r})$ for any lattice translation $\boldsymbol{R}$.

[14] Note that we choose a different normalisation of $\boldsymbol{a}$ in this paragraph to avoid factors of 2π.

In other words, $\boldsymbol{k}$ is restricted to two cases: *(a)* a *limited* number of modes where the wavelength exceeds the size of the unit cell, $|m_\alpha| \leq n_\alpha$, and *(b)* many modes where the wavelength is smaller than the unit cell. The former are contained within the *Brillouin zone*, the primitive cell[15] of the reciprocal lattice.[16] The latter $\boldsymbol{k}$ can be decomposed into a component in the Brillouin zone, and a component that can be absorbed into $u_{\boldsymbol{k}}$.

[15] In particular, the Wigner-Seitz cell.

[16] The lattice spanned by $\mathcal{A}$.

The implications of this are profound: In order to model a periodic system extending to a supercell of size n_α in each direction, we can

solve a finite number of separate eigenvalue problems for the different u_k. If the n_α are increased towards infinity, leading to ever-finer sampling of the Brillouin zone, the bulk limit is recovered. In practice, convergence with respect to this *k-grid* must be checked, but often, moderate numbers of k-points are sufficient. Many different strategies for choosing relevant $\boldsymbol{k}$ to sample can be employed, often informed by symmetry [140, 141]. Another implication is that the only periodicity exceeding the unit cell is contained within the phase of ϕ. Therefore, the density is strictly periodic with the lattice, and we can expect the forces, and therefore the BO PES, to be periodic as well.[17]

[17] From symmetry considerations alone, it is clear that once we take $n_\alpha \to \infty$, no physical observable can depend on the choice of *which* replica to choose as simulation cell.

Molecular Dynamics Working with a periodic potential $U(\boldsymbol{r} + \boldsymbol{R}) = U(\boldsymbol{r})$, we can immediately conclude that the force acting on an atom in the simulation cell is the same as the one acting on each of its replicas. Since the time-evolution is determined by the forces,[18] we can restrict MD simulations to the atoms in the simulation cell. We will discuss how this is implemented in practice in section 2.3.3.

[18] In the case of NpT, also the stress, which also does not depend on the choice of which cell is used.

This approach restricts the kinds of dynamics we can observe: Any collective vibrational modes are bounded in wavelength by the simulation cell, as only these N atoms can move independently.

Phonons Collective vibrational modes are called *phonons*. They occur in an alternative approach to modeling lattice dynamics: The BO PES is approximated with a Taylor expansion around the minimum energy configuration. The second-order approximation, corresponding to harmonic springs between atoms in the simulation cell, yields an analytical solution of the equations of motion in terms of plane waves, the phonons.[19] Higher-order corrections introduce scattering between these modes. Once again, the choice of simulation cell – typically a supercell – restricts the possible $\boldsymbol{k}$. Extrapolation schemes to larger supercells have been proposed [143, 144].

A detailed introduction to phonons can be found in reference [142].

[19] Phonons also emerge in the quantum case, where they are commonly described in second quantisation terms and interpreted as bosonic quasiparticles. This leads to the *phonon gas model*, which can be used to describe thermodynamic properties of solids. In section 2.5, we will investigate an approach that goes beyond this model, which breaks down in the case of anharmonic PES, or multiple minima.

2.3.3 Implementation

For this thesis, we must consider periodicity in two contexts: Defining potentials in a way that ensures their periodicity, and implementations of MD for periodic systems.

FOR THE FORMER CASE, we proceed from the fact that the potentials we are considering must respect global translational invariance in addition to periodicity.[20]

[20] After all, we are modeling the BO PES, which shares this invariance. Additional invariances of this form are discussed in a different context in section 3.1.

They are therefore constructed from atom-pair vectors

$$\boldsymbol{r}_{ij} = \boldsymbol{r}_j - \boldsymbol{r}_i \,, \tag{2.3.2}$$

rather than atomic positions directly. We then observe that for any fixed i, the set of atom-pair vectors $\boldsymbol{r}_{ij}$ is identical regardless which cell is chosen for i; the only difference is which replica the j belong to. Therefore, the choice of simulation cell does not impact the set of atom-pair vectors $\left\{ \boldsymbol{r}_{ij} \,\middle|\, i \in \mathcal{R}_{\text{sc}}, j \in \mathcal{R}_{\text{all}} \right\}$ which are the input for the

potential. By construction, any potential of this form is therefore invariant under the choice of simulation cell.

In practice, it is often convenient to rephrase the set of atom-pair vectors in terms of positions in the simulation cell only. This can be achieved by augmenting the definition of atom-pair vectors with the minimum image convention (MIC):

$$|\boldsymbol{r}_{ij}^{\mathrm{mic}}| = \min_{\boldsymbol{n}\in\mathbf{Z}^3} |\boldsymbol{r}_{j\boldsymbol{n}} - \boldsymbol{r}_i| \quad i,j \in \mathcal{R}_{\mathrm{sc}}\,. \tag{2.3.3}$$

This convention states that whenever we compute an atom-pair vector pointing from i to j, we take the one pointing towards the *nearest* replica (or the 'original') of j. In essence, this treats atom-pair vectors as being defined modulo lattice vectors. This introduces discontinuities: Holding $\boldsymbol{r}_i$ fixed, $\boldsymbol{r}_{ij}^{\mathrm{mic}}$ changes direction, but not magnitude, when $\boldsymbol{r}_j$ moves more than half the length of the unit cell away. Additionally, the MIC imposes an implicit cutoff on the interactions contained within the set of atom-pair vectors: Apart from degenerate situations, only a single replica can ever be be considered.

To avoid both issues, we introduce the maximum cutoff radius

$$r_{\mathrm{c}}^{\mathrm{max}} = \min_{\alpha} \frac{|\overline{\boldsymbol{a}}_\alpha \cdot \boldsymbol{b}_\alpha|}{2} \tag{2.3.4}$$

for a given simulation cell. In orthorhombic cells, it reduces simply to being half the length of the smallest lattice vector. If the interaction cutoff of the employed potential is below this value, $r_{\mathrm{c}} < r_{\mathrm{c}}^{\mathrm{max}}$, every atom in $\mathcal{R}_{\mathrm{sc}}$ can only interact with a single replica of each other position; self-interactions are also impossible.

We finally note that the MIC can be implemented using modular arithmetic[21] without explicitly constructing the replicas $\boldsymbol{r}_{j\boldsymbol{n}}$. In some cases, as we will see in sections 4.1 and 4.2, this can lead to difficulties.

[21] For non-orthorhombic systems, it is particularly convenient to first transition to fractional coordinates, where the MIC can be achieved by computing distances modulo 1. However, MIC distances obtained with this method are only correct below $r_{\mathrm{c}}^{\mathrm{max}}$ [145].

In dynamics simulations, we can either let positions evolve freely, relying on the potential to ensure that forces are computed in a periodic fashion, or return (*wrap*) positions whenever they leave the simulation cell. The latter case corresponds to imbuing the simulation cell with the topology of a circle in each coordinate direction,[22] leading to a *toroidal* system.

[22] In the general triclinic case, this is more clear in fractional coordinates: The faces of the unit cube are joined to the opposite side.

This choice can be convenient in practice, as it prevents uncontrolled growth of position vectors, but can also lead to difficulties due to the mismatch between this description and the physical situation of a bulk system composed of a *checkerboard* of replicas. For instance, for the calculation of diffusion constants, which relies on tracking the displacement of each particle from its original position, wrapping positions back leads to unphysical results [146, 147]. Additionally, as atoms close to the boundary can oscillate across it during simulations, wrapping positions at every timestep leads to jumps, which can cause cache misses in neighbourlist computations and therefore incur additional computational cost. In the present work, which investigates systems where no diffusion occurs,[23] we therefore avoid wrapping positions.[24]

[23] In other words, where positions stay bounded over time. In this case, we do not face the difficulty of positions growing over time, which would cause precision issues in long simulations.

[24] See section 4.1.3 for a construction where it is necessary to wrap positions into the simulation cell.

2.3.4 Uniqueness

We conclude by considering some properties of the positions $\mathcal{R}_{\text{sc}}$. For a given bulk system $\mathcal{R}_{\text{all}}$, even disregarding rotation and translation of the overall coordinate system, the positions within $\mathcal{R}_{\text{sc}}$ are not unique. First of all, a given bulk can be equivalently described by the primitive cell or many choices of supercell.[25] Even restricting ourselves to the primitive cell, which fixes the number of atoms N, many equivalent choices can be obtained by *(a)* shifting the origin of $\mathcal{R}_{\text{sc}}$, or *(b)* changing the shape of the cell.

[25] This gives rise to the expectation that potentials are extensive in a simple sense: When the simulation cell is doubled in one direction, we expect, modulo convergence, the energy to double as well. Forces, on the other hand, should remain unchanged. See appendix B.1 for a further discussion.

In addition to cases covered by translational invariance, both operations can lead to some, but not all, atomic positions changing when the cell definition is changed. In other words, they are non-unique, and no physical observable can depend on the arbitrary choices made in defining the simulation cell. This leads to many practical difficulties, for instance the inability to define a consistent dipole moment even for two charged particles in a periodic system: Different choices of boundary can arbitrarily change sign and direction.[26] The quantum-mechanical position operator also becomes ill-defined; its modification for periodic cases is non-trivial [149]. Momenta of distributions in periodic boundary conditions also cannot be uniquely evaluated [148, 150, 151], which will pose an essential difficulty in chapter 4.

[26] The more general case of polarisation is discussed in reference [148].

A straightforward approach to ensure *boundary invariance*, i.e., invariance with respect to the arbitrary choice of simulation cell, is to find a formulation of the desired physical quantity in terms not only of $\mathcal{R}_{\text{sc}}$, but the bulk system. In particular, as we have seen above, a formulation in terms of the set of atom-pair vectors originating from every position in $\mathcal{R}_{\text{sc}}$ is independent of the choice of boundary. We will use this in section 4.2.

Section 2.4
Atomistic Machine Learning

In this section, we briefly review core terminology and introduce notation for ML, discuss applications relevant for this thesis, and introduce the main models we will be using: kernel ridge regression (KRR) in chapter 3 and neural network (NN) regression in chapters 4 and 5. The section concludes with a brief discussions of automatic differentiation (AD), required for the practical and efficient implementation of derivatives of complex models, and of particular importance for chapter 4.

For a detailed introduction to ML and statistical learning, see Hastie et al. [13].

2.4.1 Terminology and Notation

Supervised Learning The task of supervised ML is to obtain an approximation $f : \mathcal{V} \to \mathcal{O}$ of a *ground truth* relationship $t : \mathcal{V} \to \mathcal{O}$ between an input space $\mathcal{V}$ and an output space $\mathcal{O}$ based on a set of n examples, the *training data* or *training set*

$$\mathcal{D}_{\text{train}} = \{ (\boldsymbol{x}_i, t(\boldsymbol{x}_i) = y_i) \mid i = 1...n \} \ . \tag{2.4.1}$$

We keep the definition of $\mathcal{V}$ deliberately vague at this point. For kernel methods, a set is sufficient, for instance. In practice, for computer implementation, we typically work with subsets of $\mathbf{R}^d$ represented as vectors.

In this thesis, we focus on *regression* tasks, where labels are continuous in $\mathcal{O}$. *Classification* deals with a finite number of discrete labels.

At present, we consider a restricted form of this general task, the learning of mappings between atomistic systems $\mathcal{M}$ consisting of atoms $\mathcal{A}$ to properties computed with a first-principles electronic structure method. For simplicity of notation, we focus on scalar properties,[1] $\mathcal{O} \subseteq \mathbf{R}$. In the presence of noise, for instance Gaussian noise sampled from a fixed normal distribution $\epsilon \sim \mathcal{N}$, values of the ground truth function, the *labels*, are observed as $y_i + \epsilon$. In the present setting of approximating numerical QM calculations, noise plays a minor role, but nevertheless emerges later in the context of regularisation.

[1] We can always consider constructing a separate model for each output. Multi-task learning is also possible. Some properties can also be naturally learned together, such as potential energy, forces, and stress.

Loss Functions The quality of the approximation f can be evaluated by computing a *loss* (or error) function $l(y, f(\boldsymbol{x})) \in \mathbf{R}$ for the training data $\mathcal{D}_{\text{train}}$, or *test* data[2] $\mathcal{D}_{\text{test}}$. By convention, loss functions are defined such that lower loss corresponds to a better approximation. Training an ML model therefore amounts to minimising the average of a loss function on $\mathcal{D}_{\text{train}}$. Evaluation consists of computing the loss on unseen data to estimate the degree to which the model can *generalise* outside of the training set. An overview of common loss functions, and in particular those used in this thesis, is given in appendix A.3.

[2] Statistical learning theory often makes the assumption that $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$ are drawn independently from the same distribution, the independent and identically distributed (i.i.d.) assumption. In practice, this cannot always be assured. In chapter 3, efforts are made to approximate it for the purposes of benchmarking. Ensuring that test data is not used during training, not even for the definition of hyper-parameters, is standard practice to ensure that performance estimates are relevant.

Hyperparameters ML models typically possess some parameters that are not determined during training,[3] but must be set in advance to define the overall search space of approximators. Such a parameters are called hyperparameters (HPs). In this work, we distinguish between structural and numerical HPs, where the former includes structural choices, for instance the choices of NN architecture, while the latter refers to parameters that can be varied continuously, for instance regularisation strength in kernel ridge regression (KRR). Different approaches to setting HPs exist; one standard method adopted in this thesis is to optimise HPs by computing the loss for an auxiliary *validation* set that is extracted from $\mathcal{D}_{\text{train}}$ and not used for training. An overview of approaches can be found in [152, ch. 5].

[3] For instance, structural choices and discrete parameters are not amenable to the gradient descent approaches used for NNs. Training is formally understood as an optimisation in some fixed space of functions. Hyperparameters determine that space, or the function to be minimised, while parameters are the result of the minimisation.

Regularisation Expecting to approximate t based on a finite number of examples relies on the assumption of some regularity, or smoothness.[4] Such an assumption can be partially encoded in the ansatz for f, but is often also included in the training process in the form of regularisation. Intuitively, we can imagine an unregularised f attempt to pass through every y in $\mathcal{D}_{\text{train}}$ exactly, while permitting arbitrary behaviour in between. Regularisation aims to penalise such complex models. [154]

[4] This can also be viewed as a question of sampling: Given a signal of finite bandwidth, a finite number of samples is sufficient to reconstruct it [153]; regularisation aims to match the bandwidth of the model to the available training data.

Atomistic Systems An atomistic system, or *configuration* or *structure* is described as a set of N atoms $\mathcal{M} = \{ \mathcal{A}_i \mid i = 1...N \}$, where each atom is associated with a position $\boldsymbol{r}_i$ and atomic properties $\boldsymbol{P}_i$. We typically restrict $\boldsymbol{P}_i$ to the atomic number Z_i. In periodic systems, $\boldsymbol{r}_i$ are taken to be in $\mathcal{R}_{\text{sc}}$ and $\mathcal{M}$ additionally depends on the lattice $\mathcal{B}$. In practice, we therefore describe a system $\mathcal{M}$ as a 3-tuple $(\mathcal{R}, \mathcal{Z}, \mathcal{B})$; $\mathcal{B}$ can be empty for non-periodic systems, $\mathcal{R} = \mathcal{R}_{\text{sc}}$ otherwise. Chapter 3 explores approaches to map such a description to a fixed-size vector, which is more convenient for regression methods.

One core concept in the construction of atomistic ML models is the usage of *atomic environments* $\mathcal{N}(i)$, which is a finite set of atoms in the vicinity of i. Such neighbourhoods can be constructed in different ways, the most common of which is to introduce a cutoff radius r_{c} such that $\mathcal{N}(i) = \left\{ j \;\middle|\; |\boldsymbol{r}_{ij}| \leq r_{\text{c}}, j \in \mathcal{R} \right\}$. We note that the definition of neighbourhoods in periodic systems can lead to difficulties; a common approach is to choose $i, j \in \mathcal{R}_{\text{sc}}$ and imbue $\boldsymbol{r}_{ij}$ with the MIC as discussed in section 2.3. This will be considered further in section 4.1, where we argue that the decomposition into atomic environments naturally leads to a graph description.

LET US NOW briefly consider applications of ML in the context of atomistic modeling, focusing on those investigated in this thesis.

2.4.2 Machine-Learning Potentials

In section 2.2, we already encountered the idea of a machine-learning interatomic potential (MLIP) [14–20], where the regression task is to approximate the mapping between positions $\mathcal{R}$ and total potential energy U based on a training set of first-principles reference calculations.

A tutorial introduction to MLIPs can be found in [155].

An essential consideration for MLIPs is the procurement of training data. If the MLIP encounters configurations for which it cannot make a reasonable prediction, unphysical behaviour and instability in the resulting MD can result.[5] Approaches to mitigating this difficulty, which is inherent in data-driven models, are an active area of research today. Some regression methods, for instance Gaussian process regression (GPR), which we will encounter in section 2.4.4, include an estimate of model uncertainty, which can serve as criterion to add training data during MD. Committee models can also be used [155–157]; many other approaches and variations have been suggested [158–160]. Since an ideal uncertainty estimate would be equivalent to making perfect predictions, approximate solutions must be used in practice, and verified empirically. Recent approaches that use uncertainty estimates to refine the PES on-the-fly during MD simulations [161] have yielded promising results [162–166].

[5] Such instabilities are encountered in section 5.1.4.

Many MLIPs are based on an additive ansatz for the potential energy, encountered previously in section 2.2, decomposing U into atomic potential energy contributions

$$U = \sum_{i \in \mathcal{R}_{\mathrm{sc}}} U_i \,, \tag{2.4.2}$$

where U_i depends on on a finite atomic neighbourhood $\mathcal{N}(i)$ only.[6] This construction ensures $O(N)$ scaling of computational cost[7] for energy[8] prediction as only a bounded number of atoms must be considered for each U_i. It also ensures extensivity in a trivial sense. However, such strict locality limits the ability to model long-range effects.[9] Attempts have therefore been made to re-introduce longer-range interactions into local models, for instance through explicit electrostatics [114], message passing mechanisms [31] (see section 4.1), Fourier-space methods [168, 169], locality in other spaces [43], or all-to-all transformer models [130]. Research in this area is ongoing. Models not based on a locality assumption have also been proposed, but are difficult to scale to larger systems [127, 129, 170].

[6] This architecture is common in FFs, and can also be motivated by the 'nearsightedness of electronic matter' [167], which also motivates the development of linear-scaling DFT methods; see [62, ch. 18].

[7] Assuming constant density.

[8] And consequently force and stress (see section 2.4.6).

[9] In a system of fixed size, if all degrees of freedom are seen by the model, long-range effects can in principle be modeled implicitly [127]. However, it is unclear how such approximations can be expected to generalise across system sizes.

Despite these challenges, MLIPs have become an important tool for atomistic simulations, and have attracted a large amount of research attention over the past years. Recent applications include the simulation of large proteins [131] and the study of disordered silicon [171].

2.4.3 Screening Tasks

In screening-type applications, a large pool of candidates, for instance drug-like molecules, must be searched for compounds with certain desirable characteristics [8]. This poses multiple challenges: The size of chemical space is formidable, even when restricted to subsets such as drug-like molecules [172], rendering brute-force approaches not always feasible. Additionally, even if the search space can be constrained sufficiently, the prediction of properties of interest can require much computational effort, restricting the reach of such approaches. Finally, the prediction of properties via quantum mechanical simulations requires the knowledge of structure first, additionally incurring computational cost due to structure relaxation.

ML models have been suggested as a way to circumvent some of these difficulties [21, 22]. MLIPs can be used in place of ab initio approaches to perform structural relaxation, for instance for the purpose of crystal structure prediction [173]. Structure-property models can be constructed to directly predict target properties for candidate structures without first-principles calculations, a task which we will investigate in section 3.3. Specialised ML models such as sure independence screening and sparsifying operator (SISSO) [174] can be used to construct data-driven analytical expressions for key properties, allowing rapid sampling and potentially more guided exploration of target spaces [175]. Generative models, which do not directly rely on supervised learning, have also been proposed to generate desired structures without a prior search [176–181].

2.4.4 Kernel Ridge Regression

Kernel ridge regression (KRR) [182] writes the prediction as

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i k(\boldsymbol{x}, \boldsymbol{x}_i) \tag{2.4.3}$$

where $k(\boldsymbol{x}, \boldsymbol{x}')$ is a *kernel function* $k : \mathcal{V} \times \mathcal{V} \to \mathbf{R}$, such that the kernel matrix $\boldsymbol{K}_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is symmetric and positive semi-definite.[10]

The regression weights $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_n]^\top$ are obtained by minimising the squared deviation between predictions and labels

$$\boldsymbol{\alpha}' = \arg\min_{\alpha} (\boldsymbol{K}\boldsymbol{\alpha} - \boldsymbol{y})^\top \cdot (\boldsymbol{K}\boldsymbol{\alpha} - \boldsymbol{y}) \,. \tag{2.4.4}$$

For KRR, regularisation is implemented by adding an additional term to the minimisation problem:

$$\boldsymbol{\alpha}' = \arg\min_{\alpha} (\boldsymbol{K}\boldsymbol{\alpha} - \boldsymbol{y})^\top \cdot (\boldsymbol{K}\boldsymbol{\alpha} - \boldsymbol{y}) + \lambda \boldsymbol{\alpha}^\top \cdot \boldsymbol{K} \cdot \boldsymbol{\alpha} \,. \tag{2.4.5}$$

The solution of this quadratic form is straightforward:

$$\boldsymbol{\alpha} = (\boldsymbol{K} + \lambda \mathbf{1})^{-1} \boldsymbol{y} \,. \tag{2.4.6}$$

In this form, we can also see that regularisation plays a role in improving the conditioning, and therefore numerical stability, during training.[11] Additionally, Cholesky decomposition is often used in place of

[10] This means

$$\boldsymbol{K}^\top = \boldsymbol{K}$$
$$\boldsymbol{c}^\top \cdot \boldsymbol{K} \cdot \boldsymbol{c} \geq 0 \quad \forall \boldsymbol{c} \in \mathbf{R}^n \,.$$

Equality iff $\boldsymbol{c} = 0$ defines positive definiteness. Sometimes, the term 'positive definite' is used in place of semi-definite, and 'strictly positive definite' in place of definite. [183] At present, semi-definiteness is sufficient.

This condition ensures that $k(\boldsymbol{x}, \boldsymbol{x}')$ corresponds to an inner product $\langle k(\cdot, \boldsymbol{x}), k(\cdot, \boldsymbol{x}') \rangle_{\mathcal{H}}$ in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ of real-valued functions [184].

$k_{\boldsymbol{x}} = k(\cdot, \boldsymbol{x})$ is the *reproducing kernel* of this space, which means that the operation of computing $f(\boldsymbol{x})$ where $\boldsymbol{x} \in \mathcal{V}$ and $f \in \mathcal{H}$ can be expressed in terms of an inner product within $\mathcal{H}$:

$$f(\boldsymbol{x}) = \langle f, k_{\boldsymbol{x}} \rangle_{\mathcal{H}} \,.$$

In other words, the kernel function provides a natural 'basis' to express functions in $\mathcal{H}$, providing the preliminaries for the *representer theorem* [185, 186].

This theorem ensures that the f minimising a regularised loss functional

$$\arg\min_{f \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} l(f(\boldsymbol{x}_i), y_i) + \lambda |f|^2_{\mathcal{H}} \,,$$

with a loss function l and the norm $|\cdot|_{\mathcal{H}}$ in the RKHS of k admits a solution in the form given by equation (2.4.3).

[11] We note in passing that it also allows us to relax the requirement of semi-definiteness of $\boldsymbol{K}$. After all, in practice, we simply require the term in parentheses to be invertible.

explicit matrix inversion.

Once $\boldsymbol{\alpha}$ is determined, predictions for new points $\boldsymbol{X}' = [\boldsymbol{x}'_1, \boldsymbol{x}'_2, ...]$ can be obtained by computing $\boldsymbol{L}_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}'_j)$ where $\boldsymbol{x}_i \in \mathcal{D}_{\text{train}}$. Then predictions $\boldsymbol{y}'$ are simply obtained by computing

$$\boldsymbol{y}' = \boldsymbol{L}^\top \cdot \boldsymbol{\alpha} \,. \tag{2.4.7}$$

THE CHOICE OF KERNEL, in addition to λ, is the main (structural) HP choice in KRR. Many kernel functions have been proposed. In this work, we employ the squared exponential (SE) kernel[12]

$$k(\boldsymbol{x}, \boldsymbol{x}') = \exp - \frac{|\boldsymbol{x} - \boldsymbol{x}'|^2}{2\sigma^2} \,, \tag{2.4.8}$$

which encodes an expectation of smoothness [187] that can be understood intuitively by considering $k(\boldsymbol{x}, \cdot)$ as the basis functions in which the target is expanded. The numerical HP σ controls the width, and hence the smoothness, of the employed Gaussians.

[12] Also commonly called Gaussian kernel, or radial basis function (RBF) kernel.

Other kernel functions range from the linear kernel $\langle \boldsymbol{x}, \boldsymbol{x}' \rangle$, the Laplacian kernel,[13] admitting cusps at the training data, to the Matérn family of kernels, providing a generalisation of the RBF kernel [152, 182]. We note that kernel methods can be used beyond KRR; in essence, the identification of kernel evaluation with an inner product in a high-dimensional RKHS can be used to transform linear methods into non-linear ones [183, 188, 189].

[13] The RBF kernel with the 1-norm instead of the 2-norm.

For atomistic modeling, we must tackle one additional difficulty: In many cases, we must solve regression problems with labels per system, but features per atom.[14] With KRR, the approach described in references [44, 190, 191] can be used: Starting from an additive ansatz for the per-system label, a modified kernel can be defined that sums over blocks of a larger atom-to-atom kernel matrix.

[14] These are local representations, introduced in section 3.1.

FINALLY, WE NOTE that the results of KRR can also be obtained from a Bayesian perspective, which yields Gaussian process regression (GPR). In GPR, the kernel function takes the role of the covariance of a Gaussian process,[15] which in turn is viewed as a prior probability distribution over possible functions. Adding training data, via Bayes rule [193], increasingly refines the posterior distribution to reflect added knowledge. Predictions are typically obtained as the mean of the resulting predictive distribution, and are identical to the ones obtained from KRR. The Bayesian perspective, however, affords a clear way to obtain and interpret predictive uncertainties.

An excellent visual introduction to Gaussian processes can be found in reference [192]. A full exposition is given by Rasmussen and Williams [152].

[15] In a nutshell, a collection of random variables (the input space $\mathcal{V}$) such that each subset is Gaussian. We can view it as a probability distribution over functions, which are obtained by drawing a sample for every point in $\mathcal{V}$. If the mean is zero, the covariance function entirely describes a Gaussian process.

2.4.5 Neural Networks

While KRR models explicitly base their predictions on the training data, introducing additional model parameters for every additional training point, a neural network (NN) has a fixed number of parameters, independent on dataset size.[16] NNs are implemented as a complex, but fixed, function with many adjustable parameters. Their conceptual basis is simple: A function f is constructed from multiple *layers*

Hastie et al. [13] introduce NNs in the context of statistical learning.

[16] This distinction between such *non-parametric* and *parametric* models becomes less clear once HPs are introduced. For good performance in practice, the size of a NN is adapted to the amount of available data.

that each consist of functions that take linear combinations of inputs and process the result with an *activation function*[17] σ. Letting l index the layer, defining n_l as the *width* of that layer, and denoting the output of the previous layer as $\boldsymbol{x}^{l-1}$, we obtain

$$\boldsymbol{x}^l = \left[\sigma(\boldsymbol{\alpha}_{lm}^\top \cdot \boldsymbol{x}^{l-1}) \,|\, m = 1...n_l \right]^\top . \tag{2.4.9}$$

The $\boldsymbol{x}^0$ are the inputs to the resulting network, which are then transformed through successive layers, yielding more and more abstract representations of the data. The final layer is then constructed to produce outputs with the desired dimensionality. This construction is called a fully-connected feed-forward NN, or multi-layer perceptron. In different limits, for instance network width or number of layers (*depth*), such NNs are universal function approximators [194, 195].

In practice, fully connected layers are used as building blocks for more intricate architectures, which can involve convolutions [32, 33], recurrence [196], or even the solution of an ordinary differential equation [197]. We will encounter a particular architecture, the MPNN, in section 4.1. For now, we consider a more fundamental question: How can we determine the parameters of a NN?

In contrast with KRR, analytical solutions are typically not available, as the loss function is not generally convex. We therefore must find a different approach to solve the minimisation problem

$$\boldsymbol{W} = \arg\min_{\boldsymbol{W}'} \frac{1}{n} \sum_{i=1}^{n} l(f_{\boldsymbol{W}'}(\boldsymbol{x}_i) - y_i) , \tag{2.4.10}$$

where we have introduced $f_{\boldsymbol{W}}$ as the NN with parameters $\boldsymbol{W}$.

Variations on gradient descent are typically used to tackle this minimisation. In particular, standard neural network training proceeds via mini-batch[18] stochastic gradient descent (SGD), extended in different ways, for instance by adding a momentum term, or methods with an adaptive learning rate. The selection of the minimisation method and its parameters is a HP, but standard methods like adaptive moment estimation (ADAM) [198] can often yield acceptable results. Training is terminated after a set number of iterations over the training set (*epochs*), or via *early stopping* approaches, where the loss on a validation set, typically set aside from $\mathcal{D}_{\text{train}}$, is used to terminate training when generalisation no longer improves. Early stopping can therefore be interpreted as a form of regularisation.

[17] Choice of activation function is a structural HP.

Biases can be included by adding a constant entry to the inputs.

We also note that this form can be seen as a simplified model of a neuron, with the inputs representing dendrites connecting to neighbouring neurons. This analogy lends NNs their name.

As above l is a loss function acting on single samples. More general forms are also possible.

[18] Gradient descent proceeds by computing the gradient of the loss function over the entire training set, which is often not computationally feasible. Stochastic gradient descent proceeds by taking the gradient for single examples, leading to noisy gradient estimates. Mini-batch stochastic gradient descent is a compromise, taking some (randomly sampled) subset of the full training data. This *batch size* is a HP. Batching in this form also yields computational advantages, as calculations can be vectorised.

2.4.6 Automatic Differentiation

The efficient training of NNs via gradient descent requires the ability to compute gradients of the loss function with respect to the parameters. The construction of MLIPs additionally requires gradients with respect to inputs, obtaining forces and stress.

Automatic differentiation (AD) provides this ability. It is a technique to obtain derivatives of functions implemented as computer programmes [38–40], and therefore also called *algorithmic* or *procedural*

differentiation. It is distinct from numerical differentiation, where finite difference schemes are employed, and symbolic differentiation, where analytical derivatives are obtained manually[19] ahead of time and then implemented manually. Instead, AD relies on the observation that complex computations can often be split up into elementary steps, for which derivatives are readily computed and implemented. If one can track those derivatives during the computation of the forward, or *primal*, function, the chain rule allows one to obtain derivatives.

[19] Or via computer algebra systems.

The result of tracking a given computation is the *computational graph*. Once the forward computation has concluded, the this graph can be traversed in two directions:[20] From the inputs *forwards* and from the outputs *backwards*, yielding forward and reverse mode[21] AD.

[20] In general, mixed schemes are possible, but not yet a standard feature in common AD frameworks. A unified treatment of different schemes is also possible [199].

[21] Another common name is backpropagation, or simply backprop.

For a general differentiable function

$$f : \boldsymbol{x} \in \mathbf{R}^N \rightarrow f(\boldsymbol{x}) = \boldsymbol{y} \in \mathbf{R}^M \,, \tag{2.4.11}$$

the Jacobian of f is defined as the $M \times N$ matrix $\partial \boldsymbol{f}_{ij} = \partial y_i(\boldsymbol{x})/\partial x_j$. AD can be used to compute the Jacobian-vector product (JVP) $\partial \boldsymbol{f} \cdot \boldsymbol{v}$ (forward mode) and the vector-Jacobian product (VJP) $\boldsymbol{v}^\top \cdot \partial \boldsymbol{f}$ (reverse mode) with the same asymptotic computational cost as the primal function. For M=1, i.e., scalar outputs, this reveals how MLIPs can efficiently obtain forces: Differentiating the scalar total potential energy U is a trivial VJP with $\boldsymbol{v}$=1. The ability to compute products of this type can also be leveraged to compute products of higher-order differential operators [200].

We emphasise two properties of AD that are particularly relevant for this work: *(a)* derivatives can only be obtained with respect to quantities that are explicitly used in the forward computation, and *(b)* the general calculation of Jacobians requires repeated (N or M) evaluations of JVPs or VJPs. If $M \propto N$ and the cost of f is $O(N)$, this leads to quadratic cost of calculating $\partial \boldsymbol{f}$ explicitly.[22]

[22] In some cases, this scaling can be circumvented by exploiting knowledge of the sparsity of $\boldsymbol{J}$. We will use this in section 4.2.

Section 2.5
Green-Kubo Method

This section discusses the Green-Kubo (GK) method, and in particular its application to the calculation of thermal transport coefficients. It represents an important application of MD simulations, starting with early experiments with hard sphere [201] and Lennard-Jones [5] FFs in the 1970s, and is the focus of chapters 4 and 5 of this thesis.

The thermal conductivity tensor $\boldsymbol{\kappa}$ describes the heat flux $\boldsymbol{J}$ arising in a system exposed to a stationary temperature gradient ($\boldsymbol{\nabla} T$), in accordance with Fourier's law:

$$\boldsymbol{J} = -\boldsymbol{\kappa} \cdot (\boldsymbol{\nabla} T)\,; \tag{2.5.1}$$

it quantifies the ability of a material to conduct heat. Its computational prediction is of great interest for the design of novel high-performance materials which are needed, for example, as thermal barrier coatings in engines [203], or thermoelectrics for waste heat recovery [204].

This thesis is concerned with lattice thermal transport. We therefore do not treat electronic or mass transport, and use $\boldsymbol{\kappa}$ to indicate lattice thermal conductivity. Equation (2.5.1) holds in a regime where temperature gradients are small on the atomic (microscopic) scale and a steady state has been achieved [202].

There are three families of approaches to this modeling task.

Non-Equilibrium Methods An immediately intuitive approach is to introduce an explicit temperature gradient into the simulation, and then directly observe the resulting heat flux. However, this approach is challenging to implement: It requires maintaining portions of the simulated systems at different temperatures using thermostats, introducing artificial boundaries. Additionally, long simulations are needed to achieve a steady state configuration, and, due to the small length scales accessible via simulation, unphysical large temperature gradients need to be imposed. [205–207]

Boltzmann Transport Approaches based on the phonon gas model lead to early breakthroughs in studying thermal transport in crystalline solids, for instance understanding the $1/T$ dependence of thermal conductivity at elevated temperatures [208], which arises from first-order anharmonic corrections to a harmonic approximation to the potential energy surface. With the advent of first-principles methods, such approaches based on the Boltzmann transport equation (BTE) have become a standard method for computing thermal conductivities [210–213]. However, this class of methods faces severe challenges in anharmonic materials with complex crystal structures: A perturbative treatment requires increasingly higher-order terms to treat strong anharmonicity, and the computational cost of evaluating force constants,

While a purely harmonic model displays infinite thermal conductivity, as no mechanism for energy exchange between modes is possible, anharmonic contributions can be treated as perturbations to such a model, leading to phonon-phonon interactions and finite thermal conductivity [209].

as well as the solution of the Boltzmann transport equation, becomes increasingly challenging [214]. The harmonic approximation can also break down entirely at elevated, or very low, temperatures, and for strongly anharmonic materials.

Green-Kubo Method Even though thermal transport occurs in a non-equilibrium situation, thermal conductivity can be computed from energy fluctuations in equilibrium, as initially observed by Onsager [215, 216] and later formalised by Green and Kubo [217–219]. This approach can account for any order of anharmonicity, and is therefore able to treat materials where anharmonic effects play an important role. In practice, it is implemented through MD simulations, requiring an accurate model of the BO PES. High-accuracy MD simulations can be performed using DFT when the exchange-correlation approximation is reliable [11]. For the GK method, this ab initio Green-Kubo (aiGK) approach [143, 151] suffers from its numerical costs which limits the system sizes and timescales, and therefore requires additional denoising and extrapolation approaches [143, 144, 220]. For this reason, FFs are often used for the GK method, and there is growing interest in the use of MLIPs for this task.

2.5.1 Intuition

Let us define a local energy density that corresponds to the total energy E of the system in question when integrated over its volume:

$$E = \int_V \mathrm{d}^3 r \, e(\boldsymbol{r}, t) \,. \tag{2.5.2}$$

This energy density cannot be expected to be uniquely defined: Any function that integrates to zero over V can be added to it without changing E [221], leading to a *gauge freedom* that carries through to the heat flux, which is introduced below. The final thermal conductivity, however, can be shown to be invariant to the resulting gauge transformation of the heat flux, as will be discussed further in section 2.5.3.

For a given, non-unique, energy density, we can then compute its first moment, the energy *barycentre*

$$\boldsymbol{B} = \int_V \mathrm{d}^3 r \, \boldsymbol{r} \, e(\boldsymbol{r}, t) \,. \tag{2.5.3}$$

The chage of this 'centre of mass' of the energy density is the *heat flux*[1]

$$\boldsymbol{J} = \frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{B} \,. \tag{2.5.4}$$

[1] In the systems studied in this thesis, the energy flux and heat flux is identical and will be used interchangeably.

We note that in addition to the non-uniqueness of the energy density, the definition of the barycentre, and consequently the heat flux, faces another difficulty: In a periodic system (see section 2.3), absolute positions $\boldsymbol{r}$ are not uniquely defined; the barycentre and heat flux as defined here are not boundary invariant. This issue is tackled in section 4.2 by deriving the microscopic heat flux density in a periodic system.

Setting these issues of uniqueness aside, we are now finally in a position to consider the core intuition behind the Green-Kubo method: When the system evolves in thermal equilibrium at finite temperature, slight inhomogeneities in the energy distribution arise at random, causing the barycentre to shift. This inhomogeneity might as well have resulted from an external perturbation; the heat flux that arises to correct it will be the same in both situations. We can therefore draw conclusions about the out-of-equilibrium phenomenon of thermal transport by observing local energy fluctuations *in equilibrium*.

We essentially recapitulate Onsager's regression hypothesis [216]:

> ... the average regression of fluctuations will obey the same laws as the corresponding macroscopic irreversible processes.

A more general statement, which also applies to quantum systems, is the *fluctuation-dissipation theorem* [222, 223].

2.5.2 Green-Kubo Formula

For a thorough and pedagogical derivation, see [142, ch. 3], [224, ch. 3], or [225].

A more rigorous statement can be derived by studying the behaviour of currents associated with the energy density, which, due to energy conservation, obeys a continuity equation

$$\dot{e}(\boldsymbol{r}, t) + \boldsymbol{\nabla} \cdot \boldsymbol{j}(\boldsymbol{r}, t) = 0\,; \tag{2.5.5}$$

Note that $e(\boldsymbol{r})$ is only defined in terms of its volume integral, introducing some ambiguity. The consequences of this gauge freedom will be discussed in the next section.

every local change $\dot{e}(\boldsymbol{r}, t)$ in energy causes a corresponding local current density $\boldsymbol{j}(\boldsymbol{r}, t)$. The integral over this density yields the heat flux

$$\boldsymbol{J}(t) = \int_V \mathrm{d}^3 r\, \boldsymbol{j}(\boldsymbol{r}, t)\,. \tag{2.5.6}$$

The Green-Kubo formula is obtained by studying how such energy currents arise in response to a perturbation of the system through a sufficiently small external temperature gradient, modelled as an additional term in the Hamiltonian.

LINEAR RESPONSE THEORY can then be used to express the expectation values of the energy currents arising in response in terms of the *unperturbed* time-evolution of the system. Spatially integrating the resulting relations, and keeping in mind the thermodynamic limit,[2] we arrive at the Green-Kubo formula, relating the thermal conductivity to the integral of the heat flux autocorrelation function (HFACF) as the system evolves in equilibrium:

[2] $N, V \to \infty$ while $N/V = \text{const}$.

$$\boldsymbol{\kappa}(T, p) = \frac{1}{k_\mathrm{B} T^2 V} \lim_{t\to\infty} \int_0^t \mathrm{d}\tau\, \langle \boldsymbol{J}(\tau) \otimes \boldsymbol{J}(0) \rangle_{T,p}\ , \tag{2.5.7}$$

In the present case, we are concerned with thermal transport. However, the Green-Kubo formalism can be applied to other transport phenomena, for instance diffusion.

with the Boltzmann constant k_B, system volume V, pressure p and temperature T. The $\langle\cdot\rangle$ denotes the ensemble average at the thermodynamic conditions of interest. More explicitly (see section 2.2):

$$\boldsymbol{\kappa}(T, p) = \frac{1}{k_\mathrm{B} T^2 V} \lim_{t\to\infty} \int_0^t \mathrm{d}\tau\, \left\langle \boldsymbol{J}(\Gamma_s^\tau) \otimes \boldsymbol{J}(\Gamma_s^0) \right\rangle_{\Gamma_s^0 \sim \mathcal{E}(T,p)}\ . \tag{2.5.8}$$

Alternatively, we could write the bracket as: $\sum_s p(\Gamma_s^0)(\ldots)$ with p denoting the probability of each Γ_s^0.

The starting conditions Γ_s^0 are sampled from a thermodynamic ensemble[3] $\mathcal{E}(T, p)$, and *evolved* in time for τ in the micro-canonical ensemble to Γ_s^τ; we take the average over all such initial conditions.

[3] We take T, p as representing any thermodynamic variable characterising the ensemble of interest.

An alternative, but equivalent, formulation is given by the Helfand-Einstein (HE) relation [226], identifying κ with the slope of the squared displacement of the energy barycentre

$$\kappa(T,p) = \lim_{t\to\infty} \frac{1}{t} \frac{1}{6k_\mathrm{B}T^2V} \left\langle |\boldsymbol{B}(t) - \boldsymbol{B}(0)|^2 \right\rangle_{T,p} \tag{2.5.9}$$

$$= \lim_{t\to\infty} \frac{1}{t} \frac{1}{6k_\mathrm{B}T^2V} \left\langle | \int_0^t \mathrm{d}\tau\, \boldsymbol{J}(\tau)|^2 \right\rangle_{T,p} . \tag{2.5.10}$$

Note that in this notation, the scalar equivalent of $\boldsymbol{\kappa}$ is obtained.

This form lends itself to an intuitive view of observing the barycentre undergo a random walk[4] through the material, diffusing away from its initial position. The 'speed' of its movement yields the transport coefficient. It also shows that the thermodynamic limit is required: In a finite system, the displacement would clearly be bounded as $t \to \infty$, and κ would be zero as a result [146, 227, 228].

[4] A random walk is a path where every step is drawn from a fixed probability distribution, for instance choosing steps ± 1 with equal probability.

2.5.3 Gauge Freedom

The HE relation makes it clear that any heat flux that can be written as the time-derivative of a quantity that is bounded in the infinite-time limit will lead to a vanishing thermal conductivity. Such a heat flux is called *non-diffusive*, since it does not correspond to a diffusion of the corresponding energy barycentre.

An extended discussion of this idea can be found in [151, 229].

A lemma proved by Marcolongo et al. [151] states that

$$|\kappa(\boldsymbol{J}_1 + \boldsymbol{J}_2) - \kappa(\boldsymbol{J}_1) - \kappa(\boldsymbol{J}_2)| \leq 2\sqrt{\kappa(\boldsymbol{J}_1)\kappa(\boldsymbol{J}_2)}\,, \tag{2.5.11}$$

which implies that if $\boldsymbol{J}_2$ is a heat flux that produces a vanishing thermal conductivity, the thermal conductivity obtained from the heat flux $\boldsymbol{J}_1 + \boldsymbol{J}_2$ is identical to the one obtained from $\boldsymbol{J}_1$ alone. Therefore, additive terms in $\boldsymbol{J}$ that are time-derivatives of a bounded quantity can be neglected without changing the resulting κ.

At this point, we encounter an apparent contradiction: The heat flux was defined as the time-derivative of the barycentre at the beginning. Therefore, one would expect the thermal conductivity to always vanish. The solution to this problem is noting that this section takes place in the thermodynamic limit: $\boldsymbol{r}$ and hence $\boldsymbol{B}$ is unbounded and therefore this theorem does not apply. In a periodic system, $\boldsymbol{r}$ is not uniquely defined and therefore $\boldsymbol{B}$ cannot be unambiguously defined. As we will see in section 4.2, this conundrum is resolved by considering the heat flux instead, defining the barycentre as its time integral.

These two observations can be used to resolve an ambiguity in the derivation of the GK theory encountered above: Since $e(\boldsymbol{r})$ is only defined in terms of its volume integral, the divergence of a bounded vector field[5] $\boldsymbol{p}(\boldsymbol{r})$ can be added to it without changing E [221]. This gauge term leads to an additional term in the heat flux,

$$e(\boldsymbol{r}) \to e(\boldsymbol{r},t) + \nabla\cdot\boldsymbol{p}(\boldsymbol{r},t) \tag{2.5.12}$$

$$\Rightarrow \boldsymbol{J}(t) \to \boldsymbol{J}(t) - \frac{\mathrm{d}}{\mathrm{d}t}\int \mathrm{d}\boldsymbol{r}\boldsymbol{p}(\boldsymbol{r},t)\,. \tag{2.5.13}$$

[5] The volume integral of the divergence can be written as a surface integral, which can be neglected in the thermodynamic limit.

Due to the lemma above, this additional term does not have an impact on the thermal conductivity. In sections 4.2.5 and 5.2, we will use this gauge freedom of the heat flux to remove non-contributing terms from the heat flux to reduce noise or avoid the computation of unnecessary terms.

2.5.4 Finite Simulations

The Green-Kubo relation was derived in the thermodynamic limit, and additionally contains the infinite-time limit of the integral of the

HFACF, as well as the thermodynamic ensemble average. We now briefly review the approximations required to implement it in practice, using the tools of MD simulations.

For a more comprehensive overview, see references [144, 230]. The particular approach to implementing the GK method used in this thesis is based on Knoop et al. [144] and further described in section 5.2.1.

Finite Size Despite the large amount of computing power available at modern HPC facilities, a system of infinite size cannot be simulated in practice. We instead approximate it by choosing a sufficiently large portion of the system under consideration, and then transitioning to a periodic system, as discussed in section 2.3. This procedure introduces *finite-size effects* as artefacts, for instance through unphysical scattering at the boundaries.

Finite Time Similar considerations apply to the runtime of simulations. Only finite simulation durations t_0 are feasible.

ANY VALUE FOR κ obtained for a particular choice of simulation cell size N and simulation duration t_0 with a practical implementation of the GK method will therefore depend on the choice of parameters. In order to report reliable estimates of κ, the convergence with respect to these parameters must be studied carefully, as we will explore in section 5.2. Transitioning to finite simulations also introduces noise, which is amplified by finite-precision arithmetic, as well as numerical integration schemes. Additional noise is introduced by non-diffusive terms in the heat flux.

Integration Time Consequently, quantities like the HFACF inevitably accumulate noise over the duration of the simulation. For this reason, and the fact that simulations are only run for finite t_0, the time integral in equation (2.5.7) must be truncated to a finite integration time $t_c < t_0$. This is justified by the expectation[6] that the HFACF decays to zero after some characteristic time τ_c. A suitable cutoff time can be determined in different ways, we follow the one outlined in [144], which first smoothes the HFACF with a lowpass filter and then takes the first time the HFACF crosses zero.

[6] Otherwise, κ would diverge in the $t \to \infty$ limit. We expect a thermodynamic system to eventually 'forget' previous states and decorrelate.

Noise Reduction In addition to choosing an integration cutoff, it is often necessary to de-noise the HFACF in order to avoid unphysical oscillations in the estimate for κ. This can be achieved by a combination of physically-motivated removal of non-diffusive terms in the heat flux, and additionally by applying a suitable low-pass filter. This is discussed further in section 5.2.

FINALLY, THE THERMODYNAMIC AVERAGE in equation (2.5.8) must be tackled: Only a finite number of initial states can be sampled from $\mathcal{E}(T, p)$. We also cannot directly sample the associated probability distribution, instead, we can access to the time-evolution of a system in a given ensemble through MD, extracting de-correlated samples from the resulting trajectory.

Ensemble Average Initial configurations $\left\{ \Gamma_s^0 \;\middle|\; s = 1 \dots n \right\}$ are sampled from the thermodynamic ensemble describing the situation being examined, for instance the canonical ensemble for constant temperature.[7] Then, we can evolve the systems s for some time t_0 in the microcanonical ensemble, allowing us to compute the instantaneous heat flux $\boldsymbol{J}_s(t)$ for each trajectory.

[7] Practical considerations typically lead to using the NVT ensemble to generate initial configurations even for constant-pressure situations.

As we have seen, while there are some practical difficulties in applying the GK approach, they are surmountable by a careful analysis of convergence, numerical approaches to noise reduction, and approximations to the ensemble average. While the computational cost of obtaining sufficiently converged results may limit the applicability of the method, no further conceptual difficulties remain – except one: the definition of the heat flux, which we will tackle in section 4.2.

Chapter 3

Review and Benchmark of Representations of Molecules and Materials

> Such is the allegory of otherness vanquished and condemned to the servile fate of resemblance. Our image in the mirror is not innocent, then. Behind every reflection, every resemblance, every representation, a defeated enemy lies concealed. The Other vanquished, and condemned merely to be the Same. This casts a singular light on the problem of representations …
>
> – Jean Baudrillard, *The Perfect Crime*

The definition of suitable input features is a highly relevant aspect of the construction of ML models, in particular those which do not rely on deep learning.[1] In this part of the thesis, we consider the design of such features for atomistic ML, and particularly focus on those fulfilling fundamental physical invariances. We term such features *representations*.

The role, requirements and different types of representations are discussed in section 3.1. Then, in section 3.2, we comprehensively survey the landscape of available representations, and discuss selected representations in detail. Finally, in section 3.3, we empirically compare those selected representations, benchmarking energy predictions for organic molecules, binary alloys, and Al-Ga-In sesquioxides in numerical experiments controlled for data distribution, regression method, and HP optimisation.

Overall, we find that, despite the wide variety of available representations, many common construction principles, as well as an overall taxonomy, can be identified. Mainly, we distinguish between local and global representations and between using invariant k-body functions and explicit symmetrisation to deal with invariances. Empirically, when controlling for other factors, similar behaviour is observed across representations. In particular, both prediction accuracy and computational cost increase with interaction order.

[1] In such systems, for instance graph-based potentials discussed in section 4.1, a rudimentary initial graph representation, which ensures translational invariance, is then augmented through *learned* transformations into a richer representation. The architecture of the NN itself has to ensure remaining invariances, for example permutation invariance through the use of commutative aggregation functions.

Publications

The presented work has been published as:

"Representations of molecules and materials for interpolation of quantum-mechanical simulations via machine learning,"
by Marcel F. Langer, Alex Goeßmann, and Matthias Rupp
in npj Computational Materials **8**, 41 (2022)
`doi:10.1038/s41524-022-00721-x`
Referenced as [44].

Data and Code Availability

Both data and code for the findings reported in this chapter are publicly available. The study was performed with the `cmlkit` package, which is discussed in appendix E.1.

The following repositories are related to this work:

- `https://gitlab.com/repbench/repbench-project` contains all additional code specific to this project. It is available at `https://marcel.science/repbench`.
- `https://gitlab.com/repbench/repbench-datasets` contains the used data splits and the datasets in `cmlkit` format.
- `https://gitlab.com/repbench/repbench-results` contains the data underlying all plots and tables, including the optimised models and HP search spaces.

Section 3.1
Role, Types, and Requirements

This section introduces the concept of representations and their role in atomistic ML. The distinction between local and global representations and the requirements they should fulfil are discussed.

The role of representations is to map atomistic systems, described as a set of N atomic positions $\mathcal{R}$, atomic charges $\mathcal{Z}$, and, if applicable, a periodic basis $\mathcal{B}$, to a feature space suitable to regression.

In practical terms, we therefore map the triple $(\mathcal{R}, \mathcal{Z}, \mathcal{B})$ to a set of real numbers with a fixed dimension n. We distinguish two elementary approaches: Those producing a set of features *per atom*, which we term *local* representations

$$\boldsymbol{r}^{\text{local}} : (\mathcal{R}, \mathcal{Z}, \mathcal{B}) \longrightarrow \boldsymbol{x} \in \mathbf{R}^{N \times n} , \tag{3.1.1}$$

and those that yield one representation per structure, which we call *global* representations,

$$\boldsymbol{r}^{\text{global}} : (\mathcal{R}, \mathcal{Z}, \mathcal{B}) \longrightarrow \boldsymbol{x} \in \mathbf{R}^{n} . \tag{3.1.2}$$

Local representations can immediately be applied to predict quantities that are intrinsically related to single atoms, for instance nuclear chemical shifts or core level excitations [231]. Extensive global properties, such as energies, can be modelled by additionally assuming additivity, decomposing a global property into atomic contributions.

In contrast, global representations are well-suited for properties of entire systems, such as the band gap or polarisability. However, they do not include extensivity[1] by design, and therefore may be ill-suited for global, but extensive, quantities.

[1] See appendix B.1.

With the introduction of deep learning, much effort has been put into including insights from representation development into end-to-end learning systems. Recent work has focused on unifying the approaches even further [232, 233]. At present, however, we focus on fixed representations, which are not trained beyond HP optimisation.

3.1.1 Requirements

Representations are used to assist machine learning models with the task of inferring a structure-property mapping from reference data, improving sample efficiency, predictive accuracy, and, ideally, computational efficiency. In order to achieve this task, representations should

fulfil a number of requirements, some of which depend on the property to be predicted.

I *Invariance* or *equivariance* to transformations acting on the input structure, including (a) changes in atom indexing, and often (b) translations, (c) rotations, and (d) reflections.

II *Uniqueness*, that is, *variance* against all transformations that change the predicted property: Two systems that differ in property should be mapped to different representations.

III (a) *Continuity*, and ideally (b) *differentiability*, with respect to $\mathcal{R}$.

IV *Computational efficiency* relative to the reference calculations.

V *Generality*; being able to encode any atomistic system.

VI *Structure* of representation feature space and the resulting data distribution should be suitable for regression.

We now discuss these requirements in more detail.

Invariance and Equivariance

In many cases, the structure-property mapping to be approximated with ML is known to change in a well-defined way under transformations of the structure. For instance, in the absence of external fields, energy is invariant to re-indexing (or permutation) of atoms of the same element, translation in space, as well as global rotations and reflections. Incorporating such knowledge into the ML model as an inductive bias can simplify the learning problem and reduce the amount of required training data [234–241].

We will use the term *geometric invariance* to refer to invariance with respect to rotations and, if required, reflections.

In this context, one can distinguish between *invariance* and *equivariance*. In the former case, the property to be predicted is unchanged under a given transformation. In the latter case, the transformation can equivalently be applied to the input or the output. As an example, while the energy, a scalar quantity, is unchanged when a molecular is rotated, the forces acting on each atom, which are vectors, rotate *with* the molecule.

Allowing the model to reflect these properties eliminates redundancy: Structures considered distinct can be treated as identical, and the model can naturally generalise to unseen, transformed, structures. Recently, the introduction of equivariant deep neural networks has revealed that the inclusion of equivariance can improve even the prediction of scalar quantities, as equivariant features can efficiently represent higher-order geometric information [242]. In this chapter, we focus on invariant models. Equivariant NNs are briefly discussed in section 3.4.

Uniqueness

The invariance requirement naturally complements the requirement of *variance* with respect to any other transformation that changes the

target property; two systems that differ in property should be mapped to different representations. Systems with equal representation that differ in property introduce errors [243–245]: Because the ML model cannot distinguish them, it predicts the same value for both, yielding at least one erroneous prediction. Uniqueness is necessary and sufficient for reconstruction, up to invariant transformations, of an atomistic system from its representation [246, 247].

Continuity and Differentiability

Discontinuities work against the regularity assumptions in ML models, which try to find the least complex function compatible with the training data. Intuitively, continuous functions require less training data than functions with jumps.

Beyond mere continuity, differentiable representations enable the construction of differentiable ML models. Gradients can then be used for training with gradient descent methods, or used to further constrain the interpolation function ('force matching'), improving sample efficiency [128, 190, 248]. If the ML model is intended to be used as a MLIP, continuous differentiability ensures that the forces obtained by differentiating the energy model are energy-conserving.

Computational Efficiency

In many cases, the introduction of ML-based approaches to quantum simulations aims to reduce computational cost. Therefore, the ML model must be faster to compute than the underlying reference data, leading to the requirement of computational efficiency.

We note in passing that the results of sufficiently cheaper simulations, compared to the method to be approximated, can be used to construct representations [249, 250] or to predict properties at a higher level of theory ('Δ-learning') [250–252].

Generality

Representations should be able to encode any atomistic system. While current representations handle finite and periodic systems, less work has been done on charged systems [237, 253–259], excited states [260–264], continuous spin systems, isotopes, and systems subjected to external fields [265, 266].

Structure

The role of the representation is to map atomistic systems into a space amenable to regression. Strictly speaking, for kernel regression this is the kernel feature space, that is, representation space transformed by the kernel. We limit our discussion to the representation itself.

Representations often have a Hilbert space structure, featuring an inner product, completeness, projections, and other advantages. Besides the formal space defined by the representation, the structure of the subspace spanned by the data is critical [245, 267].

This requirement is currently less well understood than the ones discussed previously, and largely evaluated empirically, which we will undertake in section 3.3.

IN ADDITION, SIMPLICITY, both conceptually and in terms of implementation, is a desirable quality of representations, albeit difficult to quantify. For practical application, the availability of an efficient and usable open source implementation is also desirable.

3.1.2 Representations, Descriptors, and Fingerprints

In this work, the term 'representation' is used to refer to features that fulfil invariance, continuity, and efficiency requirements, as well as uniqueness.

By this definition, $(\mathcal{R}, \mathcal{Z}, \mathcal{B})$ is not a representation, as it violates the invariance properties. Internal coordinates fulfil geometric invariance requirements, but are specific to any given system, and retain choice of freedom in the ordering of coordinates, violating I.a.

Descriptors and fingerprints from cheminformatics [268] and materials informatics violate II and III.a.

Simple representations such as the Coulomb matrix (section 3.2) either suffer from coarse-graining, violating II, or from discontinuities, violating III.a. In practice, representations do not satisfy all requirements exactly (section 3.2) but can achieve high predictive accuracy regardless (section 3.3); for example, for some datasets, modeling a fraction of higher-order terms can be sufficiently unique already [269]. The optimal interaction orders to utilise in a representation also depend on the type and amount of data available [270].

Section 3.2
Review

Having established the role and requirements of representations, we now turn our attention to the variety of proposed representations, seen in table 3.2.1, and aim to provide an overview and a general classification of these approaches. This section is structured accordingly. First, general principles for constructing representations are outlined, and then studied in detail for three particular representations: many-body tensor representation (MBTR), symmetry functions, and smooth overlap of atomic positions (SOAP). We then conclude by providing a short review of other representations, and discuss connections between them.

3.2.1 k-Body and Density Expansions

We identify two main perspectives on the problem of representing atomistic systems, and by extension approximating invariant (or equivariant) functions of the configuration of atomistic systems.

The first consideration in designing representations is the approach to symmetry. In particular, the approach to rotational, and to some extent parity, invariance can be used to categorise representations. Translational and permutational symmetry, on the other hand, are typically resolved in a similar fashion for essentially all representations: Translational invariance is ensured by working with atom-pair vectors, rather than absolute coordinates, and permutational invariance is enforced by summing over atomic contributions.

We note in passing that translational invariance can also be treated explicitly with symmetrisation, recovering atom-pair vectors [328].

Broadly speaking, geometric invariance can be tackled in two ways: Constructing representations from invariant functions directly, and explicit symmetrisation. For instance, if a representation is constructed from interatomic distances and angles, rotational invariance is immediately ensured. The alternative to this approach is to first compute an intermediate representation that is not invariant, and then explicitly integrate it over all possible transformations.

While both approaches are in principle equivalent in the appropriate limits, they nevertheless give rise to two distinct perspectives, and to different scaling of computational cost in practice. We now briefly introduce both, setting up the task at hand as follows: We aim to represent an arrangement of $N+1$ atoms, selecting a central atom, which we index with $i = 0$, as origin of the coordinate system.[1] Atom positions $\boldsymbol{r}_{j,j>0}$ are therefore identified with atom-pair vectors

[1] We make this choice in order to directly describe local representations, restricting the sum to a finite neighbourhood of the central atom. Global approaches are recovered by dropping this restriction and varying the index of the central atom over all N. Periodic systems require special treatment, as discussed in section 2.3 – at present, we simply take our positions to be within $\mathcal{R}_{\text{sc}}$, and implicitly assume the MIC.

		References		
Year	Repr.	Orig.	Development	Impl.
2007	SF	[16]	[255, 271–274]	[275, 276]
2010	BS	[17]	[277–279]	[280]
2012	CM	[21]	[231, 243, 251, 281, 282]	[276, 283]
2013	SOAP	[246]	[17, 236, 247, 278, 279, 284–288]	[276, 289]
2013	OMF	[249]	[290, 291]	—
2015	BoB	[292]	[293]	[294]
2015	WST	[295]	[295–300]	[301]
2016	MTP	[302]	[256, 303–305]	[306]
2017	MBTR	[307]	[308]	[276, 283]
2017	HDAD	[309]	—	—
2018	DECAF	[310]	—	[311]
2018	FCHL	[312]	[313]	[294]
2018	IDMBR	[314]	—	[315]
2018	MOB	[250]	[316–318]	—
2019	ACE	[319]	[237, 320]	[321, 322]
2020	NICE	[323]	—	[324]
2020	GM	[325]	—	—
2021	MILAD	[326]	—	[327]

Table 3.2.1: Overview of representations.
For each representation (Repr.), year of publication (Year), original reference (Orig.), references for further methodological development (Development), and availability of implementations (Impl.) are shown. See Glossary for abbreviations.

$\boldsymbol{r}_{ij} = \boldsymbol{r}_j - \boldsymbol{r}_0 = \boldsymbol{r}_j$. This ensures translational invariance a priori, and allows a concise notation subsequently, but, as we will see, requires a careful definition of body-order. We additionally use $\boldsymbol{P}_i$ to denote atomic properties, such as atomic numbers Z_i.

Body-Order Expansion This approach descends from the idea of expanding the PES in a series of functions of increasing body order k, i.e., explicitly acting on k atomic positions [329]:

$$f(\{\, \boldsymbol{r}_i, \boldsymbol{P}_i \mid i = 0, 1...N \,\}) = f_{k=1}(\boldsymbol{P}_0) \tag{3.2.1}$$

$$+ \sum_{j=1}^{N} f_{k=2}(\boldsymbol{r}_j, \boldsymbol{P}_j, \boldsymbol{P}_0) \tag{3.2.2}$$

$$+ \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{m=1}^{N} f_{k=3}(\boldsymbol{r}_j, \boldsymbol{P}_j, \boldsymbol{r}_j, \boldsymbol{P}_m, \boldsymbol{P}_0) \tag{3.2.3}$$

$$+ \ldots . \tag{3.2.4}$$

Note that the k=1 body function cannot take $\boldsymbol{r}_i$ into account due to translational invariance. Since $\boldsymbol{r}_{j>0}$ are atom-pair vectors, $\boldsymbol{r}_0$ is implicitly included in subsequent terms in the expansion.

Provided that f_k are constructed to be invariant, this expansion is invariant by design, and no further consideration of symmetry is needed. Typical building blocks of k-body functions include atomic number counts (k=1), distances, sometimes inverted or squared (k=2), angles or their cosine (k=3), dihedral or torsional angles (k=4). Less common, (al)chemical relationships can be included, for example, based on period and group in the periodic table [312].

Computational cost of this type of method scales like $O(N^k)$ due to the explicit evaluation of k-body functions. Nevertheless, as imple-

mentation and parallelisation is straightforward, k-body expansions form the basis of many classical FFs (see section 2.2.3), and a number of ML representations, for instance the MBTR and SF approaches, discussed below.

Density Expansion In a first step, the arrangement of atoms is represented as a density in space by placing a set of functions (or distributions) $\Delta(\boldsymbol{r})$ at each position $\boldsymbol{r}_i$:

We can only present an abridged version of this topic here. For a more detailed discussion, see reference [320].

$$\rho(\boldsymbol{r}, \{\, \boldsymbol{r}_i, \boldsymbol{P}_i \mid i = 1...N \,\}) = \sum_{i=1}^{N} \Delta(\boldsymbol{r}_i - \boldsymbol{r})\,. \tag{3.2.5}$$

Then, this density is expanded in a basis of functions $\phi_{\boldsymbol{nlm}}(\boldsymbol{r})$, yielding an intermediate representation

$$A_{nlm} \coloneqq \langle \phi_{nlm} | \rho \rangle\ , \tag{3.2.6}$$

Here, $\langle\rangle$ simply denotes an inner product; we compute a spatial overlap integral between the two functions. If delta distributions are used for $\Delta(\boldsymbol{r})$, we simply recover $A_{nlm} = \sum_j \phi_{nlm}(\boldsymbol{r}_j)$.

which is then symmetrised, formally by integrating over all possible transformations. Clearly, symmetrising a single such representation discards much useful information,[2] and therefore, ν copies of A_{nlm} are combined and symmetrised *jointly*, retaining $k = (\nu + 1)$-body correlations.[3] Note, however, that for finite bases, no *full* k-body information is retained, as k-tuples are not computed explicitly.

The advantage of this approach is precisely due to the avoidance of explicit evaluation of k-body functions: As the sum over neighbourhoods is executed *first*, no immediate polynomial scaling with body-order is introduced. However, symmetrisation incurs computational cost, limiting ν. In practice, $\nu = 2$ is often sufficient.

In order to facilitate the symmetrisation, a basis in terms of spherical harmonics Y_l^m is usually employed, which have well-understood transformation properties under rotations. In particular, the theory of the addition of angular momentum, or equivalently, the theory of representations of SO(3), yields prescriptions for how intermediate representations with given $(l_1, l_2, ..., l_\nu)$ can be combined into a final representation with a particular l. Rotational symmetrisation is then equivalent to obtaining final representations with $l = 0$, not requiring the evaluation of integrals over the symmetry group. Equivariant models can be constructed by permitting $l > 0$. We also note in passing that these ideas can be used to construct NNs with equivariant intermediate representations [242, 330, 331].

[2] For instance, consider rotationally symmetrising a single density: All angular information is lost, as retaining it would break rotatonal invariance. Two densities, on the other hand, can retain *relative* rotations. A thorough introduction can be found in [328].

[3] For example, the $\nu = 2$ power spectrum used in SOAP is equivalent to an angular distribution function [287].

THE NOTION OF k-body representations as introduced here, where we use either explicit k-body functions, or ν copies of density expansions, requires additional considerations once non-linearities are introduced, or terms are combined. For instance, products of distance-based terms are formally three-body, but cannot retain full angular information. Similarly, taking powers ζ or products of k, k'-body terms produces formally many-body terms, which regardless are less expressive than the full k^ζ- and $k \cdot k'$-body terms [269]. Non-linearities can yield formally infinite-body terms [232].

3.2.2 Selected Representations

We discuss three representations that fulfil the requirements in section 3.1 and for which a general implementation, including for periodic systems, and not tied to a specific regression approach, was available when this work was undertaken. These representations are empirically compared in section 3.3, and now serve as paradigmatic examples of the approaches outlined in the previous section.

Symmetry Functions

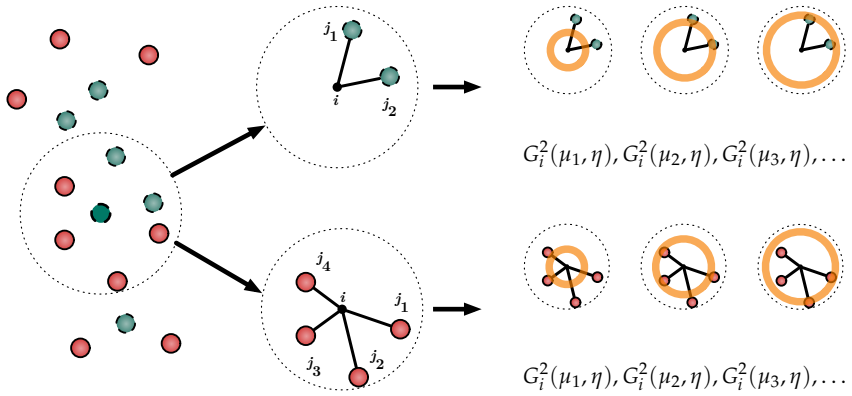


Figure 3.2.1: Illustration of the SF representation. Shown are radial functions $G_i^2(\mu,\eta)$, equation (3.2.8), for increasing values of μ. The local environment of a central atom is described by summing contributions from neighbouring atoms separately by element.

SFs [16, 271] are collections of k-body functions that describe relations between a central atom and the atoms in a local environment around it. They are typically based on distances (*radial* SFs, k=2) and angles (*angular* SFs, k=3). Each SF encodes a local feature of an atomic environment, for example the number of H atoms at a given distance from a central C atom. Figure 3.2.1 illustrates radial SFs.

For each SF and k-tuple of chemical elements, contributions are summed. Sufficient resolution is achieved by varying the HPs of an SF. For continuity (and differentiability), a cutoff function f_c ensures that SFs decay to zero at the cutoff radius r_c.

The five SFs proposed by Behler in reference [271] are

$$G_i^1 = \sum_j f_c(r_{ij}) \tag{3.2.7}$$

$$G_i^2 = \sum_j \exp\left(-\eta(r_{ij}-\mu)^2\right) f_c(r_{ij}) \tag{3.2.8}$$

$$G_i^3 = \sum_j \cos(\kappa\, r_{ij})\, f_c(r_{ij}) \tag{3.2.9}$$

$$\begin{aligned} G_i^4 = {} & 2^{1-\zeta} \sum_{j,k\neq i} (1+\lambda\cos\theta_{ijk})^\zeta \\ & \cdot \exp\left(-\eta(r_{ij}^2+r_{ik}^2+r_{jk}^2)\right) \\ & \cdot f_c(r_{ij})\, f_c(r_{ik})\, f_c(r_{jk}) \end{aligned} \tag{3.2.10}$$

$$\begin{aligned} G_i^5 = {} & 2^{1-\zeta} \sum_{j,k\neq i} (1+\lambda\cos\theta_{ijk})^\zeta \\ & \cdot \exp\left(-\eta(r_{ij}^2+r_{ik}^2)\right) f_c(r_{ij})\, f_c(r_{ik}) \end{aligned} \tag{3.2.11}$$

with the cutoff function

$$f_c(r_{ij}) = \begin{cases} 0.5\cos\left(\pi r_{ij}/r_c\right) & \text{for} \quad r_{ij} \leq r_c \\ 0 & \text{for} \quad r_{ij} > r_c\,. \end{cases} \tag{3.2.12}$$

In the above equations, index i is the central atom; j, k run over all atoms in the local environment around i with cutoff radius r_c. r_{lm} indicates pairwise distance[4] and θ_{lmn} the angle between three atoms. η and κ are broadening parameters. μ a shift parameter, and ζ determines angular resolution. $\lambda = \pm 1$ determines whether the angular part of G_i^4 and G_i^5 peaks at 0° or 180°.

The choice of which SFs to use is a structural HP. Grids of parameters are often used, for instance choosing μ in equation (3.2.8) in regular intervals. One such scheme [273] is employed in section 3.3.

Variants of SFs include partial radial distribution functions [332], improved angular resolution [272], and reparametrisations to improve scaling with the number of chemical species [255, 273, 274].

[4] In solids, the MIC is used.

Smooth Overlap of Atomic Positions

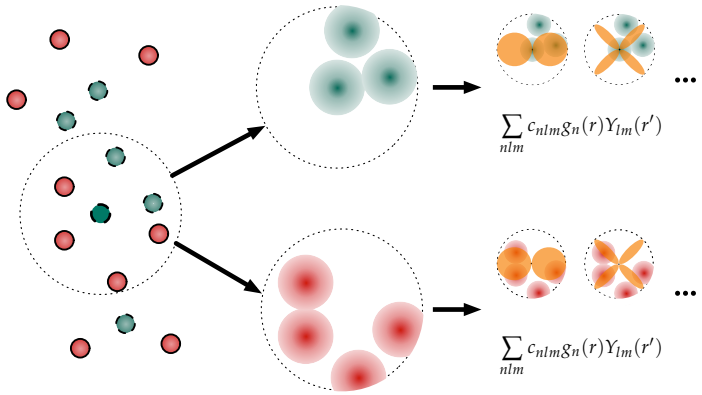


Figure 3.2.2: Illustration of the SOAP representation. The local density around a central atom is modeled by atom-centred normal distributions and expanded into radial and spherical harmonics basis functions.

SOAP [246], illustrated in figure 3.2.2, is an early example of the density-based approach. It uses Gaussian functions as $\Delta(\boldsymbol{r})$ to construct the atomic density, and employs orthogonal radial and spherical harmonics basis functions to expand the density

$$\rho(\boldsymbol{r}) = \sum_{nlm} A_{nlm}\, g_n(\boldsymbol{r})\, Y_l^m(\boldsymbol{r}), \tag{3.2.13}$$

where g_n are radial, and Y_l^m are (angular) spherical harmonics basis functions. A_{nlm}, the expansion coefficients, are then intermediate representation introduced previously.

From these coefficients, rotationally invariant quantities can be constructed, such as the power spectrum

$$p_{nn'l} = \sum_m A_{nlm} A^*_{n'lm}\,, \tag{3.2.14}$$

which can be understood as combining two copies of A_{nlm} into a joint object with $l = 0$. The power spectrum is equivalent to a radial and angular distribution function [287], and captures up to three-body interactions, as it corresponds to $\nu = 2$.

Note that the expansion of the density can be done analytically, as the expansions of Gaussians in spherical harmonics is known.

Numerical HPs are the maximal number of radial and angular basis functions, the broadening width, and the cut-off radius. Structural HPs are the type of radial basis functions.

AN ALTERNATIVE TO THE POWER SPECTRUM is the bispectrum (BS) [17], a set of invariants that couples multiple angular momentum and radial channels. It corresponds to combining $\nu = 3$ copies of A_{nlm}, and is related to three-point correlations of a four-dimensional spherical harmonics expansions. Spectral Neighbor Analysis Potential (SNAP) includes quadratic terms in the BS components [277].

Extensions of the SOAP framework include recursion relations for faster evaluation [286] and alternative radial basis functions g_n, such as third- and higher-order polynomials [286], Gaussian functions [276], and spherical Bessel functions of the first kind [247, 288]. SOAP can also be viewed as a special case of the atomic cluster expansion (ACE) approach [319, 320].

Many-Body Tensor Representation

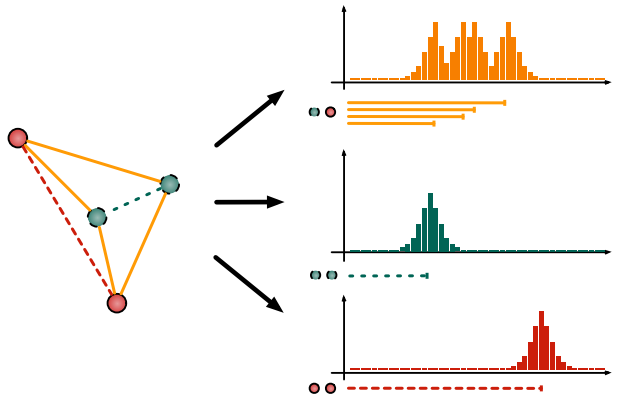


Figure 3.2.3: Illustration of the MBTR. Shown is a broadened histogram of distances (no weighting) arranged by element combination.

Finally, we consider the MBTR, illustrated in figure 3.2.3, a global representation consisting of broadened distributions of k-body terms, arranged by element combination.

For each k-body function and k-tuple of elements, all corresponding terms (for example, all distances between C and H atoms) are broadened and summed up. The resulting distributions describe the geometric features of an atomistic system:

$$f_k(x, z_1, \ldots, z_k) = \sum_{i_1 \ldots i_k} w_k \, \mathcal{N}(x|g_k, \sigma) \prod_{j=1}^{k} \delta_{z_j Z_{i_j}} , \tag{3.2.15}$$

where w_k is a weighting function that reduces the influence of tuples with atoms far from each other, and g_k is a k-body function; both w_k and g_k depend on atoms $i_1, \ldots, i_k$. $\mathcal{N}(x|\mu, \sigma)$ denotes a normal distribution with mean μ and variance σ^2, evaluated at x. The product of Kronecker δ restricts to the given element combination $z_1, \ldots, z_k$.

Periodic systems can be treated by using strong weighting functions and constraining one index to the unit cell. In practice, equation (3.2.15) can be discretised. Structural HPs include the choice of

w_k and g_k; numerical HPs include variance σ of normal distributions. Requiring one atom in each tuple to be the central atom results in a local variant [333].

3.2.3 Other Representations

Many other representations were proposed.

The Coulomb matrix (CM) [21] describes a system via inverse distances between atoms but does not contain higher-order terms. It is fast to compute, easy to implement, and in the commonly used sorted version [21], allows reconstruction of an atomistic system via a least-squares problem. However, it suffers either from discontinuities in the sorted version or from information loss in the diagonalised version as its eigenspectrum is not unique [243, 281]. A local variant exists [282].

Bag of bonds (BoB) [292] uses the same inverse distance terms as the CM but arranges them by element pair instead of by atom pair. The 'BA-representation' [293] extends this to higher-order interactions using bags of dressed atoms, distances, angles, and torsions. The inverse-distance many-body representation (IDMBR) [314] employs higher powers of inverse distances and separation by element combinations.

Histograms of distances, angles, and dihedral angles (HDAD) [309] are histograms of geometric features organized by element combination. This global representation is similar to MBTR but typically uses fewer bins, without broadening or explicit weighting.

The Faber-Christensen-Huang-von Lilienfeld (FCHL) representation [312, 313] describes atomic environments with normal distributions over row and column in the periodic table (k=1), interatomic distances (k=2), and angles (k=3), scaled by power laws.

In the FCHL18 variant [312] the full continuous distributions are used, requiring an integral kernel for regression. Among other optimisations, FCHL19 [313] discretises these distributions, similar to the approach taken by SFs, and can be used with standard vector kernels.

Wavelet scattering transform (WST) [295–300] use a convolutional wavelet frame representation to describe variations of (local) atomic density at different scales and orientations. Integrating non-linear functions of the wavelet coefficients yields invariant features, where second- and higher-order features couple two or more length scales. Variations use different wavelets [295–298, 300] and radial basis functions [296, 297, 300].

Moment tensor potentials (MTPs) [302] describe atomic environments using a spanning set of efficiently computable, rotationally and permutationally invariant polynomials derived from tensor contractions. Related representations include Gaussian moments (GM) [325], based on contractions of tensors from linear combinations of Gaussian-type atomic orbitals; the N-body iterative contraction of equivariants (NICE) framework, [323] which uses recursion relations to compute higher-order terms efficiently; ACE [237, 319, 320], which employs a basis of isometry- and permutation-invariant polynomials from trigo-

nometric functions and spherical harmonics; and, moment invariants local atomic descriptors (MILAD), which are non-redundant invariants constructed from Zernike polynomials. Extensions of density-based approaches to multiple central atoms [334], as well as architectures similar to MPNNs have been proposed [232, 233, 335].

Overlap matrix fingerprint (OMF) [249, 290, 291] and related approaches [336, 337] employ the sorted eigenvalues (and derived quantities) of overlap matrices based on Gaussian-type orbitals as representation. Eigenvalue crossings can cause derivative discontinuities, requiring post-processing [291] to ensure continuity. Using a molecular orbital basis (MOB) [250, 316], and related approaches [338], adds the cost of computing the basis, for example, localised molecular orbitals via a HF SCF calculations. Other matrices can be used, such as Fock, Coulomb, and exchange matrices, or even the Hessian, for example, from a computationally cheaper reference method.

Density-encoded canonically-aligned fingerprint (DECAF) [310] represent the local density in a canonical, invariant coordinate frame found by solving an optimisation problem related to kernel principal component analysis, representing an alternative approach to constructing invariant representations.

Tensor properties require equivariance. Proposed solutions include local coordinates from eigendecompositions [231], which exhibit discontinuities when eigenvalues cross, related local coordinate systems [310], and internal vectors (IV) [339], which are based on inner products of summed neighbour vectors at different scales, as well as covariant extensions of SOAP [235, 236] and ACE [237].

3.2.4 Connections

We now discuss relationships between representations, to which degree they satisfy the requirements in section 3.1, trade-offs between local and global representations, and relationships to other methods.

THE APPROACH TO INVARIANCE can be used to categorise representations. BoB, CM, FCHL, HDAD, IDMBR, MBTR, and SF rely on invariant k-body functions. ACE, BS, GM, MILAD, MOB, MTP, NICE, OMF, SOAP, and WST, on the other hand, rely on explicit symmetrisation. A similar distinction can be made for kernels [340].

Within a given family of approaches, further connections can be identified. For suitable HPs, SFs can be identified with terms in the distance-based MBTR, fixing one index to the central atom. If a grid of parameters is used for SFs, both representations correspond to histograms of geometric features, similar to the HDAD representation. This suggests a local MBTR or HDAD variant by restricting summation to atomic environments [333], and a global variant of SFs by summing over the whole system.

ACE, BS, GM, MILAD, MTP, NICE, and SOAP share the idea of generating tensors that are then systematically contracted to obtain rotationally invariant features. These tensors should form an orthonormal

basis, or at least a spanning set, for atomic environments. Within a representation, recursive relationships can exist between many-body terms of different orders [286, 320, 323]. References [237, 319, 320] discuss technical details of the relationships between density-based representations.

LOCAL REPRESENTATIONS CAN be used to model global properties by assuming that these decompose into atomic contributions. In terms of prediction errors, this tends to work well for energies, see appendix B.1. Learning with atomic contributions adds technical complexity to the regression model and is equivalent to pairwise-sum kernels on whole systems. Other approaches to creating global kernels from local ones exist [284].

Conversely, using global representations for local properties can require modifying the representation to incorporate locality and directionality of the property [231, 276]. A general recipe for constructing local representations from global ones is to require interactions to include the central atom, starting from k=2 [333].

TWO MODELING ASPECTS directly related to representations are which subset of the features to use and the construction of derived features. Both modulate feature space dimensionality and structure.

Adding products of 2-body and 3-body terms as features, for example, can improve performance [269], as these features formally relate to higher-order terms, but can also degrade performance if the features are unrelated to the predicted property, or if there is insufficient data to infer the relationship. Feature selection tailors a representation to a dataset by selecting a small subset of features that still predict the target property accurately enough. Optimal choices of features depend on dataset size and distribution.

In this work, we focus exclusively on representations. In kernel regression, however, kernels can be defined directly between two systems, without an explicit intermediate representation. For example, n-body kernels between atomic environments can be systematically constructed from a non-invariant Gaussian kernel using Haar integration, or using invariant k-body functions, yielding kernels of varying body-order and degrees of freedom [270, 340]. The SOAP representation was initially developed as such a kernel.

Similarly, while neural networks can use representations as inputs, their architecture can also be designed to learn implicit representations from the raw data (end-to-end learning). However, some featurisation is typically employed as pre-processing, for instance the transformation of structures into a graph representation. This is discussed in further detail in section 4.1.

In all cases, the requirements in section 3.1 apply.

SOME REPRESENTATIONS, IN PARTICULAR EARLY ones such as the CM, do not fulfil these requirements. Most representations fulfil some requirements only in the limit, that is, absent practical constraints such

as truncation of infinite sums, short cutoff radii, and restriction to low-order interaction terms. The degree of fulfilment therefore often depends on HPs. Effects can be antagonistic; for example, in SOAP, both uniqueness and computational effort increase with nlm [246]. In addition, not all invariances of a property might be known or require additional effort to model, for example, symmetries [234].

Mathematical proof or systematic empirical verification that a representation satisfies a requirement or related property are sometimes provided: The symmetrised invariant moment polynomials of MTPs form a spanning set for all permutationally and rotationally invariant polynomials [302]; the ACE framework encompasses these polynomials and allows their systematic construction [320]. For SOAP, systematic reconstruction experiments demonstrate the dependence of uniqueness on parametrisation [246].

While uniqueness guarantees that reconstruction of a system up to invariances is possible in principle, accuracy and complexity of this task vary with representation and parametrisation. For example, reconstruction is a simple least-squares problem for the global CM as it comprises the whole distance matrix, whereas for local representations, (global) reconstruction is more involved.

If a local representation comprises only up to 4-body terms then there are degenerate environments that it cannot distinguish [245], but that can differ in property. Combining representations of different environments in a system, which is typically done to predict global properties like total energies, can break the degeneracy.

However, by distorting feature space, these degeneracies degrade learning efficiency and limit achievable accuracy, even if the training set contains no degenerate systems [245]. It is currently unknown whether degenerate environments exist for representations with terms of order $k>4$. The degree to which a representation is unique can be numerically investigated through the eigendecomposition of a sensitivity matrix based on its derivatives with respect to atom coordinates [291].

Section 3.3
Benchmark

In this section, we benchmark the three selected representations, SFs, SOAP, and MBTR, on three publicly available datasets, `qm9`, `ba10`, and `nmd18`, probing to what extent the choice of representation can influence prediction accuracy. The benchmark datasets are related to screening applications, rather than MD, and we therefore focus on energy predictions, as opposed to forces and dynamical properties. For MLIPs, different benchmarking approaches must be taken, which are discussed in other works, for instance reference [341].

	Reference											
	[309]	[276]	[341]	[252]	[308]	[342]	[343]	[344]	[291]	[345]	[267]	here
Finite systems	✓	✓	×	✓	×	✓	✓	✓	✓	✓	✓	✓
Periodic systems	×	✓	✓	×	✓	×	✓	×	×	×	✓	✓
Other properties	✓	✓	×	×	×	✓	✓	✓	✓	×	✓	×
Numerical HPs	×	✓	✓	×	×	✓	×	×	×	✓	×	✓
Structural HPs	×	×	×	×	×	×	×	×	×	×	×	✓
Regression HPs	✓	✓	✓	✓	✓	✓	×	✓	×	✓	×	✓
Timings	×	×	✓	✓	×	×	×	×	×	×	×	✓

Table 3.3.1: Overview of related work. *Finite systems*: study uses datasets of finite systems, such as molecules or clusters; *Periodic systems*: uses datasets of periodic systems, such as crystalline materials; *Other properties*: evaluate properties other than energy (and forces); *numerical / structural / regression HPs*: which HPs were optimised automatically.

Works introducing novel representations often compare their performance estimates with those reported in the literature. While such comparisons are useful, it can be difficult to draw conclusions on representations only, as other factors differ, for instance the datasets, training and test set choices, regressions methods, choice of HPs, validation procedures, and reported quantities. In this work, we aim to control for such factors, in particular regression method, data distribution, and HP optimisation approach.

Several other studies systematically measured and compared prediction errors of representations, which are listed in table 3.3.1. In the table, we distinguish between studies that automatically optimise numerical HPs of representations, for example, the width of a normal distribution; structural HPs of representations, for example, choice of basis functions; and HPs of the regression method, for example, regularisation strength.

3.3.1 Datasets

qm9

The qm9 dataset [251, 346], also known as gdb9-14, contains 133 885 small organic molecules composed of H, C, N, O, F with up to 9 non-H atoms, and is a standard benchmark dataset for ML models.

It is a subset of the 'generated database 17' (GDB-17) [347]. Molecular ground state geometries and properties, including energetics, are computed with DFT using the Becke 3-parameter Lee-Yang-Parr (B3LYP) [348] hybrid functional with 6-31G(2df,p) basis set.

In this benchmark, we predict the atomisation energy at 0 K.

We use the version available at qmml.org, which offers a convenient format for parsing, and exclude all structures in the uncharacterized.txt file and those listed in the readme.txt file as 'difficult to converge', as those are potentially problematic. Total energies were converted to energies of atomisation by subtracting the atomic contributions given in file atomref.txt. We further exclude structures with ≤ 6 non-H atoms.

ba10

The ba10 dataset [308], also known as dft-10b, contains unrelaxed geometries and their enthalpies of formation for the 10 binary alloys AgCu, AlFe, AlMg, AlNi, AlTi, CoNi, CuFe, CuNi, FeV, and NbNi. For each alloy system, unrelaxed geometries with lattice parameters from Vegard's rule [349, 350] and energies are computed for all possible unit cells [351] with 1–8 atoms for face-centred cubic (FCC) and body-centred cubic (BCC) lattices, and 2–8 atoms for hexagonal close-packed (HCP) lattices, using the PBE functional [69] with projector-augmented wave (PAW) potentials and generalized regular k-point grids [352, 353]. The dataset contains 631 FCC, 631 BCC, and 333 HCP structures per alloy system, yielding 15 950 structures in total.

We predict enthalpies of formation.

We use the version available at qmml.org, removing structures with fewer than 5 atoms in the simulation cell. No further processing, beyond the stratification discussed below, is applied.

nmd18

The nmd18 dataset [354] is a Kaggle challenge [355] dataset containing 3 000 ternary oxides,[1] of potential interest as transparent conducting oxides. We predict formation and bandgap energies of relaxed structures, using either relaxed (nmd18r) or approximate (nmd18u) structures from Vegard's rule as input. Geometries and energies are computed with DFT using the PBE exchange-correlation functional as implemented in FHI-aims [356] with tight settings.

The challenge scenario is to predict formation and band-gap energies of relaxed structures from unrelaxed geometries obtained via Vegard's rule. This is equivalent to strong noise or bias in the inputs. Unlike pure benchmarking scenarios, where computationally expensive relaxed geometries are given, the challenge scenario is closer to a virtual screening application in that Vegard's rule geometries are computationally inexpensive to obtain.

The dataset contains all structures from the challenge training and leaderboard data. We report root mean squared error (RMSE), as opposed to the root mean square logarithmic error (RMSLE) used in the challenge.

Once again, the version of the dataset available at qmml.org is used. Only structures with more than 10 atoms in the simulation cell are used.

[1] $(Al_x\text{-}Ga_y\text{-}In_z)_2O_3$ with $x + y + z = 1$.

3.3.2 Method

Our overall goal in this benchmark is to isolate the impact of representations on prediction error. We therefore attempt to keep all other possible factors constant, designing the experiment to treat all representations as similarly as possible. In this section, we briefly discuss our design choices.

Stratification Rather than sampling subsets purely at random, we use multivariate stratification to ensure that subsets are representative of the overall dataset. This reduces the variance of performance estimates and ensures the validity of the i.i.d. data assumption inherent in ML.

Additional information is given in the dataset repository.

We achieve this by using a simple Monte-Carlo approach, iterating until selected statistics of each subset match the parent dataset within a fractional tolerance, typically <1 %. For dataset `qm9`, these were number of N, O and F atoms, number of molecules with 7, 8 and 9 non-H atoms, binned number of atoms (with H), and binned energy. For dataset `ba10`, these were number of all constituting elements, unit cells with 6, 7, 8, and 9 atoms, binned sizes and energies. For dataset `nmd18`, these were number of Al, Ga, In, O atoms, unit cells with 20, 30, 40, 60, 80 atoms, and binned energies.

Subsets For training and validation, data subsets were sampled as follows: An *outer validation set*[2] was drawn using the procedure detailed above (10 k molecules for `qm9`, 1 k structures for `ba10`, 600 structures for `nmd18`). From the remaining entries, *outer training sets* of sizes 100, 250, 650, 1 600, 4 000 and 10 000 for datasets `qm9`, `ba10` and 100, 160, 250, 400, 650, 1 000 and 1 600 for dataset `nmd18` were randomly drawn. These sizes were chosen to be equidistant in log-space. Each outer training set was then split into an *inner training set* and an *inner validation set* by randomly drawing the latter. We used an 80 / 20 split, yielding inner validation sets of size 20, 50, 130, 320, 800, 2 000 for datasets `qm9`, `ba10` and 20, 32, 50, 80, 130, 200, 320 for `nmd18`. The whole procedure was repeated 10 times, yielding 10 different outer train/validation splits.

[2] In the literature, the terms 'test set' and 'validation set' are sometimes used with different meaning. To avoid confusion, we use 'outer' for the subset employed to measure performance, and 'inner' for the subset employed to optimise HPs.

Regression Method We use KRR (section 2.4), with a Gaussian kernel as ML model. KRR is a widely-used non-parametric regression method, with few tunable HPs: In the present setting, we must only tune the length scale of the Gaussian kernel, σ, and the regularisation strength λ. Another advantage of KRR is that training amounts to solving a convex optimisation problem; results are therefore deterministic.

Representations We compare the selected representations discussed in section 3.2.2: SFs, SOAP, and MBTR. SFs and SOAP are local representations, representing the k-body function and density expansion approaches. MBTR is a global representation based on on k-body functions.

For the k-body approaches we consider variants with k=2 and k=2, 3. For SOAP, which corresponds to k=2, 3, a variation in body-order is not available. Details of remaining HP choices are discussed in the supplementary material of reference [44].

Hyper-Parameter Optimisation For this work, we adopt an end-to-end perspective, essentially considering HP optimisation as part of the training procedure. The 'models' being compared are therefore not a combination of representation and regressor with *fixed* HPs. Instead, we compare model families, or search spaces, fixing only broad architectural choices, and allowing ranges of possible parameters for all other HPs. For each training set size, we then optimise HPs on the training set, using the inner train/validation splits. Once HPs have been determined, the model is re-trained on the entire outer training sets and used to predict the corresponding outer validation set. For computational efficiency, we only tune HPs on one outer split, rather then re-tuning for each. Once the search space has been defined, no human input is needed, the HP tuning procedure is fully automatic.

In order to execute the described HP optimisation approach, I developed the `cmlkit` package, described in appendix E.1.

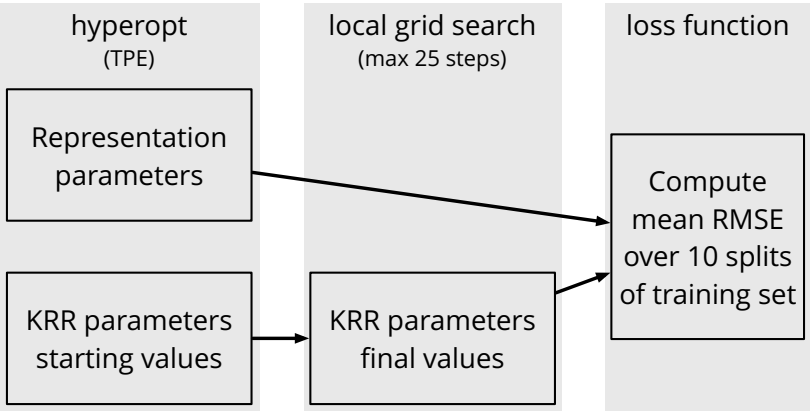


Figure 3.3.1: Sketch of the HP optimisation approach. TPEs, implemented in the `hyperopt` package [357], are used to sample representation parameters and starting parameters for a local grid search of KRR parameters. The overall loss function is the mean RMSE over inner training/validation splits.

Due to the combinatorial nature of these search spaces, a full enumeration is unfeasible. Instead, we use a model-based stochastic optimisation method based on TPEs [357, 358]. In a nutshell, this method models the space of possible models as a tree-structured set of choices, for instance, between different k-body functions, or different values of a numerical HP. TPEs treat this search space as a prior distribution over HPs, updated every time a loss is computed to increase prior weight around HP settings with better loss. Search therefore amounts to drawing samples from this distribution, and, once the loss is evaluated, updating the probability distribution. This approach has the advantage of being straightforward to parallelise, as samples can be drawn while others are being evaluated. In defining the search space, we use uniform priors throughout, discretising numerical HPs on logarithmic or linear grids as necessary.

For KRR parameters, we adopt a two-step procedure: TPE is used to determine starting parameters for a local grid search of σ and λ, which attempts to find the nearest local minimum of the loss land-

scape. The loss for this second optimisation is the mean RMSE over all ten inner splits, and its result is used as loss for TPE. This allows the fine-tuning of KRR parameters without requiring the recomputation of representations. We use the best result, judged by final loss, of three independent runs of the HP optimisation procedure.

Details on the search spaces used in this work can be found in the supplementary material of reference [44].

We aim for similarly-sized search spaces for all numerical HPs, and a comprehensive selection of structural ones, treating all representations on an equal footing as much as possible. Some choices, however, have to be excluded *a priori* to avoid excessive computational cost.

Learning Curves The main outcome of this study are learning curves, which show the dependence of prediction error ϵ on training set size n. Asymptotically, we assume the error to decay as a negative power [359, 360] $\epsilon = a' n^{-b}$. On a log-log plot, ϵ is therefore linear, $\log \epsilon = a - b \log(n)$, and the offset $a = \log a'$ and slope b can be used to characterise predictive performance of models [293]. For asymptotic fits, we weight training set sizes by the standard deviation over their respective splits to attenuate for small sample effects, as the above equation is valid only in the limit $n \to \infty$.

Note that we use base-10 logarithms for learning curves.

Timings Additionally, we report timings for the calculation of representations and kernel matrices. Experiments were run on a single core of an Intel Xeon E5-2698v4 2.2 GHz processor on the `draco` HPC system. For consistency, all timings were run on the first outer validation set of each dataset. Reported timings are the mean over three repetitions, and then further averaged over the number of entries in the respective evaluation sets. Kernel matrices are computed between all points in the respective validation set, and therefore averaged over the squared number of entries.

Pareto Frontiers Based on learning curves and timings, we also report on the relationship between computational cost and accuracy. When comparing observations in two dimensions, here time t and error ϵ, there is no unique ordering $<$, and we resort to the usual notion of dominance: Let $\boldsymbol{x}, \boldsymbol{x}' \in \mathbf{R}^d$; then $\boldsymbol{x}$ dominates $\boldsymbol{x}'$ if $x_i \leq x'_i$ for all dimensions i and $x_i < x'_i$ for some i. The set of all non-dominated points is called the Pareto frontier, indicating the 'best possible' tradeoff between compute times and error.

Error Metrics We report RMSE, the loss minimized by least-squares regression such as KRR, and thus a natural choice. Additionally, we provide the rRMSE, as a quantity that can be compared across datasets. Both are defined in appendix A.3. For comparison with literature results, we also give MAE in figures B.2 to B.4.

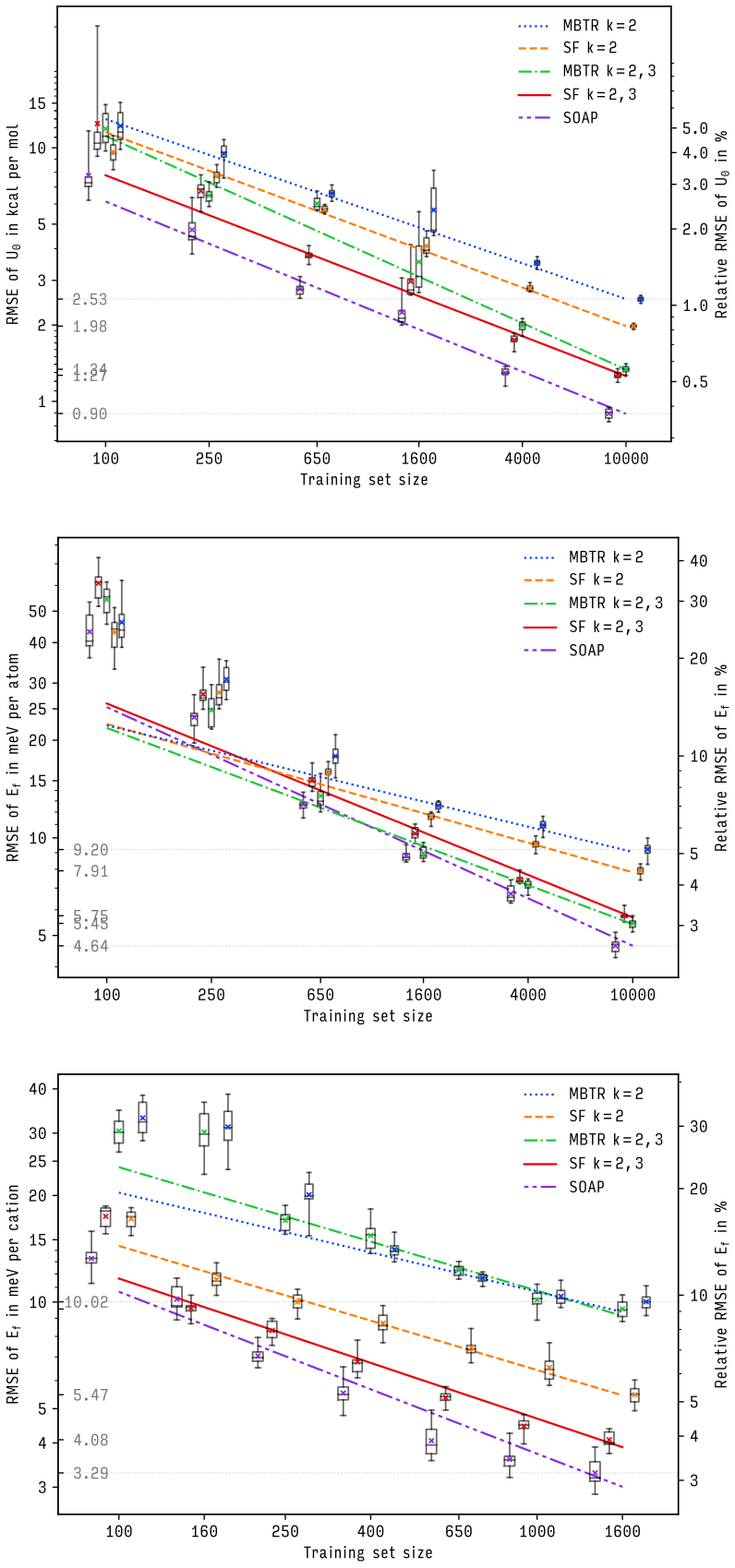


Figure 3.3.2: Learning curves for selected representations on datasets `qm9` (top), `ba10` (centre), and `nmd18r` (bottom). Shown are RMSE and rRMSE of energy predictions on out-of-sample-data as a function of training set size. Boxes, whiskers, bars, crosses show interquartile range, total range, median, mean, respectively. Lines are fits to theoretical asymptotic RMSE.

See figure B.3 for MAE.

3.3.3 Results

Figure 3.3.2 displays the learning curves for the main considered datasets, `qm9`, `ba10`, and `nmd18r`. Asymptotically, observed prediction accuracies relate as

$$\begin{aligned} \text{SF}(k{=}2,3) &\prec \text{SF}(k{=}2), & \text{MBTR}(k{=}2,3) &\preceq \text{MBTR}(k{=}2), \\ \text{SOAP} &\prec \text{SF}(k{=}2,3), & \text{SOAP} &\prec \text{MBTR}(k{=}2,3), \\ \text{SF}(k{=}2,3) &\preceq \text{MBTR}(k{=}2,3), & \text{SF}(k{=}2) &\prec \text{MBTR}(k{=}2), \end{aligned}$$

where $A \prec B$ ($A \preceq B$) indicates that A has lower (or equal) estimated error than B asymptotically. Except for $\text{MBTR}(k{=}2,3) \not\preceq \text{SF}(k{=}2)$ on dataset `nmd18r`,

$$\text{SOAP} \prec \text{SF}(k{=}2,3) \preceq \text{MBTR}(k{=}2,3) \prec \text{SF}(k{=}2) \prec \text{MBTR}(k{=}2).$$

We conclude that, for energy predictions, accuracy improves with interaction order and for local representations over global ones. The magnitude of these effects varies across datasets.

Converged prediction errors are in reasonable agreement with the literature (see tables B.1 to B.3), considering the lack of standardised conditions such as sampling, regression method, HP optimisation, target properties, and reported performance statistics.

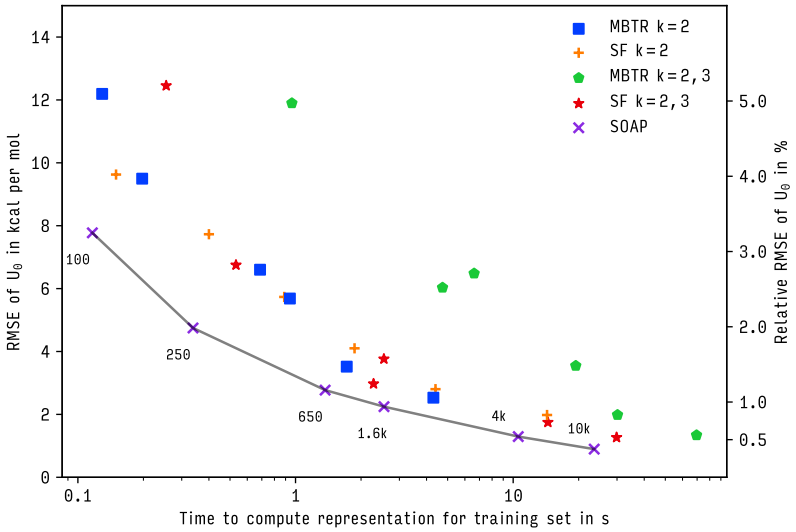
Time in ms	Dataset		
Representation	`qm9`	`ba10`	`nmd18`
MBTR $k{=}2$	0.76 ± 0.32	13 ± 5.1	340 ± 99
SF $k{=}2$	1.40 ± 0.18	3.3 ± 1.4	8.2 ± 1.1
MBTR $k{=}2,3$	12.0 ± 6.9	290 ± 140	$28\,\text{k} \pm 4.4\,\text{k}$
SF $k{=}2,3$	2.80 ± 0.85	27 ± 12	98 ± 89
SOAP	1.90 ± 0.54	9.1 ± 4.8	19.0 ± 8.6

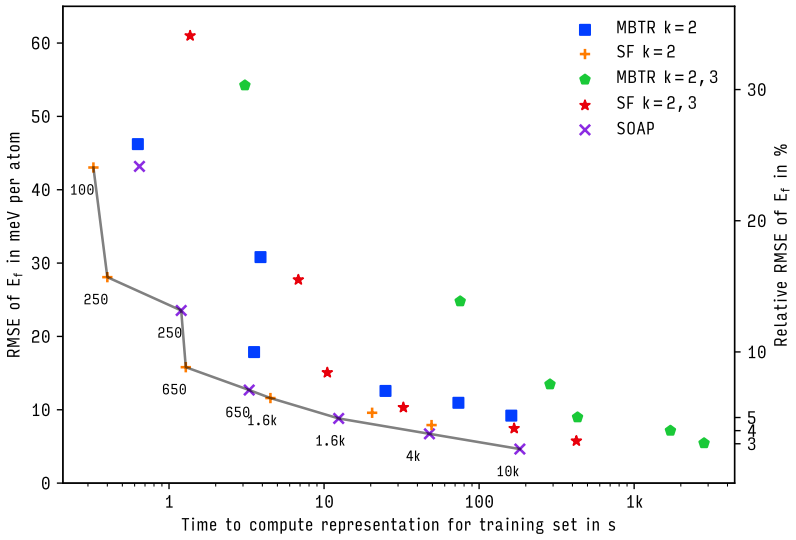
Table 3.3.2: Computational cost of calculating representations. Shown are mean ± standard deviation over all training set sizes of a dataset.

Computational costs, shown in figure 3.3.3 and tables 3.3.2 to 3.3.4, tend to increase with predictive accuracy, due to the cost of additional training data and higher body-order terms. Representations should therefore be selected based on a target accuracy, constrained by available computing resources. SOAP largely lies on the Pareto frontier, only occasionally dominated by less accurate, but faster SF($k{=}2$) representations. Computational cost of MBTR with $k{=}2,3$ increases significantly for periodic systems, and in particular for `nmd18`, which includes larger simulation cells than `ba10`.

To some extent, however, the fast calculation of local representations is balanced out by the higher cost of computing kernel matrices, as seen in table 3.3.3. Local representations require the intermediate computation of an atom-atom kernel matrix (see section 2.4.4), and therefore the calculation of the kernel scales quadratically with the number of atoms.[3] Table 3.3.4 shows the resulting overall computational cost of predictions. Despite the higher cost of computing the representation,

[3] Methods to mitigate this scaling behaviour exist; an overview of sparsity-based approaches can be found in reference [361].





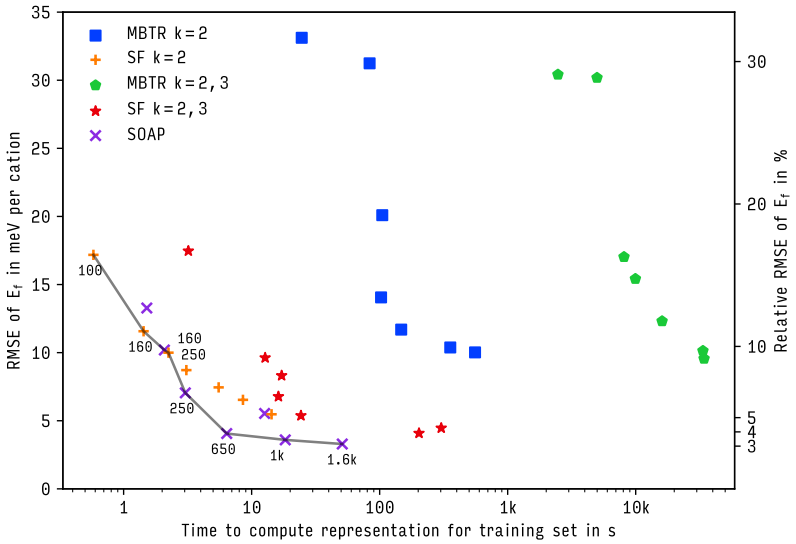


Figure 3.3.3: Compute times of selected representations for datasets `qm9` (top), `ba10` (centre), and `nmd18r` (bottom). Shown are RMSE and rRMSE of energy predictions on out-of-sample-data as a function of the time needed to compute all representations in a training set. Lines indicate Pareto frontiers; inset numbers show training set sizes.

See figure B.4 for MAE.

Time in µs	Dataset		
Representation	qm9	ba10	nmd18
MBTR $k = 2$	0.16	0.64	0.13 ± 0.05
SF $k = 2$	12.16 ± 1.80	8.90 ± 1.39	72.95 ± 4.71
MBTR $k = 2,3$	0.90	7.01 ± 0.16	0.92 ± 0.96
SF $k = 2,3$	17.41 ± 3.25	20.03 ± 4.02	106.35 ± 14.87
SOAP	59.78 ± 37.73	72.60 ± 19.70	296.23 ± 258.28

Table 3.3.3: Computational cost of calculating single system-system kernel matrix elements. Shown are mean ± standard deviation over all training set sizes of a dataset.

the global MBTR remains competitive in overall performance, with the exception of nmd18 and k=2,3.

	Dataset								
	qm9			ba10			nmd18		
Representation	t_{rep}	t_{kernel}	total	t_{rep}	t_{kernel}	total	t_{rep}	t_{kernel}	total
MBTR k=2	8s	+ 16s	= 23s	2m	+ 1m	= 3m	57m	+ 13s	= 57m
SF k=2	14s	+ 20m	= 20m	33s	+ 15m	= 15m	1m	+ 2h	= 2h
MBTR k=2,3	2m	+ 1m	= 3m	49m	+ 12m	= 1h	76h	+ 2m	= 77h
SF k=2,3	28s	+ 29m	= 29m	4m	+ 33m	= 38m	16m	+ 3h	= 3h
SOAP	19s	+ 2h	= 2h	2m	+ 2h	= 2h	3m	+ 8h	= 8h

Table 3.3.4: Overview of computational costs. Shown are runtime estimates for prediction with N_{train}=N_{pred}=10 000. Based on mean compute times t_{rep} for representations and t_{kernel} for kernel matrices from tables 3.3.2 and 3.3.3, we estimate total prediction times as $N_{\text{test}} \cdot t_{\text{rep}} + N_{\text{train}} \cdot N_{\text{test}} \cdot t_{\text{kernel}}$. Times are rounded to the nearest second, minute, or hour.

DEPENDENCE OF PREDICTIVE ACCURACY on interaction order has been observed by others [276, 277, 312, 314, 362] and might be partially due to a higher resolution of structural features; some structures cannot be distinguished from two-body (and even three-body) features alone [245]. However, such degeneracies are not the only factor: Even if structures can be distinguished in principle by distances alone, angular terms allow more immediate characterisation of structure, for instance in carbon rings of organic molecules appearing in qm9 [314]. These effects may become visible in different data regimes: In ba10, differences in accuracy between k=2 and k=2,3 appear only larger training set sizes, potential due to the resolution limit of k=2 features, whereas in qm9, where angular features are immediately relevant, differences apear even for small training set sizes. We observe an increase in the slope of the learning curves, with body-order, varying in magnitude between data-sets.

BETTER PERFORMANCE OF LOCAL REPRESENTATIONS might be due to higher resolution and better generalisation (both from representing only a small part of the whole structure), and has also been observed by others [345, 363]. The impact of assuming additivity is unclear but likely depends on the structure of the modeled property, see appendix B.1. Our comparison includes only a single global representation (MBTR), warranting further study of the locality aspect.

Similar observations have been made for equivariant NNs: In multi-layer atomic cluster expansion (MACE), where body-order can be explicitly controlled, Batatia et al. [335] observe a systematic increase in slope with body-order ν. However, the relationship between ν, degree L of employed intermediate equivariant layers, and learning curves remains an active domain of investigation [242].

PREDICTIVE ACCURACY IS WORSE for solid-state datasets compared to the molecular `qm9` one. This might indicate that periodic systems pose harder learning tasks than molecules.

MBTR performs worse for solid-state datasets than for the `qm9` one, in particular for `nmd18r`. This might be due to lack of intrinsic scaling with number of atoms, impeding interpolation between unit cells of different size. The high computational cost of MBTR with k=3 for in this setting also renders HP optimisation more difficult.

FOR THE `QM9` DATASET AT 1600 training samples, we observe an increase in RMSE standard deviation compared to neighbouring training set sizes for most methods. Comparing to MAE, see figure B.3, which exhibits no such effect, and investigating errors individually, revealed that this is due to outliers. Few predictions with high error appear in some, but not all, outer splits. The problematic structures are ring molecules, and are not present in the outer training split used for HP optimisation. This stresses the importance of carefully stratifying benchmark datasets, and highlights the limitations of the present approach to HP optimisation, which only considered one of the outer training sets.

FIGURE B.1 PRESENTS RESULTS for the `nmd18` dataset with approximate geometries obtained from Vegard's rule, which we term `nmd18u`. In contrast to relaxed structures, such geometries can be obtained at almost no cost, and could be used in virtual screening campaigns.

We observe *(i)* a strong increase in prediction errors (24 % to 26 % for rRMSE, from 3 % to 10 %), *(ii)* collapse of all representations to similar performance, *(iii)* large differences between MAE and RMSE, see figure B.2, indicating significant outliers.

From this, we conclude that the map from unrelaxed structures to ground-state energies is harder to learn than the map from relaxed structures to their energies, as it requires implicitly approximating the relaxation process. This mismatch between structures and property dominates the error, so representations are not the limiting factor in this setting.

OUR FINDINGS SUGGEST the following guidance:

- If their prediction errors are sufficient for an application, we recommend k=2 versions of simple representations such as SFs and MBTR as they are fastest to compute.
- For large systems, local representations should be used.
- For strong noise or bias on input structures, as in dataset `nmd18u`, performance differences between representations vanish, and computationally cheaper features that do not satisfy the requirements in section 3.1 (descriptors) suffice.

Section 3.4
Summary

In this chapter, we reviewed and benchmarked representations of atomistic systems for the construction of ML models of energies obtained with first-principles calculations.

We distinguished local and global representations and between using invariant k-body functions and explicit symmetrisation of a density representation to deal with invariances, and found that many previously suggested representations can be described in these frameworks. An overview of existing approaches was given.

Empirically, we observed that when controlling for other factors, for instance distribution of training and validation data, regression method, and HP optimisation, both prediction accuracy and compute time of SFs, MBTR and SOAP increase with interaction order, and for local representations over global ones.

Limitations

The benchmark in section 3.3 was restricted to a particular regression model, KRR, and to only one property: Energy. We therefore do not consider MLIPs, where forces are required, or the prediction of other properties, for instance bandgaps. We also do not study end-to-end models like NNs, where representations are learned, rather than fixed, and the regressor is optimised jointly with the representation.

In particular, this excludes equivariant NNs, and the larger context of geometric deep learning [364]. In a nutshell, these networks [240, 330] rely on the ideas discussed in section 3.2: Architectures such as NequIP [242] or So3krates [43] are inspired by density-based representations, constructing intermediate layers with $l>0$ and ensure that all operations within the network preserve equivariance by using appropriate tensor products. Other approaches, for instance DimeNet [365], rely more explicitly on k-body information, constructing messages from angles in addition to distances. Other methods, such as PaiNN [331], employ an intermediate approach: To avoid the computational overhead of transitioning to a spherical harmonics basis, vectorial features are generated from atom-pair vectors and non-linearities in 'real space' are employed. In such equivariant NNs, similar relationships between body-order and degree of equivariance have been observed [335]. An understanding of the relationship between equivariance, body-order, and dataset and property dependence so far remains elusive, and represents an intriguing direction for future work.

Chapter 4
Heat Flux for Semi-Local Machine-Learning Potentials

> I saw the best minds of my generation destroyed by madness, starving hysterical naked, …
> who passed through universities with radiant cool eyes hallucinating Arkansas and Blake-light tragedy among the scholars of war, …
>
> –Allen Ginsberg, *Howl*

The GK method [217–219], briefly introduced in section 2.5, is a framework for the prediction of the thermal conductivity of materials, including strongly anharmonic and complex compounds. It is based on observing fluctuations in the distribution of energy, the *heat flux*, during equilibrium MD simulations. It therefore requires access to the BO PES as well as a definition of the heat flux.

FFs have been used for the GK method since the 1970s [366–371], decades prior to the recent development of ab initio Green-Kubo (aiGK) approaches [143, 151]. MLIPs continue this tradition, aiming to combine computational efficiency, albeit typically reduced in comparison to FFs, with the accuracy of DFT. While different types of short-ranged MLIPs have been used to investigate thermal transport via GK [163, 372–378], more recent semi-local MLIPs [31, 41, 42, 130, 233, 242, 257, 335, 365, 379, 380] have not yet been used for this purpose, partially because a heat flux formulation that incorporates message-passing mechanisms was not available.

In this chapter, we fill that gap and explain how to implement the GK approach for such MLIPs using AD.

Since AD relies on explicitly tracing the 'forward' computation, an understanding of how atomic potential energies are computed in practical implementations of such MLIPs is required. Section 4.1 is dedicated to this task: It introduces the notion of GLPs, which take a graph of MIC atom-pair vectors as inputs, encoding atomic neighbourhoods, and compute atomic potential energy contributions either based on neighbourhoods alone (local models) or can iteratively take neighbours-of-neighbours into account (semi-local models). As preparation for the heat flux, the definition and computation of forces and stress is discussed, unifying previous formulations for FFs and MLIPs.

Having established this foundation, we are then in a position to

consider the computation of the heat flux in section 4.2. As starting point, we first re-derive a general form of the heat flux, originally due to Hardy [381], to explicitly account for periodicity. Then, strategies for efficiently implementing the heat flux with AD are discussed, and the 'unfolded' heat flux is introduced, which is applicable to GLPs and scales linearly with system size.

Related publications

Results of the presented work have been submitted for publication as:

"Heat flux for semilocal machine-learning potentials,"
by Marcel F. Langer, Florian Knoop, Christian Carbogno, Matthias Scheffler, and Matthias Rupp
in Physical Review B *in press*
`arXiv:2303.14434`
Referenced as [45].

"Stress and heat flux via automatic differentiation,"
by Marcel F. Langer, J. Thorben Frank, and Florian Knoop
in revision
`arXiv:2305.01401`
Referenced as [46].

In particular, the heat flux for semi-local MLIPs was introduced in [45], results for stress and GLPs appear in [46].

Data and Code Availability

The `glp` package uses `jax` to implement the forces and stress from section 4.1, and the heat flux formulations presented in section 4.2. It is available at `https://github.com/sirmarcel/glp`. A `pytorch` implementation of different heat flux formulations for `schnetpack` is available at `https://github.com/sirmarcel/gknet-archive`.

Data and code related to tables 4.1.1 and 4.1.2 can be found at `doi:10.5281/zenodo.7852530`.

Section 4.1

Graph Machine-Learning Potentials

Not the Edge lecture again, I said.

– William Gibson, *New Rose Hotel*

In this section, we discuss how interatomic potentials with a finite cutoff radius can be viewed as acting on a graph representation of a molecule or material. We introduce the term graph-based machine-learning potential (GLP), or *graph potential*, to describe this class of interatomic potentials. This class contains the 'bonded' terms of common FFs, local MLIPs, as well as models featuring message-passing mechanisms [31], which allow semi-local interactions.

Having introduced this unified perspective, which specifies the 'forward' computation of the potential energy, we then argue that AD affords a straightforward and efficient way to compute forces and stress tensors for GLPs, even for periodic systems. The definition of pairwise forces in many-body potentials is discussed briefly, and an overview and comparison of equivalent stress tensor formulations is provided.

We emphasise that such graph representations are not a novel concept. Graph kernels were proposed early to construct ML models for molecules, for instance in reference [382] and references therein, and graph neural networks [383] have become a popular class of models for atomistic ML. The aim of this section is to connect such graph-based MLIPs to other MLIPs and FFs, arguing that they can be treated in a unified manner.

4.1.1 Semi-Local Interatomic Potentials

The starting point for our considerations is an additive ansatz for the potential energy U, which we encountered in section 2.2.3. In such interatomic potentials, U is computed as the sum of atomic contributions U_i, which are in turn computed based on local neighbourhoods $\mathcal{N}(i)$ for each atom. Due to the requirement of translational invariance, these contributions can only depend on atom-pair vectors; absolute positions cannot be used.[1]

Atomic potential energies therefore depend on pairwise connections to neighbouring atoms – a description that can be formalised in terms of a graph, which we will now undertake. The foundation of graph potentials is a representation of the system under consideration as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with the vertices (or nodes) $\mathcal{V}$ representing the atoms in the simulation cell, labeled by Z_i. The edges $\mathcal{E}$ encode atomic neighbourhoods

$$\mathcal{E} = \left\{ \boldsymbol{r}_{ij}^{\mathrm{mic}} \;\middle|\; |\boldsymbol{r}_{ij}^{\mathrm{mic}}| \leq r_{\mathrm{c}},\, i, j \in \mathcal{R}_{\mathrm{sc}} \right\} \tag{4.1.1}$$

$$= \left\{ \boldsymbol{r}_{ij}^{\mathrm{mic}} \;\middle|\; i \in \mathcal{R}_{\mathrm{sc}},\, j \in \mathcal{N}(i) \right\} . \tag{4.1.2}$$

[1] More precisely, it is advantageous to include translational symmetry when constructing models of the BO PES; see section 3.1.1.

In this chapter, we focus on periodic systems and therefore use $\mathcal{R}_{\mathrm{sc}}$; the non-periodic case can be recovered by setting $\mathcal{B}$ larger than any possible interaction distance. We will also drop $\mathcal{Z}$ from now on. In general additional labels can be assigned to nodes and edges, for instance for the purposes of message passing; we suppress such complications for now.

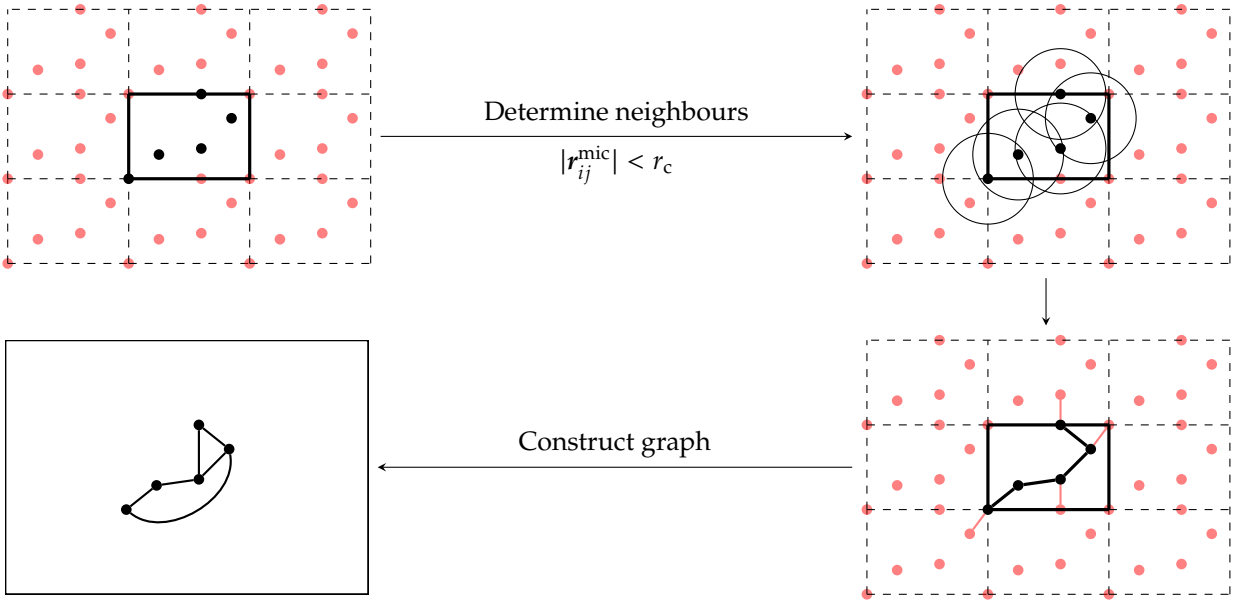


Figure 4.1.1: Construction of a graph representation.

Crucially, as we rely on the MIC to construct $\mathcal{G}$, it is boundary invariant. As discussed in section 2.3, in order to avoid discontinuities, we additionally require that r_c is chosen such that at most one replica is within a given atomic environment.[2] As opposed to the original definition of crystal graphs by Xie and Grossman [384], $\mathcal{G}$ therefore only contains single edges between vertices.[3] This construction is sketched in figure 4.1.1. We note in passing that the topology of $\mathcal{G}$ can change during the course of an MD simulation; GLPs must therefore be constructed to remain continuously differentiable as such changes occur.

IF WE RESTRICT INTERACTIONS to nearest neighbours on the graph, setting $U_i = U_i(\mathcal{N}(i))$, we recover the notion of a strictly local potential, and can therefore naturally describe any potential of this type, either MLIPs or FFs. As the number of neighbours can be expected to remain constant as N increases, provided we keep the density fixed as we take the limit, this yields $O(N)$ scaling of energy predictions. However, not all interactions of interest may be captured within r_c.

Let us therefore consider how to construct interactions beyond the cutoff. The simplest way would be to increase the cutoff radius directly, simply enlarging atomic neighbourhoods. This leads to a cubic increase in the size of neighbourhoods, which can introduce computational difficulties.[4] Semi-local GLPs [31, 41, 42, 130, 233, 242, 257, 335, 365, 379, 380] are an intriguing alternative construction. In such systems, longer-range interactions are iteratively built up from local ones by repeated interactions between neighbouring atomic environments. For M such interactions, the maximum effective cutoff radius therefore becomes $r_c^{\text{eff}} := r_c M$, retaining most of the computational advantages of the original cutoff radius while allowing interactions beyond local neighbourhoods. Alternatively, this approach can be viewed as an explicit sparsification of the full model within the extended neigh-

[2] In other words, $r_c < r_c^{\max}$ as defined in equation (2.3.4).

[3] More precisely, each edge between vertices exists exactly twice, once in each direction. From a physical perspective, this definition amounts to a kind of 'double counting', since $\boldsymbol{r}_{ij} = -\boldsymbol{r}_{ji}$; no information is added by taking both directions into account. However, this convention avoids the difficulty of defining an arbitrary ordering of edges, and can be used to distinguish derivatives at a later stage. We will make use of this in section 4.2.8.

[4] We note in passing that in the case of simple pairwise interactions, Ewald summation [99, 100] or fast multipole methods [101, 102] can be used to recover $O(N \log N)$ or linear scaling, respectively. However, we consider general many-body interactions, where such techniques are not available at this point.

bourhood defined by r_c^{eff}.

The *interaction order* M allows us to classify GLPs. At M=1, only information from direct neighbours of i is used to compute U_i, at M=2, the neighbours of all neighbours of i are included as well, and so on. In other words, M counts the maximum number of edges traversed from i to compute U_i. We introduce $\mathcal{E}^M(i)$ as that set of edges, and $\mathcal{N}^M(i)$ as the corresponding set of neighbourhoods.

For M>1, the energy at an atom i may depend on edges between atoms outside the local atomic neighbourhood. For this reason, we use the term *semi-local* MLIP, distinguishing from potentials with fully local character. However, we must emphasise that for finite M, such models are still local[5] in the extended neighbourhood within the cutoff radius r_c^{eff}. In order to realise the computational advantage associated with the smaller r_c, however, we cannot explicitly construct these extended neighbourhoods for each i.

Note that while atomic potential energies have a range of r_c^{eff}, i.e., change only due to position changes within that radius, the forces can be influenced by changes up to $2r_c^{\text{eff}}$.

[5] Non-local models are discussed briefly in chapter 6.

We finally define graph potentials as potential energy functions where $U = \sum_{i \in \mathcal{R}_{\text{sc}}} U_i(\mathcal{G})$. For this thesis, we further restrict GLPs to those where r_c^{eff} admits only interactions with single replicas.[6,7] Setting M=1, this definition includes distance-truncated terms of classical FFs (see section 2.2.3), as well as MLIPs that are not explicitly defined as graph NNs, provided that atomic potential energies are computed based on atom-pair vectors.

In this framework, as we rely on a cutoff radius to define $\mathcal{G}$, we cannot readily describe potentials with global interactions or long-range electrostatics. In some cases, for instance when inputs to electrostatics are predicted based on local environments [114, 385], some of the present considerations could be nevertheless applied. We do not pursue this here.

[6] This is not a crucial restriction, but it simplifies the comparison of heat flux formulations in section 4.2: If $r_c^{\text{eff}} > r_c^{\max}$, the MIC does not apply and $\boldsymbol{J}^{\text{mic}}$ is no longer equivalent to $\boldsymbol{J}_{\text{pot}}^{\text{unfolded}}$.

[7] This criterion is $r_c^{\text{eff}} < r_c^{\max}$ as defined in equation (2.3.4).

4.1.2 Message-Passing Neural Networks

Having introduced the abstract notion of semi-local GLPs, we now consider their explicit construction using the message-passing neural network (MPNN) architecture [31]. MPNNs operate on $\mathcal{G}$ in three stages, illustrated in figure 4.1.2: initialisation, message-passing, and readout, where each message-passing step is labeled by t=0, ..., M. During initialisation (t=0), every atom i is assigned a state vector $\boldsymbol{s}_i^{t=0}$, based on its chemical species. During the next stage, these states are *updated* with *messages* $\boldsymbol{m}_i^t$ that can depend on the states of both the receiver atom i and its neighbours $j \in \mathcal{N}(i)$, as well as on the edges $\boldsymbol{r}_{ij}$ connecting them:

$$\boldsymbol{m}_i^{t+1} = \sum_{j \in N(i)} \boldsymbol{M}_t(\boldsymbol{s}_i^t, \boldsymbol{s}_j^t, \boldsymbol{r}_{ij}) \tag{4.1.3}$$

$$\boldsymbol{s}_i^{t+1} = \boldsymbol{F}_t(\boldsymbol{s}_i^t, \boldsymbol{m}_i^{t+1}) \,. \tag{4.1.4}$$

Here, the message functions[8] $\boldsymbol{M}_t$ and update functions $\boldsymbol{F}_t$ are differentiable functions implemented as neural networks. They are learned during training, and can differ over message-passing iterations. The

To simplify the notation, we now drop the MIC label from $\boldsymbol{r}_{ij}^{\text{mic}}$, as all $\boldsymbol{r}_{ij}$ appearing in the remainder of this section are edges in $\mathcal{G}$ and therefore generated with the MIC in mind.

[8] In practice, message functions are not entirely arbitrary. To ensure smoothness, or more precisely, to ensure that the potential is a continuously differentiable function of positions, the message function typically incorporates a predefined cutoff function $f_c(r_{ij})$. This function is defined such that contributions to energies and forces continuously approach zero as r_{ij} approaches the cutoff radius r_c. If this is not the case, the resulting FF ceases to be conservative and energy conservation is violated.

message functions $\boldsymbol{M}_t$ model pairwise interactions. Their aggregation into the total message $\boldsymbol{m}_i^{t+1}$ via a sum ensures permutational invariance. The update functions $\boldsymbol{F}_t$ describe how the interactions with all neighbours influence the new state of the receiver atom. Over multiple iterations, information is *propagated* through the graph.

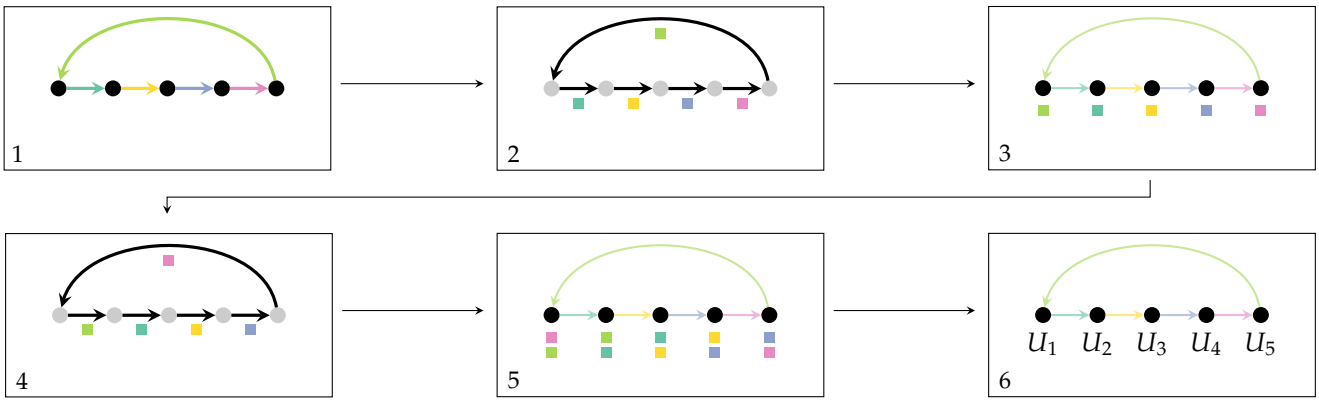


Figure 4.1.2: Sketch of neural message passing with M=2. Connections are only considered in one direction for simplicity.
(1) Initialisation: Each edge symbolises an interatomic distance and is associated with a colour, each node an atom with an empty initial state.
(2+3) Message-passing: Each node is updated based on the (empty) neighbouring state, and its incoming edge. Afterwards, each node depends on the incoming edge.
(4+5) Another message-passing step. Now, every state depends on next-to-nearest neighbours.
(6) Readout: U_i are predicted based on the final states.

After M message-passing steps, the final stage is reached where a readout function R predicts the atomic potential energy contributions U_i. This atom-independent readout function is learned during training and acts on each vector $\boldsymbol{s}_i^M$ representing the final state of atom i. The total potential energy U is then simply given by

$$U = \sum_{i \in \mathcal{V}} U_i \qquad \text{with} \qquad U_i = R(\boldsymbol{s}_i^M)\,. \tag{4.1.5}$$

Different strategies can be employed to deal with rotational invariance. Early MPNNs such as SchNet [41, 42] only use interatomic distances as inputs, discarding angular information. Message functions that take angular or higher-order information into account are also possible [233, 335, 365]. Recent equivariant MPNNs [43, 130, 238, 240, 242, 386] can make use of atom-pair vectors and can ensure that $\boldsymbol{s}_i^t$ and $\boldsymbol{m}_i^t$ transform appropriately under rotations. The presented derivations are applicable to both invariant and equivariant MPNNs.

4.1.3 Periodicity and Unfolded Construction

The 'standard' construction of GLPs discussed so far incorporates periodicity implicitly: Interactions are wrapped around boundaries, adopting the 'toroidal' view of a periodic system discussed in section 2.3. Since the atom-pair vectors within $\mathcal{N}(i)$ are identical whether i is in the simulation cell or a replica, this is equivalent to messages propagating into neighbouring replicas. By keeping all interactions within $\mathcal{R}_{\text{sc}}$, redundant computational effort for replicas is avoided.

However, this efficiency has a cost: Vertices on the graph represent both atoms in $\mathcal{R}_{\text{sc}}$ and replicas. Therefore, derivatives obtained with AD with respect to $\boldsymbol{r}_i$ contain contributions from both. This poses an essential difficulty for the computation of the heat flux, which requires

the attribution of partial derivatives to replica positions, and will be discussed further in section 4.2.

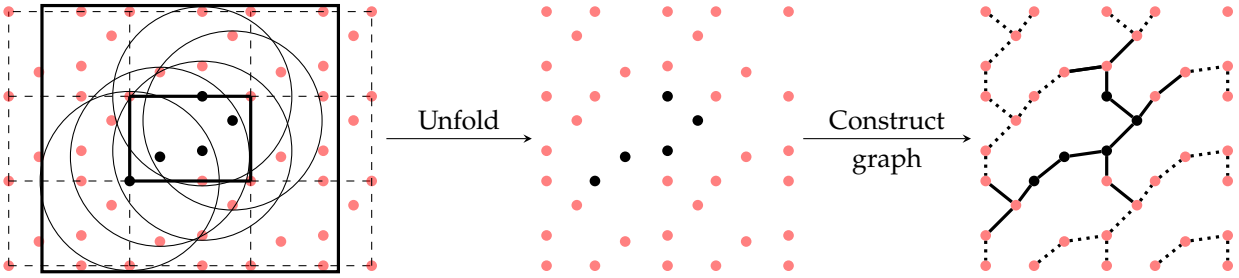


Figure 4.1.3: Construction of an 'unfolded' graph representation for the system in figure 4.1.1 with M=2. First (left), $\mathcal{R}_{\text{unf}}$ is determined by extending the simulation cell up to $r_{\text{c}}^{\text{eff}}$. Then (centre), positions in $\mathcal{R}_{\text{unf}}$ are explicitly constructed. Finally (right), the input graph $\mathcal{G}$ is constructed from all positions in $\mathcal{R}_{\text{unf}}$. Positions in $\mathcal{R}_{\text{sc}}$ are marked in black, replicas in red. Edges that contribute to $U_i \in \mathcal{R}_{\text{sc}}$ are drawn as solid, others as dotted lines.

One solution, inspired by previous work on the derivation of the stress and heat flux for periodic systems [387, 388], and illustrated in figure 4.1.3, lies in modifying the construction of $\mathcal{G}$ to reflect the 'checkerboard' picture from section 2.3, constructing replicas explicitly. To this end, the simulation cell is extended, or *unfolded*, to include all positions that can interact with positions in the simulation cell. The graph is then construction for this unfolded simulation cell, $\mathcal{R}_{\text{unf}}$, without the MIC. By construction, predictions for nodes representing atoms in $\mathcal{R}_{\text{sc}}$ are then identical, while derivatives of these potential energies U_i with respect to positions in $\mathcal{R}_{\text{unf}}$ distinguish between the simulation cell and replicas.

The unfolded simulation cell can be constructed efficiently by determining atoms that lie within $r_{\text{c}}^{\text{eff}}$ of the boundary; a brief overview over a practical implementation is given in appendix C.3. The number of additional positions is proportional to the surface area of the simulation cell and therefore scales as $O(N^{2/3})$ (see appendix C.5). It is therefore asymptotically dominated by the increase in N itself.

4.1.4 Derivatives

As seen in section 2.2, the main quantities relevant for MD simulations are derivatives of the PES. We now consider how to compute forces and stress using AD.

Forces For general MD simulations, we at least require the forces

$$\boldsymbol{F}_i = -\frac{\partial U}{\partial \boldsymbol{r}_i} \,. \tag{4.1.6}$$

Since U is a scalar, and $\mathcal{R}_{\text{sc}}$ are an explicit input to its computation, the forces can be computed as a trivial JVP[9] that can be evaluated with the same asymptotic cost as U.

An interesting situation arises if pairwise forces are desired. Strictly speaking, in a many-body potential, where interactions cannot be decomposed into pairwise contributions, such quantities are not well-defined and Newton's third law is replaced by conservation of momentum, which requires $\sum_{i=1}^{N} \boldsymbol{F}_i = 0$. Nevertheless, pairwise forces

[9] The product is over an output dimension of size 1.

with an antisymmetric structure can be defined by exploiting the construction of GLPs in terms of atom-pair vectors.[10]

U is a function of all edges within M hops,

$$U = U(\{\, \boldsymbol{r}_{ij} \;|\; ij \in \mathcal{E}^M(i) \,\}) \,. \tag{4.1.7}$$

Hence, by the chain rule,

$$\boldsymbol{F}_i = \sum_{j\in\mathcal{N}(i)} \frac{\partial U}{\partial \boldsymbol{r}_{ij}} - \frac{\partial U}{\partial \boldsymbol{r}_{ji}} \tag{4.1.8}$$

$$=: \sum_{j\in\mathcal{N}(i)} \boldsymbol{F}_{ij} \,. \tag{4.1.9}$$

The pairwise forces such defined exhibit anti-symmetry, and therefore fulfil Newton's third law. For M=1, this recovers a more standard form [371] where only neighbouring atoms are involved,

$$\boldsymbol{F}_{ij} = \frac{\partial U_i}{\partial \boldsymbol{r}_{ij}} - \frac{\partial U_j}{\partial \boldsymbol{r}_{ji}} \,. \tag{4.1.11}$$

However, for M>1, this calculation of pairwise forces includes a sum over all U_k that are influenced by a given edge

$$\boldsymbol{F}_{ij} = \sum_{k\in\mathcal{N}^M(i)} \frac{\partial U_k}{\partial \boldsymbol{r}_{ij}} - \frac{\partial U_k}{\partial \boldsymbol{r}_{ji}} \,, \tag{4.1.12}$$

subverting expectations connecting local potential energies to pairwise forces. This apparent contradiction is a consequence of the combination of the peculiar construction of GLPs and AD: In principle, it is always possible to define extended neighbourhoods up to r_c^{eff}, obtaining U_i purely as a function of atom-pair vectors originating from i. However, to construct derivatives with respect to these atom-pair vectors with AD, these extended neighbourhoods have to be computed explicitly, therefore negating the computational efficiency gains of a GLP architecture.

Stress The definition of the (potential) stress is [76]

$$\boldsymbol{\sigma} = \frac{1}{V} \frac{\partial U'}{\partial \boldsymbol{\epsilon}}\Big|_{\boldsymbol{\epsilon}=0} \,, \tag{4.1.13}$$

with U' denoting the potential energy after a transformation of all positions with the 3×3 symmetric strain tensor $\boldsymbol{\epsilon}$

$$\boldsymbol{r} \to (\mathbf{1} + \boldsymbol{\epsilon}) \cdot \boldsymbol{r} \quad \text{for} \quad \boldsymbol{r} \in \mathcal{R}_{\text{all}} \,. \tag{4.1.14}$$

While computing the derivative in equation (4.1.13) for arbitrary potentials and periodic systems has been classified as requiring 'much effort' [390] in the past, and has been the focus of much discussion [387, 389], it is straightforward with AD.

Two approaches are possible: *(a)* Equation (4.1.13) can be implemented directly by explicitly performing the strain transformation during the calculation of U, obtaining the stress with AD by differentiating with respect to $\boldsymbol{\epsilon}$, or *(b)* equation (4.1.13) can be re-written in terms of other derivatives, which are then computed with AD.

[10] For pairwise distances, and without reference to graph structure, a more formal argument along similar lines appears in the work by Admal and Tadmor [389].

In particular,

$$\frac{\partial U}{\partial \boldsymbol{r}_i} = \sum_{jk\in\mathcal{E}^M(i)} \frac{\partial U}{\partial \boldsymbol{r}_{jk}} \frac{\partial \boldsymbol{r}_{jk}}{\partial \boldsymbol{r}_i} \,, \tag{4.1.10}$$

the second term is only non-vanishing when either j or k are i.

This yields a number of equivalent definitions of the stress

$$V\boldsymbol{\sigma} = \frac{\partial U(\mathcal{R}_{\text{sc}}', \mathcal{B}')}{\partial \boldsymbol{\epsilon}} \tag{4.1.15}$$

$$= \frac{\partial U(\mathcal{E}')}{\partial \boldsymbol{\epsilon}} \tag{4.1.16}$$

$$= \frac{\partial U(\mathcal{R}_{\text{unf}}')}{\partial \boldsymbol{\epsilon}} \tag{4.1.17}$$

$$= \sum_{i \in \mathcal{R}_{\text{sc}}} \boldsymbol{r}_j \otimes \frac{\partial U}{\partial \boldsymbol{r}_i} + \sum_{\boldsymbol{b} \in \mathcal{B}} \boldsymbol{b} \otimes \frac{\partial U}{\partial \boldsymbol{b}} \tag{4.1.18}$$

$$= \sum_{ij \in \mathcal{E}} \boldsymbol{r}_{ij} \otimes \frac{\partial U}{\partial \boldsymbol{r}_{ij}} \tag{4.1.19}$$

$$= \sum_{i \in \mathcal{R}_{\text{unf}}} \boldsymbol{r}_i \otimes \frac{\partial U}{\partial \boldsymbol{r}_i} \,. \tag{4.1.20}$$

recovering formulations given by Louwerse and Baerends [390] and Thompson [387].[11,12] All formulations are compared in table 4.1.1 for the Lennard-Jones potential and in table 4.1.2 for the So3krates [43] GLP with M=2. They are found to be equivalent. Further discussion and details on these experiments can be found in appendix C.1 as well as reference [46].

A MORE COMPLEX SITUATION arises if strain derivatives of atomic energies, in other words, atomic stresses

$$V\boldsymbol{\sigma}_i \coloneqq \frac{\partial U_i}{\partial \boldsymbol{\epsilon}} \quad \text{for} \quad i \in \mathcal{R}_{\text{sc}} \tag{4.1.21}$$

are required. Their calculation requires either one backwards pass per U_i, or one forward pass for each entry in $\boldsymbol{\epsilon}$. If only reverse-mode AD is available, its evaluation therefore scales quadratically with N. Linear scaling is retained with forward mode. For GLPs with M=1, linear scaling in reverse mode can be recovered by using equation (4.1.19): Every edge can be uniquely assigned to one U_i, and therefore the derivatives can be used to construct atomic stresses. For M>1, this is not possible; similar to the observations for pairwise forces, atomic stresses take a semi-local form.

As will be discussed in section 4.2, similar issues arise in the calculation of the heat flux, which relies on derivatives of atomic potential energies U_i: A direct implementation requires separate AD passes for each U_i and leads to quadratic scaling.

TO SUMMARISE, THIS SECTION has introduced an abstract description of interatomic potentials in terms of a mapping between a graph $\mathcal{G}$ of atom-pair vectors (the edges $\mathcal{E}$) and atoms (the vertices $\mathcal{V}$) and a set of atomic potential energies U_i. In this framework, (pairwise) forces and the stress tensor can be defined and implemented in a unified way, not requiring further information about the structure of the potential. To illustrate this, and enable subsequent work, the `glp` package, using `jax` [37], briefly discussed in appendix E.3, has been developed.

The notation $\mathcal{S}'$ indicates that $\boldsymbol{\epsilon}$ is applied to all vectors in $\mathcal{S}$.

It must be stressed that the equivalence of the given formulations rests on definitions so far. For instance, the edges in $\mathcal{G}$ must be constructed with the MIC as defined in equation (2.3.3). If they are instead computed using the modulo operation in fractional coordinates, the stress cannot be obtained directly from $\mathcal{R}_{\text{sc}}$ and $\mathcal{B}$; equations (4.1.15) and (4.1.18) yield incorrect results.

Equation (4.1.20) uses $\mathcal{R}_{\text{unf}}$, the 'unfolded' simulation cell that contains all positions that contribute to atomic energies in the simulation cell. We defined it previously in section 4.1.3.

[11] In particular, equation (4.1.18) appears as equation 8 in reference [390] and equation (4.1.20) can be identified with equation 13 therein. Equation (4.1.18) is the 'atom-cell' form given in reference [387, eq. 27], equation (4.1.20) the 'atom' form in equation 25. In the case of pair-additive GLPs with M=1, equation (4.1.19) reduces to the standard form which can be found, for example, in reference [390, eq. 2].

[12] Different forms are implemented in available MLIP packages. For example, `schnetpack` [391, 392] and `nequip` [242, 393] implement equation (4.1.15), MACE in `pytorch` [394] implements equation (4.1.15), while MACE in `jax` [395] uses equation (4.1.18).

Equation	MAE in eV	maxAE in eV	MAPE in %	maxAPE in %
Fin. diff.	7.70×10^{-4}	4.18×10^{-3}	1.04×10^{-1}	5.30
4.1.15	1.19×10^{-5}	6.54×10^{-5}	1.79×10^{-3}	7.35×10^{-2}
4.1.16	8.25×10^{-6}	4.11×10^{-5}	1.27×10^{-3}	6.75×10^{-2}
4.1.17	9.17×10^{-6}	4.46×10^{-5}	1.36×10^{-3}	4.39×10^{-2}
4.1.18	1.18×10^{-5}	6.44×10^{-5}	1.79×10^{-3}	7.19×10^{-2}
4.1.19	8.22×10^{-6}	4.16×10^{-5}	1.27×10^{-3}	6.59×10^{-2}
4.1.20	9.16×10^{-6}	4.51×10^{-5}	1.37×10^{-3}	4.92×10^{-2}

Table 4.1.1: Error in stress for Lennard-Jones argon, comparing different formulations, as well as finite differences, with an analytical implementation in ASE. Results are shown for single precision arithmetic, and for $\sigma \cdot V$ in place of σ.

Equation	MAE in eV	maxAE in eV	MAPE in %	maxAPE in %
4.1.15	1.58×10^{-2}	8.26×10^{-2}	4.40×10^{-2}	7.01×10^{-1}
4.1.16	1.58×10^{-2}	8.26×10^{-2}	4.38×10^{-2}	6.92×10^{-1}
4.1.17	1.57×10^{-2}	8.37×10^{-2}	4.33×10^{-2}	6.69×10^{-1}
4.1.18	1.58×10^{-2}	8.27×10^{-2}	4.40×10^{-2}	7.00×10^{-1}
4.1.19	1.58×10^{-2}	8.26×10^{-2}	4.38×10^{-2}	6.91×10^{-1}
4.1.20	1.57×10^{-2}	8.37×10^{-2}	4.33×10^{-2}	6.71×10^{-1}

Table 4.1.2: Error in stress for SnSe, using the M=2 So3krates model discussed further in section 5.4, comparing different formulations of the stress with finite differences. Results are shown for single precision arithmetic, and for $\sigma \cdot V$ in place of σ.

Section 4.2
Heat Flux

> *... there are many subtleties associated with the calculation of transport coefficients in computer simulations. These subtleties cannot be completely divorced from a consideration of the boundary conditions.*
>
> – M.P. Allen, *Computer Simulations in Chemical Physics*

> *Gotta Go Fast!*
>
> – Sonic the Hedgehog, *Sonic X Theme Song*

As we have seen in section 2.5, the instantaneous heat flux $\boldsymbol{J}(t)$ is the central quantity required to compute thermal conductivities with the GK method. Its formulation and implementation for different types of potential has been the focus of much previous work.

Initial efforts by Irving and Kirkwood [396], later formalised by Noll [397], and continued by others [398–401] were undertaken within the wider context of establishing a connection between continuum descriptions and statistical mechanics. These early works assumed additive pairwise potentials, leading to first GK simulations with such potentials [201, 366, 367]. Shortly after, Hardy established a form of the heat flux for periodic quantum systems [381] without this assumption. However, this form was not straightforward to apply to the non-pairwise many-body FFs (see section 2.2.3) developed in the late 1980s [108–110]. This lead to the development of many differing forms of the heat flux for such cases, often based on extensions of stress-based formulas developed for pairwise potentials [368–371].

It has only recently been widely acknowledged that such many-body potentials, or, more precisely, those which are not composed of additive pairwise terms, require a fundamentally different treatment. Fan et al. [371] showed that the classical equivalent of the formulation by Hardy [381] provides a general heat flux, unifying previous formulations for many-body potentials, and highlighting the distinction between stress-based pairwise and general Hardy-like formulas. Boone et al. [402] investigated previous heat flux formulations in the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) code, finding that a heat flux for pairwise potentials is incorrect for many-body potentials. Based on a derivation by

Toori et al. [403], they provide a heat flux for three- and four-body potentials. Surblys et al. [404] provide an alternative 'centroid' formulation for LAMMPS, also based on the work by Toori et al. [403].

In parallel, heat flux formulations for DFT were developed, enabling aiGK calculations [143, 151].

THIS COMPLEX LANDSCAPE of different forms of the heat flux, and different perspectives on its derivation, presents a challenge for the present work. Here, we seek to develop a heat flux for MLIPs in general and GLPs in particular, under the constraint that derivatives are computed with AD, rather than manually implemented. While some existing forms can be applied in principle to this case, doing so is not straightforward and can lead to high computational cost, as we will highlight over the course of this section. Nomenclature differs significantly across existing forms, and periodicity is often treated only implicitly, leading to potential ambiguities in implementation.

A discussion of selected heat flux formulations and their relationship to the ones discussed in this work is given in appendix C.6.

IN THIS SECTION, we aim to clarify the formulation of the heat flux for classical many-body potentials, including semi-local GLPs, and provide guidelines for the efficient and simple implementation of the result with AD. An instructive implementation in `jax` [37] is provided in the `glp` package (see appendix E.3).

To this end, we proceed in two stages: In the first part of this section, sections 4.2.1 to 4.2.5, the Hardy heat flux formulation is re-derived with a particular emphasis on an explicit treatment of periodicity and consistent nomenclature. In the second part, sections 4.2.6 to 4.2.9, strategies for the efficient implementation of the result are discussed, and a linear-scaling formulation using AD for local and semi-local GLPs is developed. Section 4.2.10 provides a summary of results.

4.2.1 Setting the Problem

The heat flux required for the GK method was introduced in section 2.5 as the spatial integral of a heat current density $\boldsymbol{j}(\boldsymbol{r}, t)$, which arises from a continuity equation, equation (2.5.5), for the energy density $e(\boldsymbol{r}, t)$, which in turn is defined as a scalar field that integrates to the total energy of the system E. Obtaining the heat flux therefore requires *(a)* making an ansatz for $e(\boldsymbol{r}, t)$, *(b)* solving the continuity equation for $\boldsymbol{j}(\boldsymbol{r}, t)$, and *(c)* executing the spatial integral to obtain $\boldsymbol{J}(t)$.

Overall, these steps follow a later approach by Hardy [398], which the present works extends to non-pairwise potentials. Explicitly separating the computation of the heat current density from the integration allows us to introduce periodicity in a controlled manner, using the notation established in section 2.3.

4.2.2 Defining the Energy Density

Conceptually, defining $e(\boldsymbol{r})$ requires a formal connection between a continuum (or *hydrodynamical*) description in terms of spatially continuous densities, and the statistical mechanics (or MD) description in terms of statistical ensembles of point particles. While details vary across methods,[1] the essential idea is to use *localisation functions* $\Delta(\boldsymbol{r}_i - \boldsymbol{r})$, which are peaked at $\boldsymbol{r}_i$, decay to 0 as $|\boldsymbol{r} - \boldsymbol{r}_i|$ increases, and are normalised to 1, to formally map quantities assigned to discrete atoms i to densities.

In the case of inhomogeneous systems with atomically sharp interfaces or non-equilibrium approaches, the microscopic[2] details of localisation function and averaging approach gain central importance [405]. The GK method, however, is concerned with system-wide averages and equilibrium MD, and we therefore use generic localisation functions, not relying on any particular form.

The usage of localisation functions requires the ability to assign contributions to the total energy E to atomic positions,[3] which is straightforward for the kinetic energy[4] T, but not for the potential energy U.

To proceed, we restrict the present derivation to potentials constructed *a priori* from atomic contributions

$$U(\mathcal{R}) = \sum_{i=1}^{N} U_i(\mathcal{R})\,. \tag{4.2.1}$$

For now, these contributions can be an arbitrary many-body function of all N atomic positions $\mathcal{R}$. This setting differs from the case of DFT, where an energy density can be defined [221], but no direct decomposition of the total energy into atomic contributions is available.[5] Previous discussions of the heat flux have been concerned with the question of uniqueness arising from the arbitrary partitioning of a given total potential energy into atomic contributions. For potentials explicitly composed of two- or three-body contributions, numerical experiments [406] and the gauge principle [229] have shown that the exact composition of atomic contributions has no impact on the thermal conductivity. In the case of MLIPs, U_i may be many-body functions without any further decomposition into k-body contributions. Therefore, a given partitioning into U_i must be considered as-is.[6]

With these considerations, we are now in a position to make the ansatz for the energy density

$$e(\boldsymbol{r}) = \sum_{i=1}^{N} \Delta(\boldsymbol{r}_i - \boldsymbol{r})\,(U_i + T_i) = \sum_{i=1}^{N} \Delta(\boldsymbol{r}_i - \boldsymbol{r}) E_i\,. \tag{4.2.2}$$

4.2.3 Solving the Continuity Equation

The next step is solving equation (2.5.5), which we accomplish by rewriting the time derivative of $e(\boldsymbol{r})$ in a form that can be directly identified as the divergence of the heat current density $\boldsymbol{j}(\boldsymbol{r})$, and therefore the solution of the continuity equation.

We now adopt the notation of section 2.2, treating a system with N atoms contained in a volume V; the bulk limit will be considered later. Recall that the energy depends on time through atomic positions (potential energy) and the velocities (kinetic energy); there is no explicit time dependence in this setting. We therefore drop explicit mentions of t, for instance writing $e(\boldsymbol{r})$ instead of $e(\boldsymbol{r}, t)$.

[1] An early introduction to such approaches is given by Noll [397]. A recent overview can be found in reference [401].

[2] On the scale of single atoms, as opposed to the simulation cell.

[3] We note in passing that one does not necessarily need to rely on atomic positions. Recent work by Surblys et al. [404], for instance, assigns potential energy contributions arising from groups of atoms to their centre of mass. In section 4.2.5, we will also see that in solids, it is useful to assign atomic potential energies to fixed, rather than instantaneous, positions. Nevertheless, access to some form of contributions to U is required.

[4] Recall that in the framework of MD as described in section 2.2, nuclei are treated as classical point particles with kinetic energy $T_i = 1/2\, m_i v_i^2$.

[5] The derivation of the heat flux therefore proceeds differently. Carborgno et al. [143] exploit the structure of the Hamiltonian in equation (2.1.6), which consists of one- and two-body terms involving atomic and electronic degrees of freedom, to define atomic contributions to the Hellmann-Feynman forces and thereby obtain a heat flux for solids. Marcolongo et al. [151] proceed from the energy density directly, obtaining a numerically more involved heat flux.

[6] Re-partitioning can in principle be performed by defining an additional function that redistributes energy between atoms, which may be advantageous for the purpose of GK convergence [220]. In that case, the thermal conductivity remains unchanged if such a change corresponds to a non-diffusive flux as discussed in section 2.5.3. At present, this is not pursued.

To this end, we first define a *bond function*:[7]

$$\Lambda_{ij}(\boldsymbol{r}) = \int_0^1 \mathrm{d}\lambda\, \Delta(\lambda \boldsymbol{r}_i + (1-\lambda)\boldsymbol{r}_j - \boldsymbol{r})\,. \tag{4.2.3}$$

One can then show that[8]

$$\Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \Delta(\boldsymbol{r}_j - \boldsymbol{r}) = \boldsymbol{r}_{ij} \cdot \boldsymbol{\nabla}_{\boldsymbol{r}} \Lambda_{ij}(\boldsymbol{r})\,. \tag{4.2.4}$$

With this identity, any term that can be written in terms of a difference at two locations $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$ can be identified as a heat flux along the line segment between them.

The time-derivative of $e(\boldsymbol{r})$ is

$$\frac{\mathrm{d}}{\mathrm{d}t} e(\boldsymbol{r}, t) = \sum_{i=1}^{N} \left(\frac{\mathrm{d}}{\mathrm{d}t} E_i\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) + \sum_{i=1}^{N} E_i \left(\frac{\mathrm{d}}{\mathrm{d}t} \Delta(\boldsymbol{r}_i - \boldsymbol{r})\right)\,. \tag{4.2.5}$$

The first term can be tackled by splitting $E_i = U_i + T_i$ and resolving the time derivative (see appendix C.8)

$$\sum_{i=1}^{N} \left(\frac{\mathrm{d}}{\mathrm{d}t} E_i\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) \tag{4.2.6}$$

$$= \sum_{i,j=1}^{N} \left[\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \left(\Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \Delta(\boldsymbol{r}_j - \boldsymbol{r})\right)\right]\,, \tag{4.2.7}$$

and finally applying the identity from equation (4.2.4) to obtain

$$= \boldsymbol{\nabla}_{\boldsymbol{r}} \cdot \left[\sum_{i,j=1}^{N} \boldsymbol{r}_{ij} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \Lambda_{ij}(\boldsymbol{r})\right]\,. \tag{4.2.8}$$

The second term is resolved by the chain rule

$$\sum_{i=1}^{N} E_i \left(\frac{\mathrm{d}}{\mathrm{d}t} \Delta(\boldsymbol{r}_i - \boldsymbol{r})\right) = -\sum_{i=1}^{N} E_i \Big(\boldsymbol{\nabla}_{\boldsymbol{r}} \Delta(\boldsymbol{r}_i - \boldsymbol{r}) \cdot \boldsymbol{v}_i\Big) \tag{4.2.9}$$

$$= -\boldsymbol{\nabla}_{\boldsymbol{r}} \cdot \Big(\sum_{i=1}^{N} E_i \Delta(\boldsymbol{r}_i - \boldsymbol{r}) \boldsymbol{v}_i\Big)\,. \tag{4.2.10}$$

Comparing with equation (2.5.5), we find

$$\boldsymbol{j}(\boldsymbol{r}) = \sum_{i,j=1}^{N} \boldsymbol{r}_{ji} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \Lambda_{ij}(\boldsymbol{r}) + \sum_{i=1}^{N} E_i \boldsymbol{v}_i \Delta(\boldsymbol{r}_i - \boldsymbol{r})\,. \tag{4.2.11}$$

WHILE SO FAR WE have relied on purely algebraic manipulations, this result can also be justified from a physical perspective. Any energy change at atom i must be balanced out by corresponding changes in the atoms j it interacts with, with an energy current flowing between them.[9] If we can divide up the change in U_i into contributions that can be attributed to different j, and identify the corresponding terms in U_j, we know the magnitude of the current between i and j. It is a natural assumption that the current flows directly between them, parallel to $\boldsymbol{r}_{ij}$. The first term in equation (4.2.11) is one way to construct a vector field accordingly. The second term describes an alternative process to change local distribution of energy: Rather than energy flowing between atoms, the energy situated at a given atom is 'dragged along' by the atom moving. Together, these two processes describe how the energy density can change, and therefore yield the heat current density, solving the continuity equation.

[7] In essence, a bond function gives a contribution only where $\boldsymbol{r}$ lies on the line segment between $\boldsymbol{r}_i$ and $\boldsymbol{r}_j$. It can be seen as the two-dimensional equivalent of the localisation function.

[8] A more rigorous derivation of similar relations is given by Noll in [397], based on the work of Irving and Kirkwood [396]. A short proof is provided in appendix C.7.

Note that this manipulation relies on the ability to freely rename i and j, as both appear in a sum running over $1...N$. For this reason, we have not yet transitioned to a periodic system.

Note the flipped signs.

[9] This does not mean that pairwise fluxes are always zero – this is only the case if $\partial U_i/\partial \boldsymbol{r}_j = \partial U_j/\partial \boldsymbol{r}_i$, which only applies to pair potentials. In general, the energy change in i is divided up in some way between its neighbours.

4.2.4 Heat Flux in the Bulk

At this point, $\boldsymbol{j}(\boldsymbol{r})$ has been obtained for N particles with open boundary conditions. As discussed in section 2.3, practical simulations of bulk systems, however, are performed in a periodic setting. We therefore adopt the notation established in that section, obtaining

$$\boldsymbol{j}^{\text{bulk}}(\boldsymbol{r}) = \sum_{\substack{i\in\mathcal{R}_{\text{all}}\\ j\in\mathcal{R}_{\text{all}}}} \boldsymbol{r}_{ji} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j \right) \Lambda_{ij}(\boldsymbol{r}) + \sum_{i\in\mathcal{R}_{\text{all}}} E_i \boldsymbol{v}_i \Delta(\boldsymbol{r}_i - \boldsymbol{r})\,. \tag{4.2.12}$$

The result is illustrated in figure 4.2.1. As this current density is defined

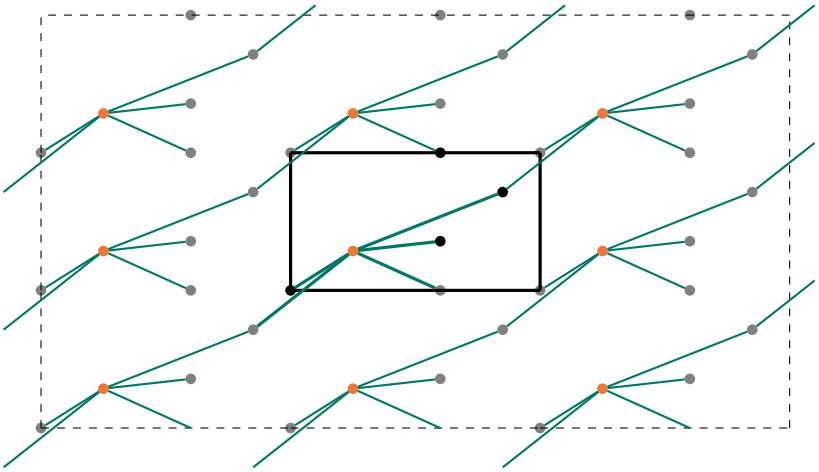


Figure 4.2.1: Illustration of the first term of the bulk current density in equation (4.2.12), showing bond functions as lines connecting atoms. The system from figure 4.1.2 is used, assuming that U_i depends on interactions up to M=2. Only bond functions involving the atom highlighted in orange and its replicas are shown. The simulation cell and a 3×3 supercell are shown as solid and dashed lines, respectively.

directly on the bulk system, it is not sensitive to the particular choice of simulation cell; the issue of boundary invariance has been avoided. Crucially, each bond function between pairs of symmetrically equivalent positions appears exactly once in each cell.

Therefore, the integral over any simulation cell,[10] yields

$$\boldsymbol{J} = \int_{\text{s.c.}} \mathrm{d}^3\boldsymbol{r}\, \boldsymbol{j}^{\text{bulk}}(r) \tag{4.2.13}$$

$$= \sum_{\substack{i\in\mathcal{R}_{\text{sc}}\\ j\in\mathcal{R}_{\text{all}}}} \left(\boldsymbol{r}_{ji} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j \right) \right) + \sum_{i\in\mathcal{R}_{\text{sc}}} E_i \boldsymbol{v}_i \tag{4.2.14}$$

$$=: \boldsymbol{J}_{\text{pot}} + \boldsymbol{J}_{\text{conv}}\,, \tag{4.2.15}$$

where we have used the fact that all contributions where $i \notin \mathcal{R}_{\text{sc}}$ can be computed equivalently for the replica of i in the simulation cell.[11]

We note in passing that the contributions in front of $\boldsymbol{v}_i$ in the first term of equation (4.2.14) are, in general, not equivalent to those appearing in the stress formulations of equations (4.1.18) to (4.1.20). This heat flux is also *not* the time-derivative of a barycentre[12] $\boldsymbol{B}$. Indeed, for the purposes of the HE relation, the barycentre must be defined as the time integral of the heat flux [146, 147].

With this, we have obtained the desired general form of the heat flux. It is similar to the classical equivalent of the Hardy formula [381],

[10] We assume a fixed number of atoms in the simulation cell. Supercells pose no additional challenge, but simply add a multiplier.

The terminology of these different contributions to the heat flux is discussed in section 4.2.5.

[11] The derivatives of U_i for $i \notin \mathcal{R}_{\text{sc}}$ are identical to the ones of $i \in \mathcal{R}_{\text{sc}}$, except for re-indexing the positions $\boldsymbol{r}_j$ that contribute. $\boldsymbol{r}_{ji}$ are invariant under that change, and so is $\boldsymbol{v}_j$.

[12] If the sum over j would instead run over $\mathcal{R}_{\text{sc}}$, we would recover the time-derivative of the energy barycentre of a non-periodic system. However, the resulting heat flux term $\boldsymbol{J}_{\text{pot}}$ would be non-diffusive. For this reason, this derivation has taken care to explicitly introduce the ranges of all involved sums.

as given, for instance in reference [371, eq. B3], but includes an explicit treatment of periodicity. It is valid, in principle, for any potential of the form in equation (4.2.1).

4.2.5 Terminology of Heat Flux Contributions

The heat flux in equation (4.2.15) consists of two terms: The 'potential' term $\boldsymbol{J}_{\text{pot}}$ describes the flow of energy between pairs of atoms, while the 'convective' term $\boldsymbol{J}_{\text{conv}}$ describes the transport of energy located at individual atoms through their movement.[13] This standard terminology is somewhat deceptive, as pointed out by Ercole [407]: $\boldsymbol{J}_{\text{conv}}$ cannot be neglected even if no convection occurs, as its cross-correlation with $\boldsymbol{J}_{\text{pot}}$ may be non-vanishing [408]. In appendix D.5, this is shown for the case of zirconia at varying temperatures.

Instead, in solids, where atomic positions stay bounded over time, it is useful to decompose $\boldsymbol{J}$ differently [212, 367]:

$$\boldsymbol{J} = \sum_{\substack{i\in\mathcal{R}_{\text{sc}}\\ j\in\mathcal{R}_{\text{all}}}} \left(\boldsymbol{r}_{ji}^{0} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j \right) \right) + \frac{\mathrm{d}}{\mathrm{d}t} \sum_{i\in\mathcal{R}_{\text{sc}}} \boldsymbol{u}_i E_i \tag{4.2.16}$$

$$=: \boldsymbol{J}_{\text{int}} + \boldsymbol{J}_{\text{disp}}\,, \tag{4.2.17}$$

introducing fixed reference positions $\boldsymbol{r}_i^0$ and displacements $\boldsymbol{u}_i(t)$ from these positions. The 'interaction' term $\boldsymbol{J}_{\text{int}}$ contains the components of $\boldsymbol{J}_{\text{pot}}$ that determine the thermal conductivity in solids, the remaining 'displacement' term $\boldsymbol{J}_{\text{disp}}$ is non-diffusive if $\boldsymbol{u}_i$ and E_i are bounded. This is demonstrated, again for zirconia, in appendix D.5.

4.2.6 Implementing the Heat Flux

As E_i is directly available in the present setting, $\boldsymbol{J}_{\text{conv}}$ presents no further difficulty.[14] $\boldsymbol{J}_{\text{pot}}$, however, requires disentangling the contributions of every atom, including those in the bulk, to every atomic potential energy U_i, in a double sum with $|\mathcal{R}_{\text{sc}}| \cdot |\mathcal{R}_{\text{all}}|$ terms.

For general many-body potentials, where a direct decomposition of U_i into pairwise contributions is not available, this presents a challenge, and has lead to the development of a variety of specialised expressions surveyed in the introduction of this section. Let us now review strategies for tackling it.

We restrict the present discussion to potentials with a maximum interaction cutoff radius,[15] $r_{\text{c}}^{\text{eff}}$. In other words, potentials where

$$U_i = U_i(\{\, \boldsymbol{r}_j \mid \boldsymbol{r}_j \in \mathcal{R}_{\text{all}}, |\boldsymbol{r}_{ij}| \leq r_{\text{c}}^{\text{eff}} \,\}) \tag{4.2.18}$$

$$\Rightarrow \frac{\partial U_i}{\partial \boldsymbol{r}_j} = 0 \quad \forall \boldsymbol{r}_j \in \mathcal{R}_{\text{all}} \quad \text{where} \quad |\boldsymbol{r}_{ij}| > r_{\text{c}}^{\text{eff}}\,. \tag{4.2.19}$$

With this restriction, the sum over $\mathcal{R}_{\text{all}}$ reduces to a sum over $\mathcal{R}_{\text{unf}}$, the 'unfolded' simulation cell, which contains the simulation cell itself and all replica positions within a shell of width $r_{\text{c}}^{\text{eff}}$ (see section 4.1.3).

Even with this restriction, however, the sum still contains $O(|\mathcal{R}_{\text{sc}}| \cdot |\mathcal{R}_{\text{unf}}|) = O(N^2)$ terms, as $|\mathcal{R}_{\text{unf}}| \approx N + N^{2/3}$. Its naive evaluation

[13] In the literature, different naming conventions for these terms exist. For instance, Fan et al. [371] refer to $\boldsymbol{J}_{\text{conv}}$ as 'kinetic' term, while Carbogno et al. [143] refer to $\boldsymbol{J}_{\text{pot}}$ as 'conductive' or 'virial' term. We do not adopt this terminology as $\boldsymbol{J}_{\text{pot}}$ is not composed of virials, i.e., contributions to the stress, in all cases: For GLPs with M>1, the terms appearing in the heat flux no longer sum to the stress.

Intermediate steps and further details can be found in appendix C.9.

For the sake of generality, we will proceed with $\boldsymbol{J}_{\text{pot}}$ instead of $\boldsymbol{J}_{\text{int}}$. Results can be equivalently obtained for $\boldsymbol{J}_{\text{int}}$ by substituting the $\boldsymbol{r}$ prefactors with $\boldsymbol{r}^0$. Outside of this section, we will often use $\boldsymbol{J}$ to refer to refer to both $\boldsymbol{J}_{\text{int}}$ and the full heat flux $\boldsymbol{J}_{\text{pot}} + \boldsymbol{J}_{\text{conv}}$.

[14] From a computational perspective, $\boldsymbol{J}_{\text{conv}}$ simply requires a sum over all atoms in the simulation cell, re-using quantities that have already been computed to predict the energy. Therefore, if the potential scales as $O(N)$, so does the computation of $\boldsymbol{J}_{\text{conv}}$. However, its implementation still presents additional complexity, which is why it was avoided in the initial implementation of the heat flux used for SchNet and zirconia. The re-implementation in `glp` always computes the full heat flux.

[15] The case of unrestricted sums over the bulk is beyond the scope of the present work. In some cases, for instance known point charges and pairwise Coulomb interactions [409], Ewald summation [99] can be used to compute the heat flux.

An estimate of the number of additional positions is given in appendix C.5.

would therefore scale quadratically with system size,[16] rendering it undesirable for the GK method. However, we have not yet considered the structure of the Jacobian $\partial U_i / \partial \boldsymbol{r}_j$. Due to equation (4.2.19), it exhibits a known sparsity pattern: Only terms where $|\boldsymbol{r}_{ij}| \leq r_\mathrm{c}^\mathrm{eff}$ are nonzero. The number of such terms is $O(N)$, as the size of the N neighbourhoods of each i remains constant as N is increased at constant density.

In principle, we can therefore expect to be able to compute $\boldsymbol{J}_\mathrm{pot}$ with computational cost that scales linearly with system size. Indeed, in section 4.2.9, an AD-based linear-scaling approach to implementing $\boldsymbol{J}_\mathrm{pot}$ is presented, which we term the 'unfolded' heat flux $\boldsymbol{J}_\mathrm{pot}^\mathrm{unfolded}$. In this approach, $\mathcal{R}_\mathrm{unf}$ is explicitly constructed.

However, this requires a modification of the implementation of a given potential. For instance, GLPs as introduced in section 4.1 avoid the construction of replica positions beyond r_c entirely; they are functions of $(\mathcal{R}_\mathrm{sc}, \mathcal{B})$, or, equivalently, of $\mathcal{G}$. In order to better understand the heat flux for such cases, we also consider the heat flux for potentials which are explicitly constructed from $(\mathcal{R}_\mathrm{sc}, \mathcal{B})$, in section 4.2.7, and for GLPs, in section 4.2.8. We find that only local GLPs, where $r_\mathrm{c}^\mathrm{eff}{=}r_\mathrm{c}$ and $M{=}1$, admit an efficient formulation of the heat flux; for semi-local GLPs, the 'unfolded' approach is preferable.

[16] In these considerations, we implicitly assume that the computational cost of the potential under consideration scales linearly with N, or, more precisely, the computational cost of obtaining any given U_i remains constant as N is increased at constant density. In other words, we largely focus on GLPs, where this is ensured by construction.

4.2.7 Heat Flux with Minimum Image Convention

If $r_\mathrm{c}^\mathrm{eff}{<}r_\mathrm{c}^\mathrm{max}$, each U_i only depends on at most one replica of each $\boldsymbol{r}_j$.[17] In that case, the partial derivative in equation (4.2.14) can be computed with respect to the equivalent position in the simulation cell. However, the atom-pair vector $\boldsymbol{r}_{ji}$ must still connect $\boldsymbol{r}_i$ in the simulation cell with the respective position $\boldsymbol{r}_j$ which may be a replica.[18] This can be ensured by adopting the MIC, yielding

$$\boldsymbol{J}_\mathrm{pot}^\mathrm{mic} \coloneqq \sum_{i,j \in \mathcal{R}_\mathrm{sc}} \left(\boldsymbol{r}_{ji}^\mathrm{mic} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j \right) \right) . \tag{4.2.20}$$

While this form of the heat flux requires no replica positions, it is also unsuitable for the efficient implementation with AD: As different factors are multiplied with each entry of the Jacobian $\partial U_i / \partial \boldsymbol{r}_j$, the evaluation of this heat flux requires the computation of the explicit full Jacobian, leading to quadratic scaling as discussed in section 2.4.6.

However, as it requires no modification in the implementation of a given potential, and can be implemented directly, albeit inefficiently, with AD, we use $\boldsymbol{J}_\mathrm{pot}^\mathrm{mic}$ as baseline for the development of more specialised approaches.

[17] Recall the definition of $r_\mathrm{c}^\mathrm{max}$ in equation (2.3.4).

[18] If $\boldsymbol{r}_{ji}$ is simply taken between positions in the simulation cell, the resulting heat flux is bounded and therefore yields vanishing thermal conductivity.

4.2.8 Heat Flux for Graph Machine-Learning Potentials

For GLPs, U_i is a function of all edges within M hops on the graph:

$$
\begin{aligned}
U_i^{M=1} &= U_i\left(\left\{ \boldsymbol{r}_{ij}^{\text{mic}} \;\middle|\; j \in \mathcal{N}(i) \right\}\right) \\
U_i^{M=2} &= U_i\left(\left\{ \boldsymbol{r}_{ij}^{\text{mic}} \;\middle|\; j \in \mathcal{N}(i) \right\} \cup \left\{ \boldsymbol{r}_{jk}^{\text{mic}} \;\middle|\; j \in \mathcal{N}(i), k \in \mathcal{N}(j) \right\}\right) \\
U_i^{M=3} &= \dots
\end{aligned}
$$

Recall that $\mathcal{N}(i)$ denotes all vertices within r_c, and that edges $\boldsymbol{r}_{ij}$ as defined in section 4.1 have been computed with the MIC. All atom-pair vectors corresponding to edges are therefore subject to the MIC, and marked as such.

We consider the local case, M=1, first. Simply substituting in equation (4.2.14) yields the 'local' heat flux

$$
\boldsymbol{J}_{\text{pot}}^{\text{local}} \coloneqq \sum_{ij\in\mathcal{E}} \left(\boldsymbol{r}_{ji}^{\text{mic}} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_{ij}^{\text{mic}}} \cdot \boldsymbol{v}_j \right) \right) . \tag{4.2.21}
$$

The form in equation (4.2.21) was derived by Fan et al. [371] using a slightly different argument: In general, the set of atom-pair vectors between $\mathcal{R}_{\text{sc}}$ and $\mathcal{R}_{\text{all}}$ can be used to define a separate set of inputs for each U_i, without referring to the additional structure of a GLP. Therefore, equation (4.2.21) should in principle be applicable to any potential, not only GLPs with M=1. In practice, however, this would require the construction of separate neighbourhoods with size r_c^{eff} for each atom i and then separate message-passing steps within each enlarged environment. This would compromise the computational efficiency gained by the semi-local structure of GLPs, which are explicitly constructed to avoid the consideration of such large environments, and share computational work between adjacent neighbourhoods.

While this expression contains only $|\mathcal{E}|$ terms, it still requires the explicit evaluation of each term in the Jacobian $\partial U_i / \partial \boldsymbol{r}_{ij}^{\text{mic}}$, rendering it quadratically scaling when implemented with AD. This problem can be alleviated by recognising that for M=1, the inputs to each U_i are separate, as the atom-pair vectors in $\mathcal{N}(i)$ are used *only* in the computation of U_i. Therefore

$$
\frac{\partial U_i}{\partial \boldsymbol{r}_j} = \frac{\partial U_i}{\partial \boldsymbol{r}_{ij}^{\text{mic}}} = \frac{\partial U}{\partial \boldsymbol{r}_{ij}^{\text{mic}}} ; \tag{4.2.22}
$$

it is sufficient to compute the *gradient* of U with respect to $\mathcal{E}$. This yields the a form of the heat flux that relies exclusively on the *edges* in the graph,

$$
\boldsymbol{J}_{\text{pot}}^{\text{edges}} \coloneqq \sum_{ij\in\mathcal{E}} \left(\boldsymbol{r}_{ji}^{\text{mic}} \left(\frac{\partial U}{\partial \boldsymbol{r}_{ij}^{\text{mic}}} \cdot \boldsymbol{v}_j \right) \right) . \tag{4.2.23}
$$

This 'edges only' formulation of the heat flux is attractive from an implementation standpoint: It requires only derivatives that can be obtained with a single gradient computation, and does not require the construction of an unfolded simulation cell. As seen in figure 4.2.2, while overall scaling is similar to the unfolded heat flux, it is substantially faster. It is therefore tempting to use it even for $M \geq 1$, where it is not equivalent to $\boldsymbol{J}_{\text{pot}}$. However, as seen in figure D.24, predictions with this form differ from the full heat flux. This difference of approximately 5 % across temperatures does not reduce with convergence (not shown). Investigating to what extent $\boldsymbol{J}_{\text{pot}}^{\text{edges}}$ can be used as an approximation for the full heat flux is left for future work.

For local GLPs with M=1, we have therefore obtained an alternative form of the heat flux that can be efficiently implemented using AD.

For semi-local GLPs with $M \geq 1$, atom-pair vectors are *shared* between atomic energy contributions, and equation (4.2.22) does not apply. Instead, we can compute the heat flux with equation (4.2.20), or equivalently as

$$
\boldsymbol{J}_{\text{pot}}^{\text{semi-local}} \coloneqq \sum_{\substack{i\in\mathcal{V} \\ jk\in\mathcal{E}}} \boldsymbol{r}_{ji}^{\text{mic}} \left(\left[\frac{\partial U_i}{\partial \boldsymbol{r}_{kj}^{\text{mic}}} - \frac{\partial U_i}{\partial \boldsymbol{r}_{jk}^{\text{mic}}} \right] \cdot \boldsymbol{v}_j \right) . \tag{4.2.24}
$$

Like equations (4.2.20) and (4.2.21), this form requires access to an explicit Jacobian $\partial U_i / \partial \boldsymbol{r}_{kj}^{\text{mic}}$ and therefore scales quadratically when implemented with AD.

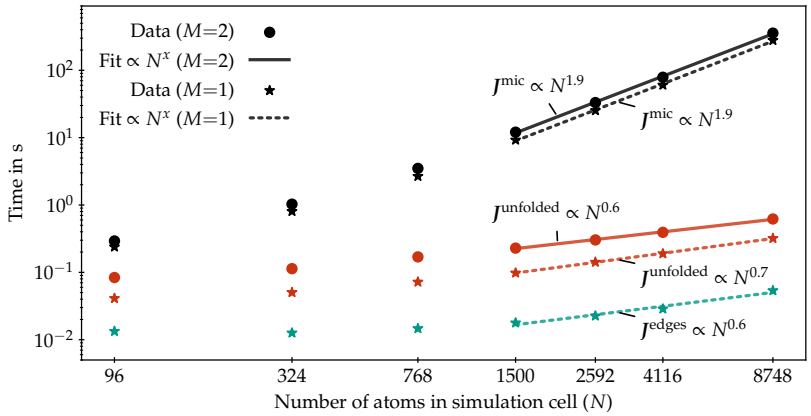


Figure 4.2.2: Computation time per timestep for different system sizes N for zirconia, evaluating SchNet MPNNs with M=1 and M=2, for different heat flux formulations. Only equivalent forms of the heat flux are shown; $\boldsymbol{J}_{\text{pot}}^{\text{semi-local}}$ and $\boldsymbol{J}_{\text{pot}}^{\text{local}}$ have been omitted as they scale identically to $\boldsymbol{J}_{\text{pot}}^{\text{mic}}$. Benchmarks were performed on a single Tesla Volta V100 32GB GPU, computing only $\boldsymbol{J}_{\text{pot}}$. To estimate the asymptotic scaling, a function proportional to N^x has been fitted to the results for large N. Note that on this setup with limited memory, the asymptotic limit cannot be reached.

4.2.9 Unfolded Heat Flux

Finally, we consider the 'unfolded' heat flux, which is obtained by restricting the sum over $\mathcal{R}_{\text{all}}$ in equation (4.2.14) to $\mathcal{R}_{\text{unf}}$,

$$\boldsymbol{J}_{\text{pot}}^{\text{unfolded}} := \sum_{\substack{i\in\mathcal{R}_{\text{sc}} \\ j\in\mathcal{R}_{\text{unf}}}} \left(\boldsymbol{r}_{ji} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j \right) \right) . \tag{4.2.25}$$

As discussed in section 2.4.6, the efficient use of AD requires expressions in the form of either JVPs or VJPs. In other words, our task is to rewrite $\boldsymbol{J}_{\text{pot}}^{\text{unfolded}}$ in equation (4.2.25) such that one sum can be executed inside the derivative.

As shown in appendix C.11, this can be done and yields

$$\boldsymbol{J}_{\text{pot}}^{\text{unfolded}} = \sum_{j\in\mathcal{R}_{\text{unf}}} \frac{\partial \boldsymbol{B}}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j - \sum_{j\in\mathcal{R}_{\text{unf}}} \boldsymbol{r}_j \left(\frac{\partial U}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j \right) , \tag{4.2.26}$$

defining an intermediate quantity

$$\boldsymbol{B} := \sum_{i\in\mathcal{R}_{\text{sc}}} \boldsymbol{r}_i^{\text{const}} U_i . \tag{4.2.27}$$

The first term in equation (4.2.26) is a JVP, the second term a VJP. Therefore, they can be be computed with the same asymptotic computational cost as $\boldsymbol{B}$ and U, respectively. Provided that the computation of U_i is linear in the number of input positions, overall computational cost is $O(|\mathcal{R}_{\text{unf}}|) = O(N + N^{2/3}) = O(N)$. Linear scaling is restored.[19]

The vector-vector product in the first term should be taken between the denominator of the partial derivative and $\boldsymbol{v}_j$, not between $\boldsymbol{B}$ and $\boldsymbol{v}_j$.

Here $\boldsymbol{r}_i^{\text{const}}$ denotes positions that are numerically identical to $\boldsymbol{r}_i$ but are treated as constants during the calculation of derivatives.

[19] We remark that since all positions in $\mathcal{R}_{\text{unf}}$ must be considered explicitly, and since the size of $\mathcal{R}_{\text{unf}}$ scales cubically with the effective cutoff radius, this approach is limited to moderate numbers of interaction steps M. However, we find that in practice, this limitation is not critical, as M=2 is often sufficient for good predictive performance.

4.2.10 Results

In sections 4.2.1 to 4.2.5 we developed a formulation of the heat flux for the purposes of the GK method[20] for potentials decomposed into atomic contributions as defined in equation (4.2.1) and using the notation for periodicity introduced in section 2.3. The result is a variation on the well-known Hardy formula and given in equation (4.2.15).

[20] In other words, without considering the microscopic structure of continuum quantities.

We then considered different ways to make use of knowledge of the particular structure of different potentials to efficiently implement this formula. A number of specialised forms were derived, which are summarised in table 4.2.1.

Name	Equation	Scaling	Prerequisites
$\boldsymbol{J}_{\text{pot}}^{\text{unfolded}}$	4.2.26	$O(N)$	Explicit construction of $\mathcal{R}_{\text{unf}}$
$\boldsymbol{J}_{\text{pot}}^{\text{edges}}$	4.2.23	$O(N)$	GLP, M=1
$\boldsymbol{J}_{\text{pot}}^{\text{local}}$	4.2.21	$O(N^2)$	GLP, M=1
$\boldsymbol{J}_{\text{pot}}^{\text{semi-local}}$	4.2.24	$O(N^2)$	GLP, $M\geq1$
$\boldsymbol{J}_{\text{pot}}^{\text{mic}}$	4.2.20	$O(N^2)$	$r_{\text{c}}^{\text{eff}}<r_{\text{c}}^{\text{max}}$

Table 4.2.1: Summary of heat flux formulations

If the given prerequisites are fulfilled, and $\boldsymbol{J}_{\text{conv}}$ is added, all these formulas are mathematically identical to the general heat flux in equation (4.2.15): For $M\geq1$, $\boldsymbol{J}_{\text{pot}}^{\text{unfolded}}$, $\boldsymbol{J}_{\text{pot}}^{\text{semi-local}}$, and $\boldsymbol{J}_{\text{pot}}^{\text{mic}}$ are equivalent; $\boldsymbol{J}_{\text{pot}}^{\text{edges}}$ and $\boldsymbol{J}_{\text{pot}}^{\text{local}}$ are not. This is shown in figure D.24. For M=1, all formulations in table 4.2.1 are identical, as shown in figure D.25.

The computational cost of selected formulations can be seen in figure 4.2.2. The efficient re-formulation of the heat flux, $\boldsymbol{J}_{\text{pot}}^{\text{unfolded}}$, remains at least one order of magnitude faster than the unoptimised $\boldsymbol{J}_{\text{pot}}^{\text{mic}}$,[21] with computational cost scaling approximately linearly. $\boldsymbol{J}_{\text{pot}}^{\text{edges}}$, which is only applicable to M=1, displays the same scaling of computational cost as $\boldsymbol{J}_{\text{pot}}^{\text{unfolded}}$, at overall reduced computational cost.

[21] And equivalently, $\boldsymbol{J}_{\text{pot}}^{\text{semi-local}}$ and $\boldsymbol{J}_{\text{pot}}^{\text{local}}$.

WITH THIS, WE HAVE established the infrastructure required to efficiently perform GK calculations with semi-local GLPs.

Example implementations of equations (4.2.20), (4.2.23) and (4.2.26) using `jax` are provided in the `glp` package [410]. Additionally, different heat flux formulas are also implemented in `pytorch` and can be found at `https://github.com/sirmarcel/gknet-archive`.

Section 4.3
Summary

This chapter discussed the implementation of the GK method for the calculation of thermal conductivities with MLIPs based on MPNNs. Such simulations require access to the heat flux, a quantity computed from derivatives of atomic contributions to the potential energy.

In order to efficiently compute the required derivatives for such models with AD, a general understanding of the 'forward' computation is required. To this end, in section 4.1, the concept of graph potentials, or GLPs, was introduced. An account of their construction and the calculation of forces and stress with AD was provided. We verified that a number of previous formulations of the stress are equivalent and can be implemented directly with AD. This section also introduced the notion of local and semi-local potentials: While in the former, interactions take place entirely within atomic neighbourhoods with cutoff radius r_c, the latter admits repeated interactions between such neighbourhoods, leading to correlations up to $r_c^{\text{eff}} \geq r_c$.

Finding that prior formulations of the heat flux are not suitable for semi-local GLPs and AD, adapted forms of the heat flux were derived in section 4.2: The 'unfolded' heat flux in equation (4.2.26) applies to semi-local potentials, and the 'edges' heat flux in equation (4.2.23) applies to local potentials. Using AD, the computational cost for both scales asymptotically linearly with the number of atoms N in the simulation cell,[1] enabling the practical use of MLIPs based on MPNNs for the GK method.

[1] Provided the density is held constant as N is increased.

Limitations

The framework developed in this chapter explicitly excludes non-local interactions, which are present, for instance, in MLIPs that feature attention mechanisms,[2] or those that model all-to-all interactions,[3] as well as in long-range electrostatics. None of the provided heat flux formulations apply to such models, for which equation (4.2.14) must be evaluated explicitly.[4]

[2] Such as SpookyNet [130] or So3krates [43].

[3] Such as (s)GDML [127–129].

Beyond this fundamental restriction, the 'unfolded' heat flux is limited in practice by the need to explicitly construct an extended non-periodic system: As the effective cutoff radius r_c^{eff} increases,[5] the number of additional positions grows cubically. In the experiments reported in the following chapter, N is therefore limited to ≈ 5000 atoms, and M to ≤ 3. However, for the studied systems, little advantage is found for $M > 2$.

[4] In the case of the forces, equation (4.1.6) applies, and for the stress, it is preferable to rely on end-to-end AD, for instance equation (4.1.15). Care must be taken to construct replica positions explicitly in that case.

[5] For instance through higher numbers of message-passing steps M.

Chapter 5
Thermal Conductivity with Message-Passing Neural Networks

> … who scribbled all night rocking and rolling over lofty incantations which in the yellow morning were stanzas of gibberish …
>
> –Allen Ginsberg, *Howl*

In the previous chapter, we established a method to compute the heat flux for semi-local MLIPs, providing the missing piece to use recently developed MLIPs based on MPNNs for thermal conductivity calculations with the GK method. In this chapter, we apply this ability to the calculation of the thermal conductivity of selected materials.

First, in sections 5.1 to 5.3, zirconia (ZrO_2) is studied in detail. Zirconia, alongside silicon, was one of the only two crystalline materials that had been investigated with aiMD when this project was started, in a work by Carbogno et al. [143] that introduced a heat flux definition and a size extrapolation scheme for DFT.[1] Its highly anharmonic PES [414, 415] make it a prototypical candidate for treatment with the GK method. It has also recently been investigated with a local MLIP based on a SOAP-like descriptor and Bayesian regression by Verdi et al. [163], as well as a FF based on the Buckingham potential [416]. Experimental measurements of thermal conductivity, surveyed in appendix D.8, are also available. This prototypical material can therefore serve as a test case.

[1] Recently, this number was increased to approximately 60 materials in a large-scale effort by Knoop et al. [411] (see section 5.4). Additionally, Li_3ClO [412] and ICE-X [413] were recently investigated.

In section 5.1, the training and evaluation of a GLP using the SchNet architecture [41, 42] is discussed: A simple training scheme based on NpT aiMD simulations yields a potential that remains stable up to 1800 K. In this temperature range, it can be used to predict thermal conductivities in the monoclinic and tetragonal phases of zirconia, using a workflow based on work by Knoop et al. [144], discussed in section 5.2. The results are shown in section 5.3 and are consistent with other experimental and computational studies. Using this potential, we also verify the heat flux formulations of section 4.2.

Having established workflows and the heat flux, we then further explore the practical feasibility of GK calculations with GLPs in section 5.4. Using the recently developed equivariant So3krates GLP [43], thermal conductivity for two additional materials is investigated: tin selenide (SnSe) and silicon (Si).

These materials were chosen to cover the case of high and low anharmonicity, and correspondingly low and high thermal conductivity, compared to zirconia. Anharmonicity can be quantified with the anharmonicity measure[2] σ^{A} introduced by Knoop et al. [417]. For SnSe, $\sigma^{\mathrm{A}} = 0.350$ at 300 K [142], and for Si $\sigma^{\mathrm{A}} = 0.177$ at 400 K. ZrO_2, in comparison, is slightly more anharmonic than Si in the monoclinic phase at 300 K, with $\sigma^{\mathrm{A}} = 0.183$, and strongly anharmonic in the tetragonal phase at 1400 K, with $\sigma^{\mathrm{A}} = 0.565$.[3]

For SnSe at 300 K, a So3krates potential trained with 3000 reference calculations is in good agreement with experiments, size-extrapolated aiGK results, and MLIP results for the thermal conductivity at 300 K. For Si at 400 K, i.e., in the regime of low anharmonicity, size convergence is found to pose a challenge: Full size convergence could not be achieved with the currently available implementation of the heat flux. The predicted value of κ underestimates experimental values by approximately 45 %. In addition to convergence issues, biased sampling of the PES is conjectured as a possible cause.

[2] In a nutshell, it measures the deviation between forces obtained with a harmonic model and those obtained with DFT. $\sigma^{\mathrm{A}} \leq 0.2$ is considered 'very harmonic' in reference [417].

[3] Details on the calculation of σ^{A} for Si and ZrO_2 can be found in appendix D.1.

Related publications

Some results in this chapter have been submitted for publication:

"Heat flux for semilocal machine-learning potentials,"
by Marcel F. Langer, Florian Knoop, Christian Carbogno, Matthias Scheffler, and Matthias Rupp
in Physical Review B *in press*
`arXiv:2303.14434`
Referenced as [45].

"Stress and heat flux via automatic differentiation,"
by Marcel F. Langer, J. Thorben Frank, and Florian Knoop
in revision
`arXiv:2305.01401`
Referenced as [46].

In particular, results for ZrO_2 have been reported in reference [45], and results for SnSe have been reported in reference [46].

Data and Code Availability

Data and code related to the reported results can be found at
`doi:10.5281/zenodo.7767432` for zirconia,
`doi:10.5281/zenodo.7852530` for tin selenide, and
`doi:10.5281/zenodo.7963152` for silicon.

First-principles calculations are additionally available at
`doi:10.17172/NOMAD/2023.03.24-2` for zirconia, and
`doi:10.17172/NOMAD/2023.06.04-1` for silicon.

Section 5.1
Potential for Zirconia

Deep learning still comes from approximately ten thousand hours of work on any given subject.

– Hannu Rajaniemi, *The Quantum Thief*

In this section, we construct a MLIP for zirconia, based on the SchNet [41, 391] MPNN architecture implemented in `schnetpack` [391]. We discuss the generation of training data, based on aiMD trajectories in the NpT ensemble, as well as the training scheme, and then investigate convergence with respect to two main parameters, the cutoff radius r_c and the number of interaction steps M, on the basis of auxiliary physical quantities and test set error. The section concludes with an investigation of the limits of the resulting potential and its suitability for the task at hand.

5.1.1 Training Data

We generate training data using aiMD for a set of trajectories in the NpT ensemble, heating a 96-atom supercell constructed from a tetragonal primitive cell to a set of target temperatures, 750 K, 1500 K, 2250 K, and 3000 K, covering the range of temperatures up to the melting point. A tetragonal primitive cell was chosen to match previous work by Carbogno et al. [143]; we will later find that the model generalises reasonably well to the monoclinic phase.

For this work, we did not implement an on-the-fly learning scheme [339], which can be used to refine the model during simulations, and can reduce the amount of training data, but inevitably couples the generation of data and model parameters. We aim to investigate the convergence with respect to model parameters, and therefore prioritise the simplicity of a fixed training set. In practice, we find this approach sufficient for the present application; see section 5.1.4.

Each trajectory was started from the same 96-atom simulation cell, velocities initialised at 10 K based on the Boltzmann distribution, and then ran for 2500 timesteps with Δt=4 fs. We use the Berendsen[1] barostat [86], with the pressure and temperature time constants τ_p and τ_T set to 10 ps, ensuring that the temperature range is traversed slowly, maximising coverage. This can be seen in figure 5.1.1.

Over these trajectories, 10 000 total single-point calculations were performed, using `FHI-aims` [356], with the PBEsol [70] functional, 2 ×

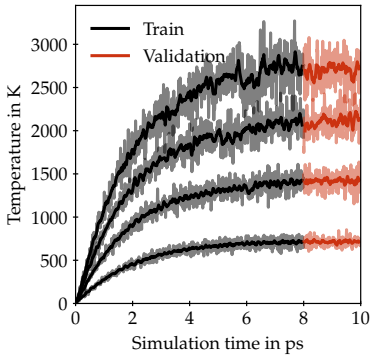


Figure 5.1.1: Temperature versus runtime for the four NpT trajectories used for training. Train and test sets are marked by colour; the solid lines indicate the rolling mean over 0.1 ps, shading the unfiltered data.

[1] In particular, its implementation in `ase`, which varies the length, but not the angle of lattice vectors. Figure D.1 shows the magnitude of the lattice vectors for the training and validation data.

2×2 k-points, and light basis sets, with an additional basis function for O, following the computational approach of reference [143]. The calculations were set up and run using `FHI-vibes` [89] and `ase` [88]. We use the first 2000 steps of each trajectory for training, and the remaining 500 steps for validation.

Before training, the mean of the potential energy over that entire dataset, −3 290 119.25 eV, is subtracted from all energies.[2] Predictions of the MPNN are scaled with a factor of 20 eV to ensure that they can cover the full range of training data, where the standard deviation of energies is 12.84 eV.

As test data,[3] we obtained the calculations by Carbogno et al. [418] from the NOMAD repository [419]. This data was not used during training.

[2] In principle, this offset can also be learned or added within the model. However, given the limited available precision, it is prudent to eliminate it from the beginning.

[3] The dataset consists of three trajectories of about 15 000 steps per temperature (300K to 2400K in intervals of 300K) with potential energy and force labels for each step. We use every tenth step of all trajectories to create the test set.

5.1.2 Training and Hyperparameters

As labels, we use the potential energy, the forces on each atom, and the stress in a joint squared error loss function, with weights 0.001, 0.999 and 100.0 respectively, and the ADAM optimiser [198]. Whenever a plateau of the loss on the validation set is encountered (with a patience of 20 epochs), the learning rate is reduced by a factor of 1/4 from a start of 10^{-4} to a minimum of 10^{-6}. If the loss has not improved for 200 epochs, the training is terminated. Training was performed on two Tesla Volta V100 32GB GPUs using a batch size of 100.

Change	R^2 Energy in %	R^2 Forces in %
No change	99.5	97.6
Patience halved	99.4	97.5
Learning rate decay 1/2	99.6	97.6
Patience reduced to 1/10	99.0	97.0
Batch size 25	99.8	97.8
Stress weight changed to 10	99.6	97.6
256 → 128 → 1 output network	99.6	97.7

Table 5.1.1: Impact of modifications to HPs on test set error

In this work, we consider the cutoff radius r_c and the interaction depth M as main HPs of the model and investigate convergence in detail. Other parameters were chosen empirically, based on the final validation loss after training, and not optimised explicitly. We find that the default settings of SchNet, 128 atom features, 128 filter width, and a two-layer (128 → 64 → 1) output network, provide satisfactory performance, with no significant improvement observed when increasing the size of the network. Training settings were chosen such that maximum training times do not exceed approximately seven days. Nevertheless, we probe the robustness of the chosen parameters in table 5.1.1, finding that the test set error does not strongly depend on variations in the choice of HPs.

5.1.3 Convergence with Cutoff Radius and Number of Interactions

We now focus on the cut-off radius r_c, which determines the amount of local information available, and the number of message passing steps M, which controls the range of interactions on the graph.

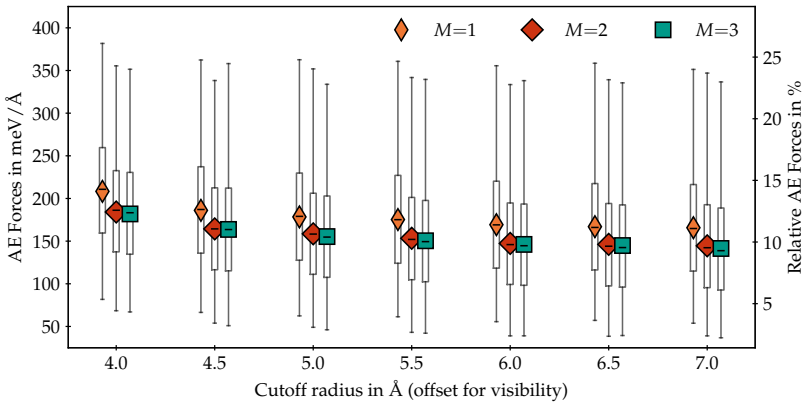


Figure 5.1.2: AE (see appendix A.3) for the forces on the test set, for models with different cutoff radii r_c and numbers of message passing steps M. Horizontal offsets have been added to distinguish M. Boxes, whiskers, bars, markers show interquartile range, total range, median, and mean, respectively. The relative AE is scaled by the standard deviation 1.37 eV/Å. Samples in the upper quartile are predominantly from trajectories above 2000 K, where limited training data is available.

First, test set errors were evaluated, shown in figure 5.1.2. Going from M=1, i.e., no message passing, to M=2 yields an approximately constant decrease in error across r_c. An additional message passing step yields only marginal improvements.

Predictive accuracy increases with cutoff radius, but saturates as the diameter of local environments exceeds the diameter of approximately 10 Å of the simulation cells used in training, approaching an all-to-all model. In this regime, all degrees of freedom are seen in the simulation cell, and message-passing cannot propagate additional information. However, M=1 is equivalent to a non-linear pair potential, while M>1 can model higher-order interactions, leading to lower errors. We therefore proceed with M=2 in the following, which is sufficient to demonstrate the effect of message passing for heat flux predictions and minimises additional computational cost.

Since predictive accuracy on a fixed test set cannot fully predict model performance for practical applications [126], where larger regions of the potential energy surface are explored, we evaluate *dynamical* properties as well. Figure 5.1.3 shows the vibrational density of states (VDOS) for different choices of r_c, evaluated for trajectories at 300K, started from identical initial configurations. For r_c=5 Å and higher, the reference VDOS from aiMD is adequately reproduced. Further increasing the cutoff only leads to marginal improvements or deviations at higher frequencies.

We therefore choose r_c=5 Å and M=2 as 'production' settings for the following.

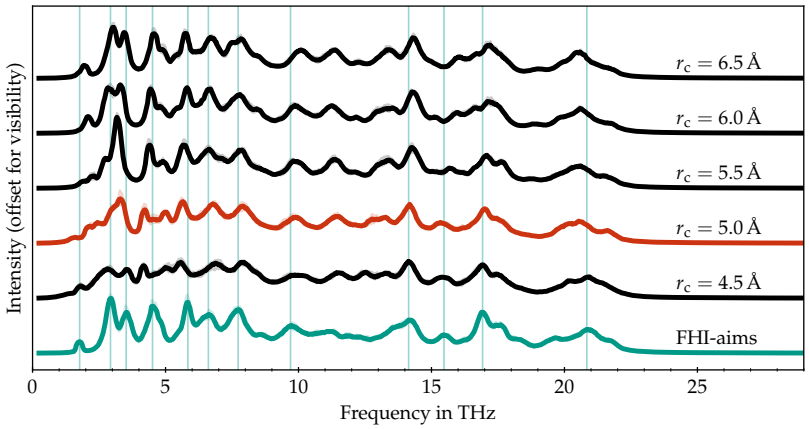


Figure 5.1.3: Comparison of VDOS for MPNNs (M=2) with different cutoff radii compared to a baseline computed with `FHI-aims`. The chosen production setting is highlighted in red. Vertical lines indicate peaks in the `FHI-aims` result. Constant vertical offsets are applied to distinguish curves. Results are averaged over three trajectories of 60ps (Δt=4 fs), in the tetragonal phase at 300 K, with matching initial configurations. Shaded areas indicate the minimum and maximum.

The corresponding figure with M=1 can be found in figure D.2.

5.1.4 Testing the Potential

Having chosen a set of HPs, we now further test the resulting MLIP. In order to assess its ability to model the dynamics of zirconia, we first consider the phonon band structure and density of states. The result is displayed in figure 5.1.4; the final model captures major features of the band structure, reproducing reference results adequately.

As a test of model stability and applicability, we investigate the temperature dependence of the unit cell volume, probing the monoclinic to tetragonal phase transition of zirconia, which occurs at around 1480 K and is accompanied by a discontinuous change in volume [420]. To this end, we perform an NpT simulation,[4] increasing the temperature with a heating rate of 1 K/ps. We compare with a similar simulation by Verdi et al. [163], which uses a heating rate of 0.5 K/ps. We note that this simple approach cannot be expected to produce a quantitative estimate of the transition temperature, which requires thermodynamic integration [163]. It can, however, indicate whether the transition is captured at all.

Figure 5.1.5 shows the result: Despite not being explicitly trained for this task, the model qualitatively captures the phase transition, at a similar temperature to another MLIP. However, it over-estimates unit cell volume by approximately 1 %, and becomes unstable above 2000 K.

Further tests reveal that, starting with approximately 1900 K, stable NVE simulations are no longer guaranteed: Out of eleven trajectories with 1 ns duration each, one encounters an instability. Attempting the same at 2000 K leads to failure in every trajectory.

Below 1800 K, we observe no instabilities in the potential in tetragonal and monoclinic cells, despite running hundreds of nanoseconds of MD at varying supercell sizes over the course of this work. In this temperature range, the MLIP is therefore considered stable.

To better understand the cause of the MLIP breakdown at elevated temperatures, we now investigate atomic displacements over time. The results for the monoclinic and tetragonal phases can be seen in figures D.5 and D.7. With increasing temperature, oxygen atoms

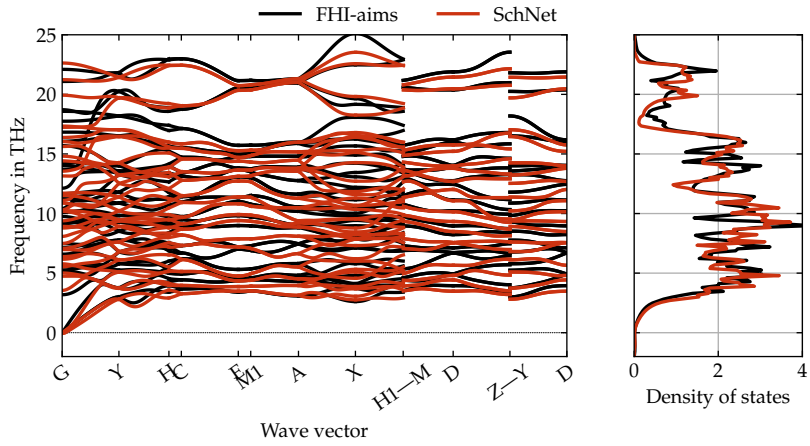
[4] We use the Martyna-Tobias-Klein barostat [93] with time constant τ=5 ps and stochastic velocity rescaling thermostats [85] with τ=5 ps for the positions and τ=3 ps for the lattice, with a simulation timestep of 1 fs, and a simulation cell of 324 atoms using `i-pi` [421].

become more mobile, displaying increasing displacements from their average positions. As the temperature of the monoclinic to tetragonal phase transition, 1400 K, is approached, different types of dynamical events are observed: Oxygen atoms become temporarily displaced into other local minima, or even participate in exchange-type oxygen diffusion. This latter effect becomes more pronounced with increasing temperature. While the MLIP remains stable for isolated events, increased diffusion above 1900 K leads to the breakdown of the potential, as shown, for instance, in figure D.6.

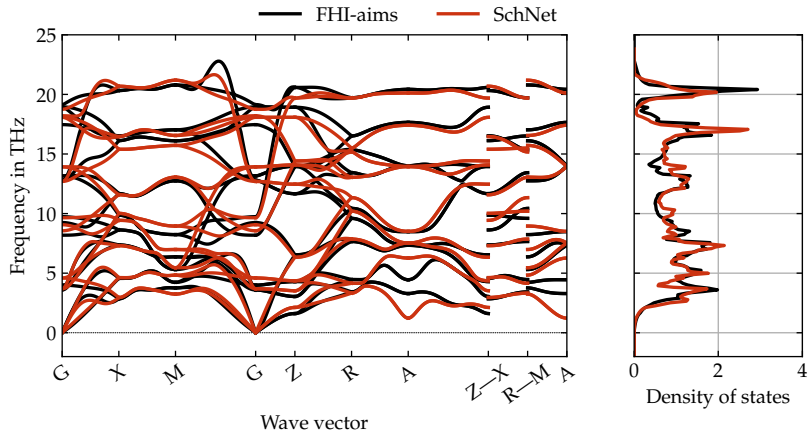
This might be due to the limited amount of training data for these processes, especially for the thermodynamic conditions close to the tetragonal to cubic phase transition.[5] Displacements for the training data are shown in figure D.8, revealing that this behaviour is also present in the training data, in line with recent literature [422], albeit in the form of slightly different diffusion events due to the smaller simulation cell and shorter trajectory lengths. However, we note that the portion of the data used for training contains limited samples for such events, as they occur towards the latter parts of the trajectories, which were reserved for validation.

[5] In figure D.1, magnitudes of lattice vectors in the training data are shown. Only two of the four total trajectories contain data in the cubic phase.

THESE OBSERVATIONS SUGGEST that our approach has yielded an MLIP capable of describing the dynamics of monoclinic and tetragonal zirconia up to temperatures of approximately 1800 K with sufficient accuracy for equilibrium MD and the GK method. At higher temperatures, an accurate description of defect formation is required, the development and validation of which is beyond the scope of the present work. For now, with its limitations in mind, we proceed with the SchNet MLIP to GK calculations.



(a) Monoclinic



(b) Tetragonal

Figure 5.1.4: Phonon band structure and density of states in the monoclinic (top) and tetragonal (bottom) phases, using the final SchNet MLIP with M=2 and r_c=5 Å, compared to a `FHI-aims` reference calculation. Results are shown for a 324-atom supercell. Convergence with respect to supercell size was checked.

Other values for M and r_c are shown in figures D.3 and D.4; higher r_c yields no significant improvements, while M=1 degrades accuracy.

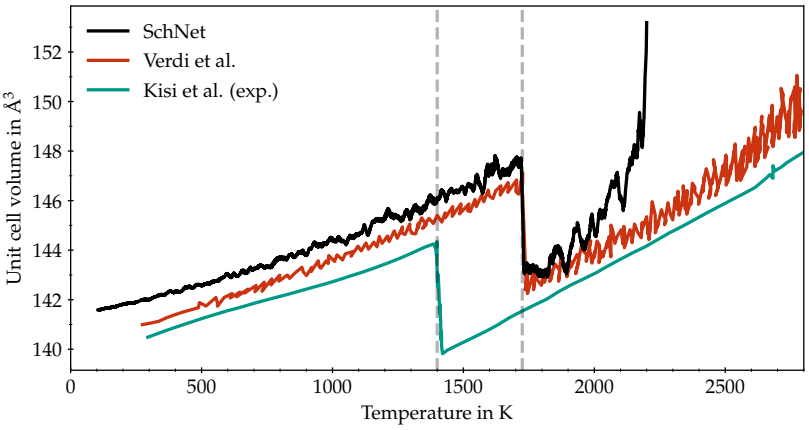


Figure 5.1.5: Unit cell volume, shown as rolling average over 10 ps, versus externally imposed temperature, compared to a similar MLIP simulation [163] and experimental reference values [420]. Vertical lines indicate the estimated transition temperatures.

Section 5.2
Green-Kubo Convergence and Workflow

Homo sapiens is about pattern recognition, he says.
Both a gift and a trap.

– William Gibson, *Pattern Recognition*

In section 2.5, a number of approximations required for a practical application of the GK method were discussed. Now, we investigate the impact of these approximations for zirconia and the SchNet MLIP, and determine parameters that yield converged results. The primary parameters determining convergence are the size of the simulation cell N and the simulation duration t_0. Additional parameters enter through the employed noise reduction scheme, the determination of the integration cutoff, and through computational efficiency considerations.

In principle, all such parameters can depend on *each other*; the level of convergence reached with regards to some parameters can influence the behaviour of others. Additionally, the convergence behaviour is dependent on the considered temperature and phase. In order to cope with this combinatorial problem, where an exhaustive search of all parameters is impractical, we first develop a general understanding of the behaviour of the system, fixing some parameters *a priori*. We then investigate the sensitivity of results with respect to these initial choices, and relax them where possible.

This section is organised accordingly: After initially describing the parameters and choices to be considered in section 5.2.1, we perform a number of overview experiments in section 5.2.2 to establish general trends. Then, in section 5.2.3, we consider noise reduction, investigating the impact of different parameters and choices. Next, computational parameters are investigated in section 5.2.4. Finally, in section 5.2.5 we study the convergence with respect to simulation size and duration. The final choices of parameters are summarised in section 5.2.6.

5.2.1 Workflow and Parameters

We recall the Green-Kubo relation from equation (2.5.8), and now recast it in the concrete form that is used in the present implementation, which is based on Knoop et al. [144]. For a simulation cell[1] of size N, we simulate n trajectories, indexed by s, with starting conditions

[1] The origin of the supercells used is discussed in appendix D.6; experimental lattice parameters are used, as opposed to determining them using the MLIP.

sampled from an NVT thermalisation trajectory, for a duration t_0. At every n_{hf}-th timestep, we compute the instantaneous heat flux[2] $\boldsymbol{J}_s(\tau)$, subtracting a non-diffusive gauge term.[3]

For the resulting set of time-series $\boldsymbol{J}_s(\tau)$, we first subtract the time average $\langle \boldsymbol{J}_s \rangle$, and then compute the set of HFACFs

$$\boldsymbol{C}_s(\tau) = \frac{1}{t_0 - \tau} \int_0^{t_0-\tau} \mathrm{d}\tau' \, \boldsymbol{J}_s(\tau + \tau') \otimes \boldsymbol{J}_s(\tau') \tag{5.2.1}$$

for each trajectory, which is subsequently averaged to yield $\boldsymbol{C}(\tau)$. The integral of $\boldsymbol{C}(\tau)$ yields the cumulative thermal conductivity

$$\boldsymbol{\kappa}(\tau) \coloneqq \int_0^{\tau} \mathrm{d}\tau' \, \boldsymbol{C}(\tau') \,, \tag{5.2.2}$$

which is subject to a lowpass filter, parametrised by a filter frequency ω_{filter}. From this filtered $\boldsymbol{\kappa}(\tau)$, the smoothed HFACF is obtained via finite differences to determine a cutoff time t_{c}, yielding the final thermal conductivity

$$\boldsymbol{\kappa} = \boldsymbol{\kappa}(t_{\mathrm{c}}) \,. \tag{5.2.3}$$

In this work, we consider the isotropic thermal conductivity[4]

$$\kappa = \mathrm{tr}(\boldsymbol{\kappa})/3 = \int_0^{t_{\mathrm{c}}} \mathrm{d}\tau \, \mathrm{tr}(\boldsymbol{C}(\tau))/3 = \int_0^{t_{\mathrm{c}}} \mathrm{d}\tau \, C(\tau) = \kappa(t_{\mathrm{c}}) \,, \tag{5.2.4}$$

The cutoff time is then taken to be the the first zero crossing of the HFACF, $C(t_{\mathrm{c}}) = 0$.

To summarise, we must consider the following parameters:

- (N, t_0), characterising size and duration of simulations,
- n, the number of independent trajectories,
- ω_{filter}, the filter width,
- n_{hf}, the spacing of heat flux computations.

We additionally briefly consider the relative importance and implementation of the two components of the noise reduction approach.

5.2.2 Exploration

We begin by investigating the dependence of $\kappa(\tau)$ on N and t_0, making the preliminary choices of n=11, n_{hf}=1, and ω_{filter}=1 THz. Considered choices for N and t_0, also shown in figure 5.2.5, are

- 'production' setting (1500, 1 ns), and its variant with reduced t_0,
- 'unconverged' setting (96, 0.1 ns), and its variant with increased t_0,
- 'light' setting (768, 0.5 ns) chosen as balance between convergence and computational cost,
- 'tight' setting (4116, 2 ns).

[2] As discussed previously, we use $\boldsymbol{J}_{\mathrm{int}}$ in place of the full $\boldsymbol{J}$ that includes a convective term. The impact of this choice is discussed in appendix D.5.

[3] The gauge term slightly differs from the one used by Knoop et al.For the present simulation durations and system sizes, we find no significant differences. Details are discussed in appendix D.4.

[4] In principle, convergence can be considered separately for the different components. However, no reference results are available for individual components, so this was not pursued.

Main convergence parameters N and t_0 are represented as (N, t_0) in this section.

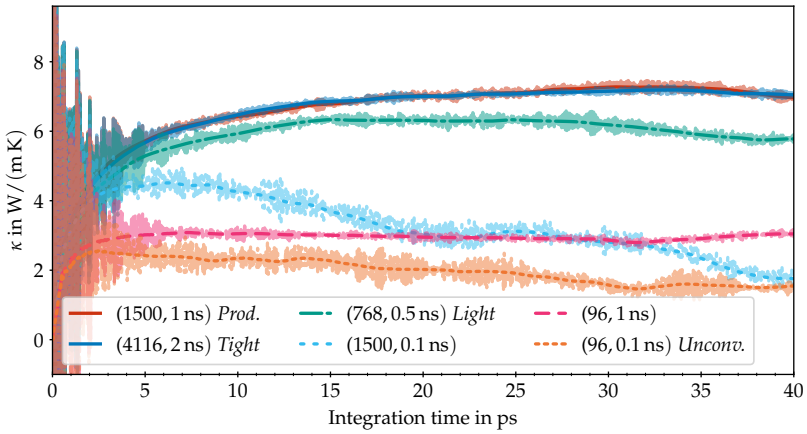


Figure 5.2.1: $\kappa(\tau)$ for monoclinic zirconia at 300 K for different N and t_0. Opaque lines indicate $\kappa(\tau)$ with applied noise reduction, translucent ones without.

This set of choices allows us to separate the convergence with respect to N and t_0, and investigate the suitability of the final settings for the remainder of this work.
At 300 K, figure 5.2.1, we find that lower N and t_0 yield correspondingly lower estimates for κ. At larger t_0, noise removal has less impact, as overall noise is reduced.

Both observations are consistent with expectations: Larger simulation cells can capture long-wavelength phonons and reduce unphysical boundary scattering. Longer simulation times allow spurious correlations to average out, and ensure that all relevant transport processes are captured. The same pattern holds at 1400 K, figure 5.2.2,

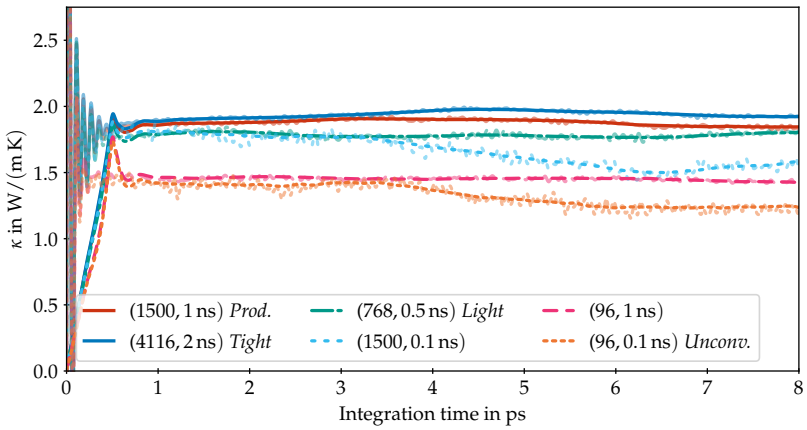


Figure 5.2.2: $\kappa(\tau)$ for tetragonal zirconia at 1400 K for different N and t_0. Opaque lines indicate $\kappa(\tau)$ with applied noise reduction, translucent ones without.

with significantly reduced overall noise. At this elevated temperature, the PES is less harmonic, and average phonon lifetimes are reduced, and consequently, $\kappa(\tau)$ converges faster with integration time. This also yields faster convergence with respect to N and t_0. The employed noise reduction approach is effective for both 'unconverged' and 'converged' settings, as well as temperatures and phases.

So far, we have considered $\kappa(\tau)$ averaged over eleven trajectories. Let us now broadly investigate the impact of this averaging. We

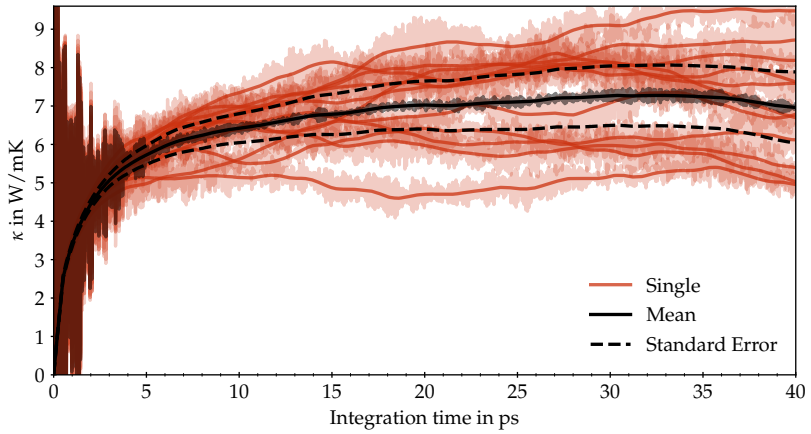


Figure 5.2.3: $\kappa(\tau)$ for monoclinic zirconia at 300 K, comparing ensemble average and single trajectories. Opaque lines indicate $\kappa(\tau)$ with applied noise reduction, translucent ones without.

consider (1500, 1 ns) in figure 5.2.3. The averaged $\kappa(\tau)$ exhibits significantly reduced noise compared to individual trajectories, both with and without explicit noise reduction. The standard error of the ensemble average exceeds the level of residual noise. This observation extends across temperatures and phases (not shown).

5.2.3 Noise Reduction

In section 5.2.2, we observed that, after sufficient integration times, and for sufficiently converged choices of t_0, averaged $\kappa(\tau)$ exhibits little residual noise. However, noise plays a significant role in the automated determination of the integration cutoff, which is based on the HFACF $C(\tau)$, rather than $\kappa(\tau)$ directly.

Noise reduction must therefore be considered. We follow the approach of Knoop et al. [144]. It consists of *(a)* the removal of a gauge term[5] from $\boldsymbol{J}$, and *(b)* filtering $\kappa(\tau)$ with a lowpass filter. The impact of both components is shown in figure 5.2.4. Without noise reduction, high-frequency fluctuations dominate the HFACF $C(\tau)$ and prevent an automatic determination of the integration cutoff time t_c. Removing a gauge term, reduces, but does not fully remove such noise. In contrast, the filtering approach is able to reduce most noise even without the removal of the gauge term.

However, the removal of the gauge term becomes relevant for the 'unconverged' setting, shown in figure D.9. In order to treat all choices of N and t_0 on an equal footing, we therefore employ both mechanisms jointly. At higher temperature, see figure D.10, overall noise is reduced, but similar observations can be made.

[5] See appendix D.4.

HAVING ESTABLISHED THE necessity of noise removal, we now study the impact of ω_{filter}. Knoop et al. propose to automatically determine ω_{filter} based on the lowest significant frequencies of the VDOS. However, this implies that ω_{filter} depends on the particular trajectory under consideration, and can vary across temperature. In this work, we intend to compare results across temperatures and trajectories, and

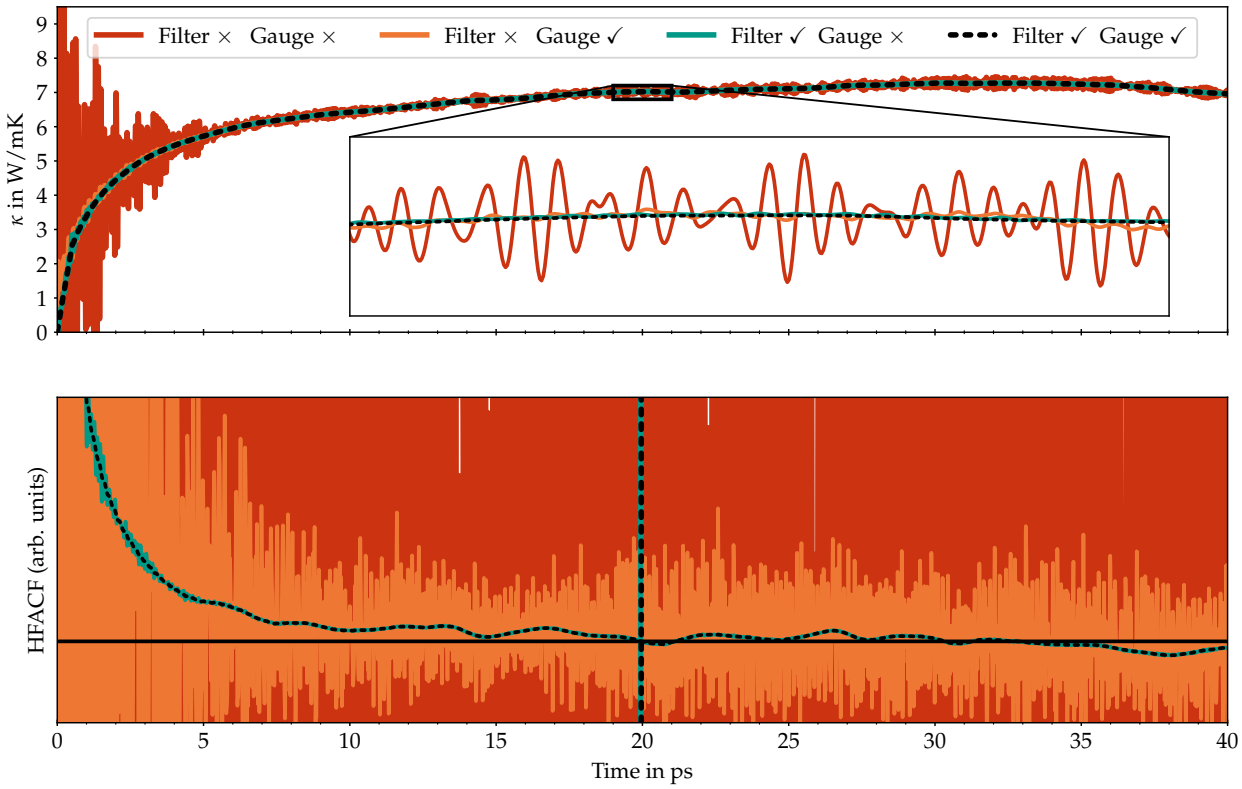


Figure 5.2.4: $\kappa(\tau)$ (top) and $C(\tau)$ (bottom) for monoclinic zirconia at 300 K for 'production' parameter choices (1500, 1 ns), comparing different components of the noise reduction approach. The checkmark ✓ indicates that the component is used, while × indicates that it is disabled. For filtering, ω_{filter}=1 THz was used. The vertical lines in the bottom plot indicate the cutoff time t_c. See figures D.9 and D.10 for other settings and temperatures.

therefore require a robust and consistent choice. We therefore choose a fixed ω_{filter}=1 THz, which approximately corresponds to the first peak in figure 5.1.3, and study its overall impact on $\kappa(\tau)$ and $C(\tau)$. The result can be seen in figures D.11 to D.14.

No strong dependence of the final thermal conductivity on the filter frequency is observed, and ω_{filter}=1 THz is found to be a robust choice across temperatures, phases, and levels of convergence.

5.2.4 Spacing and Number of Trajectories

Since the computation of the heat flux incurs additional computational cost, and since the processes relevant for heat transport occur on longer timescales than single simulation timesteps, the heat flux can be computed at a larger spacing n_{hf}>1 for efficiency.

Figures D.15 and D.16 show the dependence of $\kappa(\tau)$, and the value for κ determined by the first zero crossing of the HFACF, on this parameter. Across temperatures, spacings up to n_{hf}=3 yield identical results. We choose n_{hf}=2 to ensure consistency.

To some extent, the number of independent trajectories, n, is considered in the reported standard error. Nevertheless, an insufficient number of trajectories could lead to strong statistical fluctuations in reported results, which should be avoided. In figures D.17 and D.18, the dependence of κ on n is briefly considered; the chosen number of trajectories, n=11, is found to be more than sufficient.

5.2.5 Size and Time

Having determined noise reduction and computational settings, we can now finally investigate convergence with respect to N and t_0.

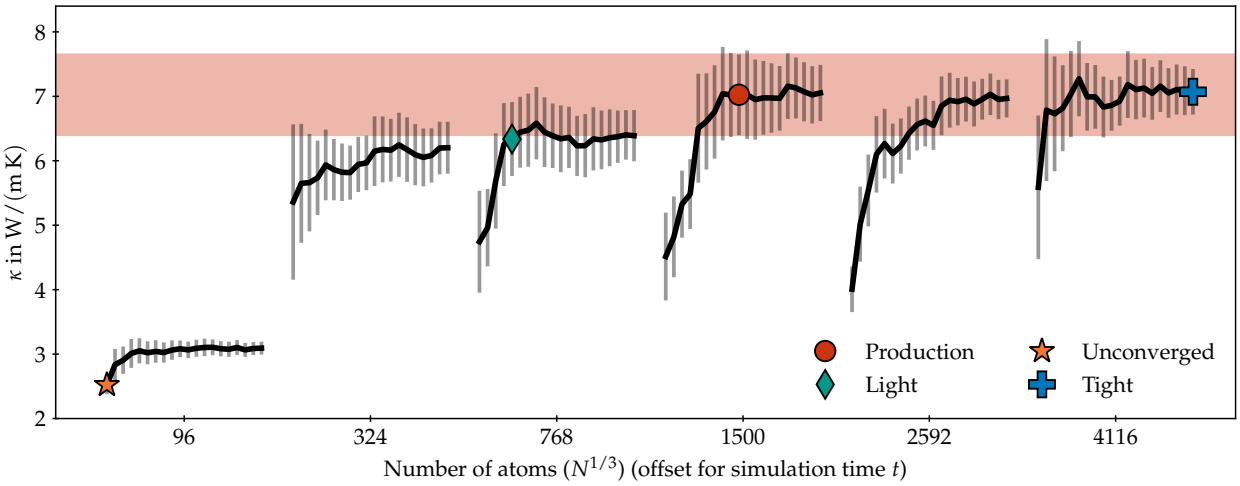


Figure 5.2.5: κ for monoclinic zirconia at 300 K for different choices of N and t_0. Error bars indicate the standard error across trajectories. N is shown as $N^{1/3}$, which is proportional to the length scale of the simulation cell. For each choice of N, t_0 from 0.1 ns to 2.0 ns are shown with a horizontal offset. 'Production', 'light', 'unconverged' and 'tight' choices are indicated; for the 'production' setting, the standard error is also shown as a shaded band.

See figures D.19 and D.20 for other temperatures and phases.

Figure 5.2.5 shows convergence with respect to simulation duration t and simulation cell size N at 300 K, figures D.19 and D.20 results for other temperatures. The chosen 'production' settings, (1500, 1 ns), are found to contain the results for higher N and t_0 within the associated standard error, with the exception of 1400 K in the monoclinic phase. In that case, unsystematic convergence behaviour at N>324 is observed, which may be due to the emerging instability of the monoclinic phase at that temperature. Nevertheless, differences between the chosen settings and κ at higher N and t_0 do not exceed 10 %.

We conclude that (1500, 1 ns) is sufficient for the computation of κ across the temperatures and phases relevant for the present work. Additionally, the 'light' settings (768, 0.5 ns) are found to provide an acceptable estimate of the converged κ at reduced computational cost.

5.2.6 Summary

For N and t_0, the 'production' settings (1500, 1 ns) are used for most results reported in section 5.3. In the case of computationally expensive heat flux variations, 'light' settings (768, 0.5 ns) are used instead. Additional choices are: ω_{filter}=1 THz for filtering, n_{hf}=2 as spacing for heat flux computation, and the usage of $\boldsymbol{J}_{\text{int}}$ in place of $\boldsymbol{J}$.

Section 5.3
Results for Zirconia

Having trained a GLP for zirconia, and having established a set of GK settings, we are now finally in a position to consider the thermal conductivity in zirconia across temperatures. First, however, we must verify heat flux formulation derived in section 4.2.

Consequently, this section is split into two parts. First, heat flux formulations are compared. Then, finally, the thermal conductivity is computed across temperatures and phases, and compared with other approaches.

5.3.1 Heat Flux

Figures 5.3.1 and 5.3.2 compares the linear-scaling 'unfolded' heat flux $\boldsymbol{J}^{\text{unfolded}}$ with the quadratically-scaling baseline $\boldsymbol{J}^{\text{mic}}$. Additionally, the purely local heat flux $\boldsymbol{J}^{\text{local}}$ is shown, which neglects all interactions beyond the neighbourhood cutoff radius r_c.

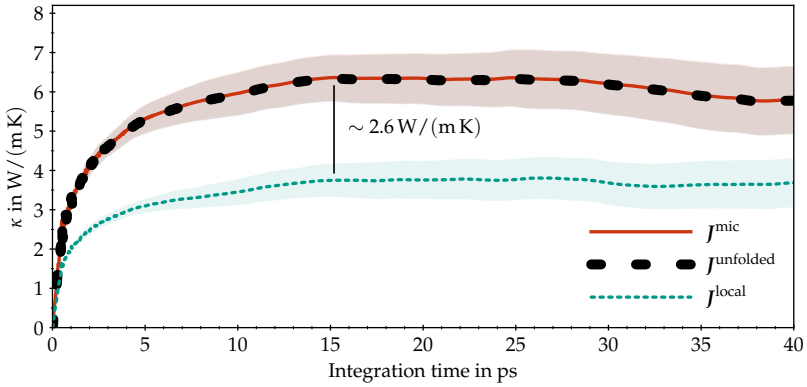


Figure 5.3.1: $\kappa(\tau)$ for selected heat flux formulations for monoclinic zirconia at 300 K, using a SchNet GLP ($M=2$, $r_c=5$ Å). The vertical line indicates the difference in determined κ for the respective heat flux formulations, based on the first zero crossing of the associated HFACF. Results are shown for 'light' convergence settings (768, 0.5 ns), with filtering with $\omega_{\text{filter}}=1$ THz, but no removal of a gauge term.

The results confirm that the efficient re-formulation of the heat flux, $\boldsymbol{J}^{\text{unfolded}}$, is equivalent to $\boldsymbol{J}^{\text{mic}}$, and therefore the full Hardy heat flux. The local flux $\boldsymbol{J}^{\text{local}}$ is not, underestimating the thermal conductivity by approximately 40 % due to missing interactions beyond $M=1$. These observations are found to be consistent across temperatures and phases, as seen in figure 5.3.2. A similar effect has been observed when formulations applicable to pairwise additive potentials are used for many-body FFs [402, 404].

In figure 5.3.3, a SchNet model with $M=1$ was used to recompute the heat flux for the trajectories used for figure 5.3.1. The result confirms

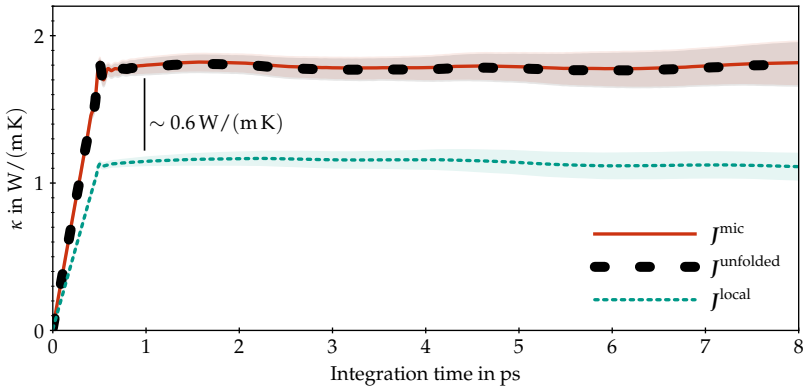


Figure 5.3.2: $\kappa(\tau)$ for selected heat flux formulations for tetragonal zirconia at 1400 K. See figure 5.3.1 for details.

that in the case of M=1, the local heat flux $\boldsymbol{J}^{\text{local}}$ is equivalent to $\boldsymbol{J}^{\text{mic}}$ and $\boldsymbol{J}^{\text{unfolded}}$. Figure D.25 further shows $\boldsymbol{J}^{\text{edges}}$ and $\boldsymbol{J}^{\text{semi-local}}$ which are equivalent as well. We can conclude that for M=1, all derived heat flux formulations are equivalent and can be used interchangeably; the most efficient choice, as seen in figure 4.2.2, is $\boldsymbol{J}^{\text{edges}}$.

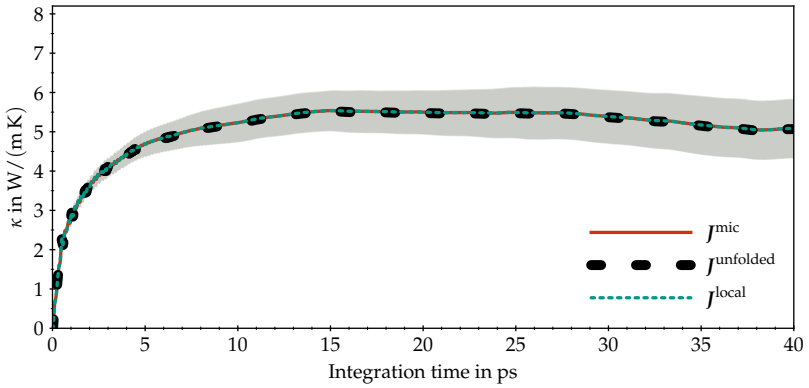


Figure 5.3.3: $\kappa(\tau)$ for selected heat flux formulations for monoclinic zirconia at 300 K, using a SchNet GLP (M=1, r_c=5 Å). For this figure, the M=1 model was used to recompute the heat flux for the trajectories used for figure 5.3.1. Other settings are identical to that figure.

IN SUMMARY, WE verified the theoretical considerations of section 4.2: For M>1, not all heat flux formulations are equivalent. The efficient reformulation $\boldsymbol{J}^{\text{unfolded}}$, however, is equivalent to the general heat flux $\boldsymbol{J}^{\text{mic}}$. In the case of M=1, all considered formulations are equivalent.

5.3.2 Thermal Conductivity

We can now finally compute the thermal conductivity at a larger scale, investigating multiple temperatures for zirconia.[1]

The result can be seen in figure 5.3.4. GLP predictions are in good agreement with both experimental measurements in the monoclinic phase, and theoretical MLIP predictions in the monoclinic and tetragonal phases. As this work uses similar lattice parameters and the same exchange-correlation functional as the work by Verdi et al. [163], the observed close agreement is to be expected. Remaining differences between the MLIP results may be due to larger simulation cells used in

[1] As discussed in section 5.1, the SchNet GLP used in this part of the thesis can be used for equilibrium MD simulations up to 1800 K, but not for an accurate determination of lattice parameters across the tetragonal to monoclinic phase transition. For this reason, and since the focus of the present work is the heat flux, we use experimentally determined lattice parameters.

Values originate from references [420, 423], as reported in reference [163]. Lattice parameters are given in appendix D.6. At 1400 K, we report results for both the tetragonal and monoclinic phases, as both lattice parameters are available and monoclinic simulations are sufficiently stable.

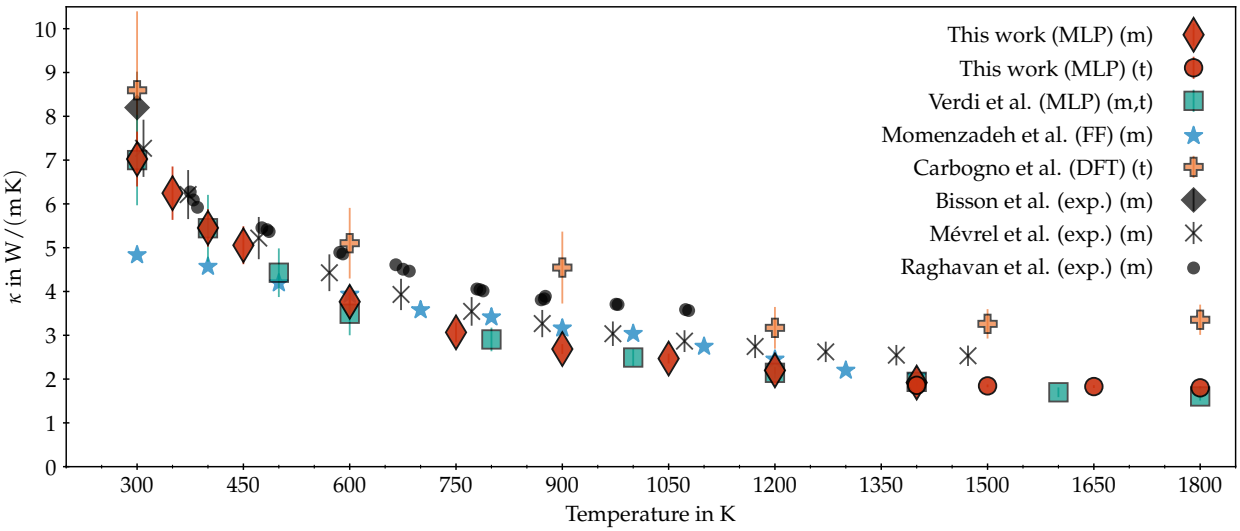


Figure 5.3.4: Thermal conductivity across temperatures computed with a SchNet GLP ($M=2$, $r_c=5$ Å) using experimentally determined lattice parameters [420, 423], compared with another MLIP [163], size-extrapolated *ab initio* GK [143], a FF [416] based on the Buckingham potential, and experimental measurements [424–426].
Error bars are shown as given in the respective publications, the present work provides the standard error across trajectories. 't' and 'm' indicate tetragonal and monoclinic phase.

the present work, enabled by the favourable scaling of computational cost due to the efficient heat flux implementation, and the semi-local nature of the employed GLP. Compared to experiment, both MLIPs are found to systematically underestimate κ by approximately 10 % to 20 %, which may be related to the intrinsic approximation of a finite-range MLIP, or the underlying density functional approximation.

Larger differences are observed with the size-extrapolated *ab initio* GK results reported by Carbogno et al. [143], which, however, were computed for the tetragonal phase at all temperatures. Additionally, due to the high computational cost of first-principles calculations, only three trajectories of 60 ps each were used, which is reflected in the larger statistical error.

The FF based on the Buckingham potential (see section 2.2.3) used by Momenzadeh et al. [416] underestimates thermal conductivity at 300 K by approximately 35 %, where MLIP results are in better agreement with experimental data. At higher temperatures, FF predictions are in line with experimental and MLIP results.

OVERALL, PREDICTIONS OF the present work are consistent with other MLIPs, as well as experimental measurements. We therefore conclude that, for this material, a GLP can be used to compute fully size- and time-converged thermal conductivity across temperatures and phases, provided the potential remains stable.

Section 5.4
Additional Materials

Sections 5.1 to 5.3 were dedicated to validating the heat flux formulations developed in section 4.2 using the SchNet MPNN and zirconia. We can now turn our attention to a more recently developed GLP, the equivariant transformer So3krates [43],[1] and other materials, to further explore the applicability of GLPs to the task of computing thermal conductivities with the GK method.

In particular, we investigate the regime of high anharmonicity and consequently low thermal conductivity with SnSe, and the regime of low anharmonicity and high thermal conductivity with Si. The latter case presents a challenge for the GK method, as low anharmonicity implies long phonon lifetimes and mean free paths, and hence large simulation duration and simulation cell size [428, 429]. It can therefore serve to probe the limits of the implementation of the GK method developed for this thesis.

Additionally, alternative approaches for obtaining training data are explored. For SnSe (section 5.4.1), we make use of a dataset created in a recent work by Knoop et al. [411], who performed a large-scale study of strongly anharmonic materials with the aiGK approach. They computed κ at 300 K for 24 materials with experimentally measured thermal conductivities, as well as 34 additional materials where no such measurements are available. For each material, the dataset[2] contains the trajectories used to computed thermal conductivities, i.e., differing numbers of NVE simulations with up to 60 ps duration each, as well as short NVT trajectories at different volumes with $\approx$ 3000 total calculations used for thermalisation and determination of lattice constants. We use *only* the latter for training and validation,[3] and then perform GK calculations with the resulting GLPs. For Si (section 5.4.2), training samples are created with a stochastic phase-space sampling scheme that does not require aiMD at all.

[1] Implemented using `jax` in `mlff` [427].

[2] Available on the NOMAD repository [430].

[3] Additionally, one of the NVE trajectories is used for comparing VDOS.

5.4.1 Tin Selenide

The first investigated material is tin selenide (SnSe), which features low thermal conductivity and high anharmonicity (σ^{A}=0.35 at 300 K [142]). In addition to the work by Knoop et al., this material has been previously investigated theoretically with a force constant potential (FCP) by Brorsson et al. [431], a MLIP using MTPs by Liu et al. [375], as well as in a number of experiments surveyed by Wei et al. [432].

Training The model was trained on aiMD NVT thermalisation trajectories at 300 K, performed by Knoop et al. [411]. All trajectories were combined, yielding an overall dataset of 3489 structures at four different volumes. From that dataset, 2400 random samples were used for training, 600 for validation, and the remainder for testing.

Model training for SnSe was carried out by Florian Knoop. All presented results were computed by me.

The number of interaction steps M is varied from M=1 up to M=3. We fix the cutoff radius to r_c=5 Å, the embedding dimension to F=132 and the maximal degree in the equivariant branch to l_{max}=3. Non-local corrections in So3krates are not used, as the GLP framework as defined in section 4.1 does not describe non-local models, and consequently, no heat flux is available.

The model is trained by minimising a joint loss of the potential energy and forces with loss weightings of 0.01 and 0.99 respectively, using the ADAM optimiser [198]. The training is stopped after 2500 epochs, with a batch size of 10. After each epoch, the performance of the current model is evaluated on the validation data and the best-performing model is saved for production. The initial learning is set to 10^{-3} and is reduced every 100k steps using exponential learning rate decay with a decay factor of 0.7. No early stopping is employed. Training times on a single Nvidia A100 40 GB GPU range from 2h54min for M=1 up to 6h46min for M=3.

Testing First, errors on the test set were evaluated. They can be found in tables D.4 to D.6. For all M, errors are lower than reported by Liu et al. [375]. While going from M=1 to M=2 yields a large improvement in accuracy, M=3 only yields minor improvements.

To further test the model, the VDOS and phonon band structure are compared with results obtained with `FHI-aims`. The results can be seen in figures D.26 and D.27: For M=2, 3 So3krates is in excellent agreement with results obtained with DFT.

Thermal Conductivity Finally, we can proceed to GK calculations. Following the overall workflow developed for the previous sections, approximately cubic simulation cells at different sizes were constructed from the primitive cell at 300 K[4] obtained by Knoop et al., and thermalised in the NVT ensemble for 0.2 ns. From the resulting trajectory, starting configurations for 11 trajectories were extracted. Finding that (864, 2 ns) yield converged results (see figure D.28), we ran 11 MD simulations using the `glp` package, computing $\boldsymbol{J}$ at every step (Δt=4 fs).

[4] Lattice constants 11.634 Å, 4.196 Å, and 4.404 Å.

Table 5.4.1 shows the result: For M=2, 3, the GLP is in excellent agreement with the work by Brorsson et al. [431], which uses a FCP. A larger difference is observed with the work by Liu et al. [375], who, however, use the PBE exchange-correlation functional, as opposed to PBEsol, which was used for the present work. The observed thermal conductivity is consistent with experiments,[5] as well as the size-extrapolated DFT result of Knoop et al. [411]. The anisotropy of $\boldsymbol{\kappa}$ in SnSe is captured as well.

[5] The comparatively large variance in quoted experimental values is due, in part, to the difficulty of obtaining pure single crystals of SnSe. For instance, Zhao et al. [433] reported an ultralow thermal conductivity, but later criticism by Wei et al. [432] states that these results are not intrinsic to pure SnSe, as sample density does not match the expected value.

Source	Method	κ in W/(m K)	κ_x	κ_y	κ_z
This work	So3krates, M=1	0.99 ± 0.10	0.53 ± 0.03	1.31 ± 0.13	1.12 ± 0.12
"	So3krates, M=2	1.13 ± 0.07	0.48 ± 0.04	1.59 ± 0.07	1.20 ± 0.07
"	So3krates, M=3	1.13 ± 0.10	0.56 ± 0.05	1.56 ± 0.15	1.32 ± 0.16
Brorsson et al. [431]	FCP	1.12	0.57	1.46	1.32
Liu et al. [375]	MLIP	0.86 ± 0.13	0.57 ± 0.05	1.25 ± 0.24	0.76 ± 0.08
Knoop et al. [411]	DFT (extrapolated)	1.40 ± 0.39	–	–	–
Review by Wei et al. [432]	Experiments	0.45 to 1.9	–	–	–

Table 5.4.1: Thermal conductivity of SnSe at 300 K. The right-hand side of the table shows the components of $\boldsymbol{\kappa}$.

5.4.2 Silicon

After investigating a material with low thermal conductivity, we now discuss Si, a material with high thermal conductivity and low anharmonicity (σ^{A} = 0.18 at 400 K). Thermal transport in Si has been studied extensively in the past, both theoretically with first-principles methods [143, 210, 434], FFs [143, 371, 406, 429, 435] and MLIPs [376, 378], as well as experimentally [436–442]. And overview of reported values for κ is given in table 5.4.2.

Convergence Convergence issues for the GK method FFs and aiGK due to low anharmonicity have been discussed previously.

He et al. [429] review prior work with FFs and perform a convergence study at 300 K with the Tersoff FF, finding that at least N=64000 atoms and on the order of t_0=10 ns of simulation time per trajectory are required. Also working with the Tersoff FF, Dong et al. [435] report convergence with N=1728 and t_0=10 ns at 300 K. Fan et al. [371] observe no significant size dependence at 500 K with a total simulation time[6] T_0=200 ns. Carbogno et al. [143] report convergence with approximately 12 ns and N=1728 at 300 K and N=4096 at 1000 K.

[6] $T_0 := nt_0$.

For the SW FF, Howell [406] find that N=1728 and $T_0 \geq$80 ns at 500 K and T_0=60 ns at 1000 K are sufficient. Also for the SW FF, Schelling et al. [443] determine N=1728 and T_0=3 ns to 6 ns as converged.

In the case of aiGK, Carbogno et al. [143] develop an extrapolation scheme to correct finite-size effects, using N=96 atoms and a total simulation duration of T_0=0.2 ns in the aiMD trajectories. The extrapolation technique is stated to be responsible for up to 50 % of thermal conductivity at low temperatures with these settings.

For MLIPs, Qian et al. [378] use a Gaussian approximation potential (GAP) trained with data obtained via stochastic sampling of normal modes, and compute thermal conductivity both with a BTE approach and the GK method. No information on convergence is given. Another MLIP work by Li et al. [376] uses a BTE approach [444] for temperatures below 1000 K, and the GK method at 1200 K with N=20000; simulation duration is not given.

IN SUMMARY, NO CLEAR consensus on convergence for FFs and MLIPs can be identified, beyond requiring at least N=1728 atoms and ap-

proximately 10 ns of simulation time per trajectory. Anticipating challenging convergence behaviour, and aiming to compare with results by Carbogno et al. [143], we therefore choose 400 K, rather than room temperature, as target temperature.

Training Training data for Si was generated with a stochastic phase-space sampling scheme described in reference [445], generating independent samples at different temperatures based on harmonic normal modes. For training data, samples were generated at 200 K to 800 K in steps of 200 K, stretching a 216-atom supercell of the 0 K primitive cell with 7 different uniform strains from 0.95 to 1.25. For each of the 28 possible combinations, 64 samples were generated, yielding a total of 1792 geometries, which were computed with `FHI-aims`, the light basis sets, the PBEsol [70] exchange-correlation functional, and a $2\times2\times2$ k-grid. As a test set, the same procedure was repeated for 300 K to 900 K with 5 strains from 0.9 to 1.3 and 32 samples each, generating 320 test structures. In total, approximately 2200 first-principles calculations were performed. A similar scheme was used by Qian et al. [378].

Training proceeded as described previously for SnSe, using 1200 structures for training, and 300 for validation. The maximum number of epochs was set to 3000. Training on an Nvidia V100 32 GB GPU took 2h52min for M=1, 5h28min for M=2, and 8h15min for M=3.

Testing Errors on the test set were evaluated first and are given in tables D.7 to D.9. For M=2, 3, errors are comparable to GAP as reported in reference [378], where the test set is sampled at the same temperatures as training, and only two temperatures are used.

Next, the phonon band structure, displayed in figure D.29, is computed for all three models. For M=2, 3, good agreement with `FHI-aims` is found. Anticipating high computational cost due to convergence requirements, we therefore continue with M=2.

Thermal Conductivity With the final 'production' M=2 model, GK calculations at 400 K were performed, using the 0 K lattice obtained with `FHI-aims` for simplicity.[7] From an 8-atom cubic primitive cell, supercells with 512, 1728, and 4096 were constructed and thermalised with the Langevin thermostat for 0.4 ns, extracting 11 starting configurations for GK runs. For each, NVE MD with 20 ns duration and a timestep of Δt=2 fs was run, computing the heat flux every 20 fs.[8]

Figure 5.4.1 shows convergence behaviour in both simulation cell size N and simulation duration t_0. For sufficiently high values of t_0, a systematic increase of κ with simulation cell size is observed, which has not yet terminated at the largest N investigated. With the current implementation of the heat flux in `glp`, and 32 GB of GPU memory, larger simulation cells could not be studied.

Therefore, no converged value for κ at 400 K can be reported here. The value obtained for N=4096 and t_0=20 ns, (56 ± 4) W/(m K), underestimates experimental values, as shown in table 5.4.2, by 45 % and the extrapolated aiGK result by Carbogno et al. [143] by 55 %. An-

[7] The lattice constant is 5.444 Å. Dependence of results on lattice parameters was checked; no strong impact was observed.

[8] With these settings, using `glp` and an Nvidia V100 32 GB GPU, one trajectory at N=4096 requires approximately 200 GPU hours. The calculation of the heat flux, responsible for half the time, can be parallelised across timesteps for efficiency. N=1728 and N=512 require 100 and 50 GPU hours respectively.

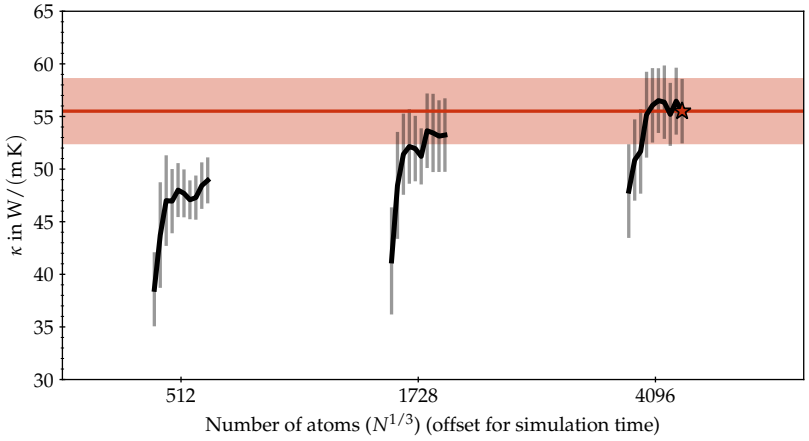


Figure 5.4.1: κ for Si at 400 K for different choices of N and t_0. Error bars indicate the standard error across trajectories. N is shown as $N^{1/3}$, which is proportional to the length scale of the simulation cell. For each choice of N, t_0 from 1 ns to 20 ns are shown with a horizontal offset. Largest simulation is indicated, the associated standard error is also shown as a shaded band.

other MLIP using the GK method, reported by Qian et al. [378], also underestimates κ, albeit by only 24 %.

References	PES/Sample	Method	κ at 300 K	κ at 400 K	κ at 500 K
[436, 437, 439, 441, 442]	Natural Si	Experiment	148	101	75
[438, 441, 442]	^{28}Si	Experiment	153	94	74
Carbogno et al. [143]	DFT (PBEsol)	GK (extrapolated)	–	127 ± 29	–
Carbogno et al. [143]	DFT (LDA)	GK (extrapolated)	–	105 ± 25	–
Plata et al. [434]	DFT (PBE)	BTE	144	–	–
Broido et al. [210]	DFT	BTE	156	–	–
He et al. [429]	Tersoff FF	GK	197 ± 34	–	–
Howell [406]	SW FF	GK	–	–	206 ± 8
Qian et al. [378]	GAP MLIP	GK	124 ± 22	77 ± 19	61 ± 17
Qian et al. [378]	GAP MLIP	BTE	137	99	77
Li et al. [376]	DeepPot-SE MLIP	BTE	140	100	64
This work	So3krates MLIP	GK (unconverged)	–	56 ± 4	–

Table 5.4.2: Overview of reported thermal conductivity for Si. Experimental values have been averaged over references. Only selected, representative, FF results are shown. Error bars for reference [143] have been estimated from the nearest data point.

Setting aside the issue of convergence, our model appears to underestimate thermal conductivity obtained via the GK method in this case. Comparing BTE and GK approaches with the GAP MLIP, Qian et al. observe a difference of around 22 % at 400 K, with the GK method underestimating κ as well. As a similar sampling scheme is used in their work, this may indicate a systematic bias due to not relying on aiMD to generate samples. As all training and test data, as well as the phonon band structure used for model validation, rely on the harmonic approximation, biased representation of anharmonicity may have gone unnoticed. A comparison of training methods, and calculation of results with BTE methods, or the extrapolation scheme by Carbogno et al. [143] and Knoop et al. [411], is left for future work.

Section 5.5
Summary

This chapter discussed the application of the methods developed in chapter 4 in practice, studying thermal transport in selected materials with two different GLPs.

Sections 5.1 to 5.3 discussed the prediction of thermal conductivity of zirconia (ZrO_2) across temperatures using the SchNet [41, 391] GLP. Using a simple training scheme based on NpT aiMD simulations was found to yield an MLIP that remains stable up to 1800 K, but becomes unstable with the onset of oxygen diffusion at elevated temperatures. Nevertheless, the potential was found to predict thermal conductivities at lower temperatures in excellent agreement with another MLIP [163], and in good agreement with experimental measurements [424–426]. We also verified that the heat flux formulations in section 4.2 are equivalent to the Hardy form, and that neglecting semi-local interactions leads to an underestimation of κ by 40 %.

To study applicability in different settings, we then investigated both a material with high anharmonicity, SnSe at 300 K, and a low-anharmonicity material, Si at 400 K, with the So3krates [43] GLP. Good agreement with other GK approaches [375, 411, 431], as well as experiment [432], was achieved for SnSe, using So3krates GLPs trained on thermalisation trajectories of a previous aiGK study [411]. For Si, using a So3krates model trained on samples generated via normal mode sampling, size convergence could not be achieved with the present implementation, and a likely underestimation of κ was observed.

Limitations

We identify a number of limits and challenges for GLP-driven GK simulations in practice.

First of all, as seen for Si, and to a lesser extent ZrO_2 at low temperatures, materials with low anharmonicity, which require large simulation cells and durations, pose a computational challenge for MLIPs, where typically available implementations are not yet as scalable and efficient as FFs. In such settings, the use of extrapolation methods for GK, or thermal transport approaches based on the BTE, which are suitable for harmonic systems, may be preferable.

Beyond convergence, which can in principle be tackled via an optimised implementation, we have also encountered a core challenge with MLIPs: the procurement of suitable training data. In this thesis, simple training schemes based on aiMD (ZrO_2, SnSe) and stochastic

sampling (Si) have been employed, which do not consider model uncertainty and do not feature active learning. While good accuracy has been achieved in the investigated cases, shortcomings of such methods have become apparent: Stability of MD simulations at higher temperatures is compromised in ZrO_2, where defect formation is not fully modeled, and potential bias has been identified in Si.

This challenge is related to the question of model reliability. In a practical setting, a measure of uncertainty in predictions is required to avoid unphysical predictions, which lead to instabilities, or a biased sampling of the PES. Such an uncertainty metric can then also be employed to select additional training data. A brief discussion of such methods can be found in section 2.4.2.

Chapter 6
Conclusion

> Science is failing and taking notes.
>
> –Cory Doctorow, *Walkaway*

In this final chapter, a summary of the thesis is presented, followed by an outlook of extensions of the presented work. To conclude, a general perspective on potential future developments is given.

Summary

Chapter 3 focused on the problem of representing atomistic systems for ML. An overview of common techniques for constructing such representations, a classification in terms of approaches to symmetry, and a literature review of methods, was provided. Selected representations (SFs, MBTR, SOAP) were then compared empirically on benchmark datasets related to computational screening, controlling for other factors such as regression method, HP optimisation, and data distribution. The results showed that increased interaction order leads to higher predictive accuracy across datasets, but also leads to higher computational cost.

Chapter 4 discussed the application of semi-local MLIPs to the simulation of thermal transport in solids. While such MLIPs are constructed from local atomic environments, they admit interactions beyond them through iterative message passing schemes, increasing their receptive field while maintaining the efficiency of fully local models. A unified framework for describing such models, which we termed GLPs, was developed and first applied to the calculation of forces and stress, resolving previous ambiguities related to periodic boundary conditions, and providing a straightforward way to implement these quantities with AD.

Building on this framework, we then considered the definition of the heat flux, the central quantity required for the calculation of thermal conductivities with the GK method. Finding that previous formulations of the heat flux cannot be efficiently applied to GLPs, or are defined ambiguously for the periodic case, we re-derived the Hardy formulation of the heat flux from the continuity equation, obtaining a form that explicitly considers periodicity. In order to apply the result

in practice, reformulations of the heat flux suitable to efficient implementation with AD were discussed, and a linear-scaling 'unfolded' form that applies to semi-local potentials was developed.

Chapter 5 verified and applied these theoretical contributions in practice. To this end, we investigated the thermal conductivity of zirconia with a GLP based on the SchNet [41, 42] MPNN architecture. We found that a simple training scheme based on NpT MD simulations yields an MLIP describing the dynamics of zirconia up to 1800 K, but leads to instabilities once non-trivial dynamics related to oxygen diffusion occurs. Nevertheless, the MLIP was found to be sufficient to predict thermal conductivities with the GK method, with good agreement with experimental measurements and other MLIP results.

Having verified our approach to the GK method, we then applied a more recently developed equivariant GLP, So3krates [43], to the prediction of the thermal conductivity of SnSe and Si. For SnSe, which is highly anharmonic and exhibits consequently low thermal conductivity, So3krates results are in excellent agreement with other MLIPs and extrapolated DFT results, as well as experimental values. In the case of high thermal conductivity and low anharmonicity, Si, size convergence proved to be challenging, and thermal conductivity was underestimated by approximately 45 %. With these experiments, the construction of suitable training datasets, model stability in unseen situations, and convergence in settings of high thermal conductivity, were identified as key challenges for future applications of GLPs to thermal transport simulations.

Outlook

We discuss open questions and extensions of the presented work.

Doped and High-Temperature Zirconia In sections 5.1 to 5.3, pure zirconia in the monoclinic and tetragonal phases up to temperatures of 1800 K was investigated. While this system has served as a good benchmark case for the developed methods, it is of limited practical interest: Industrial applications of zirconia typically focus on elevated temperatures and the tetragonal and cubic phases, stabilised through doping [203, 415].

Investigating the full space of dopants, dopant concentrations, and temperatures of zirconia, in particular with the addition of active learning, would be a promising direction for future inquiry. In this context, the role of oxygen diffusion and spontaneous defect formation [422], observed in passing in section 5.1, could be investigated further. While the typical dopant concentrations for thermal barrier coatings based on zirconia, on the order of 9 mol%, are accessible with DFT, the general case of lower dopant concentrations requires additional consideration, discussed below. We also note that charged oxygen defects, which emerge, for instance, from doping with yttria (Y_2O_3) [446–448], may serve as a challenging test case for MLIPs, in particular those which aim to model electrostatics and charge transfer. GLPs, which

aim to implicitly model such behaviour, could also be benchmarked further.

Active Learning and Reliability The instability encountered at elevated temperatures in section 5.1, and potential biased sampling discussed in section 5.4.2, highlight the difficulty of using MLIPs for practical investigations: MLIPs are inherently limited by the available training data, and hence cannot be expected to accurately model unseen behaviour. Yet, such behaviour is often the object of scientific inquiry, and the training data required for good performance is therefore not always known in advance. While this problem is intractable in principle, practical approaches based on uncertainty quantification and active learning exist and are a rapidly evolving field of research (see section 2.4.2). However, closely related to the considerations in the previous paragraph, workflows based on active learning and on-the-fly refinement of a MLIP are typically available only for specific models, MD simulation environments, or electronic structure codes.[1] This complicates the evaluation of novel MLIPs; more modular and generic approaches are required. Their development is another promising route for subsequent work.

[1] For instance, the Vienna Ab initio Simulation Package (VASP) [287, 449], used for zirconia by Verdi et al. [163].

Large-Scale Validation of GLPs for Thermal Transport This thesis presented results for three out of approximately 100 materials previously investigated with the aiGK approach. While the results are encouraging, further investigation of more materials is needed to determine whether this approach scales across the full dataset. Additionally, the generalisation of MLIPs across temperatures could also be tested. Any failure cases encountered in this large-scale validation could be used to guide the development of more advanced approaches.

Heat Flux for Non-Local Models While the general heat flux in equation (4.2.15) is applicable to any potential decomposed into atomic energy contributions, the linear-scaling formulations ready for implementation with AD are restricted to potentials with a finite interaction cutoff radius r_c^{eff}. This represents a challenge for the recently developed models, discussed briefly below, that include long-range and non-local interactions. Furthermore, this restriction even excludes simple pairwise electrostatics or dispersion corrections. In the latter case, assuming that charges and other parameters are held fixed, the evaluation of the heat flux can in principle proceed through Ewald summation [409]. If charges are modeled with semi-local or local models, the approach of section 4.2 may still be applied to evaluate inner derivatives. In the case of charge equilibration, AD-based implicit differentiation [450] could be used to obtain inner derivatives. The development of an efficient heat flux formulation for such cases represents another direction for future work.

Explicit Jacobians for GLPs The 'unfolded' heat flux from section 4.2 aimed to avoid explicitly computing the Jacobian $\partial U_i / \partial \boldsymbol{r}_j$, as its dir-

ect evaluation scales[2] as $O(N^2)$. However, this computational scaling represents the fully general case, where no internal structure of the target Jacobian is known. In GLPs its sparsity pattern is known ahead of time, as it is determined by the effective interaction cutoff radius r_c^{eff}; only a linear-scaling number of entries are nonzero. Therefore, techniques for the calculation of sparse Jacobians [39, ch. 7] are readily applied to this case. AD systems commonly used for MLIPs do not yet generally support such techniques, but a sparse Jacobian functionality for `jax` [37] is being developed[3] at the time of writing. The ability to compute full Jacobians, and potentially higher-order derivatives [200], may be useful in the context of calculating force constants, or for higher-order optimisation techniques. It would also provide an alternative to the 'unfolded' heat flux, avoiding the modification of the computation of the potential energy.

Dynamics Benchmarks and Infrastructure While energy or force errors are often reported for newly proposed MLIPs, it has been shown that such errors can only provide a rough estimate of model performance in practice [126]; this has also been observed in section 5.1 of this thesis. While more comprehensive benchmarks based on MD simulations are available [126, 341], they are not yet widely adopted in the MLIP community.[4] In addition to other factors, such as the expertise required to run MD simulations, we would like to suggest that this situation is also a consequence of a lack of appropriate tools: While interfaces between MD packages and MLIPs exist,[5] they are typically tightly coupled to a particular MLIP.

As we have argued in section 4.1, this tight integration is not always required: Any potential that follows the overall architecture of a GLP can be treated on equal footing when computing forces and stress, the core quantities required for MD simulations.[6] The `glp` [410] framework developed for this thesis aims to contribute to making such application-oriented benchmarks more accessible for ML researchers.

Importance and Meaning of Semi-Local Interactions In this thesis, much work has been devoted to implementing the heat flux for semi-local GLPs, enabling GK calculations with this class of models. However, the importance of such interactions in practice remains an open question. In this work, we have observed lower errors for models with $M>1$, for instance in figure 5.1.2 and tables D.4, D.5, D.7 and D.8, as well as improvements in VDOS and phonon band structures (figures D.26, D.27 and D.29) and predictions of κ (table 5.4.1), but improvements beyond $M=2$ are often marginal.

The relative importance of the larger receptive field afforded by semi-local interactions, as opposed to the construction of interactions with increasingly high body-order through message-passing iterations, also remains uncertain at present. Recent work aimed at decoupling the two mechanisms [335] indicates that the impact of body-order dominates. However, at the time of writing, no such results are available for solids. An additional complication arises through the restricted size

[2] The overhead of computing the Jacobian in the worst case is $\min(N, 3N)=N$ as the input dimension is $3N$ and the output dimension is N; see [39, ch. 7]. We assume linear scaling of the computation of the set of U_i.

[3] `https://github.com/mfschubert/sparsejac`.

[4] A recent review by Isayev and Anstine provides a concise summary [451]:

> ... the number of different [MLIP] models greatly exceeds the cases in which they have been uniquely successful for understanding a chemical or materials science challenge. Most [MLIPs] have only been tested on a handful of systems in simple trial studies, and as a consequence, the area of applied [MLIP] modeling lags behind model development.

[5] The `nequip` code [393], for instance, features LAMMPS integration.

[6] Indeed, from the perspective of MD, any potential can be treated on an equal footing, since only derivatives of U are required, which can in principle be computed with AD. However, in practice, management of implementation details such as neighbourlists or the particular implementation of the MIC is required, and is conveniently handled at the abstraction of a GLP.

of simulation cells. For instance, the simulation cells used for training in this thesis admit two to three interaction steps before symmetrically equivalent atoms are reached.

Beyond such empirical considerations, the question of what kinds of physical interactions can be expressed with message-passing mechanisms must also be considered. The information bottleneck induced by such mechanisms, where information about atomic environments must be compressed into a feature vector on each atom, restricts the types of interaction that can be modeled.[7] Further restrictions are imposed by symmetry. To further improve such models, a systematic investigation of relevant physical mechanisms and the ability of MPNNs to capture them is required.

[7] In the ML literature, these problems are discussed as *oversquashing* [452] of long-distance messages, and *oversmoothing* [453, 454], where node states become indistinct as M increases.

Perspective

We conclude this thesis by outlining general challenges and developments related to modeling periodic systems with long-range effects.

Training with Extended Systems The training of MLIPs for solids with defects and dopants, or more general disruptions of periodicity, at low concentrations presents a number of challenges. Even assuming that first-principles calculations are feasible for the phenomenon of interest, large simulations calls may be required, limiting the amount of available training data. It may therefore be required, in the spirit of the GEMS approach recently developed to model large bio-molecules [131] with a MLIP based on SpookyNet [130], to train jointly on few large-scale 'top-down' calculations to capture long-range behaviour, and 'bottom-up' calculations in smaller simulation cells to obtain sufficient data for local interactions.

An early mathematical perspective on the question of generalisation to large simulation cells is given in reference [455].

However, consistent training in such settings requires models with awareness of long-range structure: Local and semi-local models are unaware of structure changes outside of the effective interaction radius r_c^{eff}, and can therefore not be expected to effectively learn in such a setting. This motivates the use of *global* models, discussed below.

An additional consideration is the treatment of different energy scales arising from short- and long-range contributions. Effective training, especially if models with separate components for different ranges are employed, may require range-separated energy and force labels, which are not generally available in electronic structure codes.

Global Models in Solids Local and semi-local models retain a locality assumption through the cutoff radius r_c. Beyond this assumption, global, or long-range, models have been proposed. Here, it is instructive to distinguish between long-range (pairwise) energy contributions, for instance through electrostatics or pairwise approximations to van der Waals interactions, and non-local or global models [43, 127, 130] that can consider all atoms in the system at once.

See reference [456] for a recent review of models for extended systems.

While methods are available to evaluate the former, for instance through Ewald summation [99, 100] or fast multipole methods [101,

102], which are routinely employed in many FFs, the latter presents a challenge in solids. In that case, a formally infinite number of positions must be considered. One way to approach this task may be taking inspiration from Ewald summation and evaluate the required expressions in reciprocal space. Such approaches have been used to compute descriptors for periodic systems [168, 169, 457], and have recently been proposed for MPNNs [458]. An alternative is to retain physical interactions, where approaches for periodic systems have been developed already. Two routes present themselves at this point: The first approach is predicting parameters for physically-known energy expressions, for instance charges for electrostatics [385], in which case the parameters do not depend on such interactions, and some of the limits of locality are retained. The second approach is to allow iterative optimisation of parameters based on long-range energy, for instance in NNs that incorporate charge equilibration [114].

Solvers in Models Beyond the direct modeling of physical interactions, the combination of numerical solvers, differential equations [197], and NNs presents intriguing possibilities. In essence, it enables the use of physical mechanisms as general interactions in NNs. For instance, the use of physics-inspired interactions for MPNNs has recently been proposed. In one example, interactions steps then play the rule of successive iterations of solving a problem of coupled harmonic oscillators [459]. Such systems also emerge in recent treatments of many-body dispersion [460, 461]; techniques could potentially be shared.

In this context, AD-based implicit differentiation [450] can also play an important role to obtain derivatives. The ability to include differential equations and dynamics into NNs may also open the road to a tighter coupling between quantum chemistry and machine learning methods, allowing learned functional forms to replace hand-tuned approximations, and conversely, using well-understood physical mechanisms as building blocks for novel NN architectures.

Bibliography

[1] Usage data obtained from `https://www.archer2.ac.uk/support-access/status.html`, on 2022-12-26. Keys for QM software: `vasp`, `castep`, `nwchem`, `onetep`, `siesta`, `cpmd`, `elk`, `gpaw`, `fhi aims`, `hande`, `rmt`, `casino`, `ukrmol+`, `xcompact3d`, `crystal`, `quantum espresso`, `chemshell`, and `cp2k`. Keys for MD software: `gromacs`, `amber`, `lammps`, `dl poly`, and `namd`. (cit. on p. 11).

[2] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, 'Equation of State Calculations by Fast Computing Machines', The Journal of Chemical Physics **21**, 1087 (1953) (cit. on p. 11).

[3] W. W. Wood and F. R. Parker, 'Monte Carlo Equation of State of Molecules Interacting with the Lennard-Jones Potential. I. A Supercritical Isotherm at about Twice the Critical Temperature', The Journal of Chemical Physics **27**, 720 (1957) (cit. on p. 11).

[4] A. Rahman, 'Correlations in the Motion of Atoms in Liquid Argon', Physical Review **136**, A405 (1964) (cit. on p. 11).

[5] L. Verlet, 'Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules', Physical Review **159**, 98 (1967) (cit. on pp. 11, 25, 27, 45).

[6] J. W. Ponder and D. A. Case, 'Force Fields for Protein Simulations', Advances in Protein Chemistry, 27 (2003) (cit. on pp. 11, 29).

[7] M. González, 'Force fields and molecular dynamics simulations', Journées de la Neutronique **12**, 169 (2011) (cit. on pp. 11, 27).

[8] E. O. Pyzer-Knapp, C. Suh, R. Gómez-Bombarelli, J. Aguilera-Iparraguirre and A. Aspuru-Guzik, 'What Is High-Throughput Virtual Screening? A Perspective from Organic Materials Discovery', Annual Review of Materials Research **45**, 195 (2015) (cit. on pp. 11, 29, 40).

[9] P. Hohenberg and W. Kohn, 'Inhomogeneous Electron Gas', Physical Review **136**, B864 (1964) (cit. on pp. 11, 22).

[10] W. Kohn and L. J. Sham, 'Self-Consistent Equations Including Exchange and Correlation Effects', Physical Review **140**, A1133 (1965) (cit. on pp. 11, 22).

[11] A. M. Teale, T. Helgaker, A. Savin, C. Adamo, B. Aradi, A. V. Arbuznikov, P. W. Ayers, E. J. Baerends, V. Barone, P. Calaminici, E. Cancès, E. A. Carter, P. K. Chattaraj, H. Chermette, I. Ciofini, T. D. Crawford, F. De Proft, J. F. Dobson, C. Draxl, T. Frauenheim, E. Fromager, P. Fuentealba, L. Gagliardi, G. Galli, J. Gao, P. Geerlings, N. Gidopoulos, P. M. W. Gill, P. Gori-Giorgi, A. Görling, T. Gould, S. Grimme, O. Gritsenko, H. J. A. Jensen, E. R. Johnson, R. O. Jones, M. Kaupp, A. M. Köster, L. Kronik, A. I. Krylov, S. Kvaal, A. Laestadius, M. Levy, M. Lewin, S. Liu, P.-F. Loos, N. T. Maitra, F. Neese, J. P. Perdew, K. Pernal, P. Pernot, P. Piecuch, E. Rebolini, L. Reining, P. Romaniello, A. Ruzsinszky, D. R. Salahub, M. Scheffler, P. Schwerdtfeger, V. N. Staroverov, J. Sun, E. Tellgren, D. J. Tozer, S. B. Trickey, C. A. Ullrich, A. Vela, G. Vignale, T. A. Wesolowski, X. Xu and W. Yang, 'DFT exchange: sharing perspectives on the workhorse of quantum chemistry and materials science', Physical Chemistry Chemical Physics **24**, 28700 (2022) (cit. on pp. 11, 46).

[12] R. Car and M. Parrinello, 'Unified Approach for Molecular Dynamics and Density-Functional Theory', Physical Review Letters **55**, 2471 (1985) (cit. on pp. 11, 27).

[13] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. (Springer, New York, 2009) (cit. on pp. 11, 37, 41).

[14] S. Lorenz, A. Groß and M. Scheffler, 'Representing high-dimensional potential-energy surfaces for reactions at surfaces by neural networks', Chemical Physics Letters **395**, 210 (2004) (cit. on pp. 12, 39).

[15] G. Li, J. Hu, S.-W. Wang, P. G. Georgopoulos, J. Schoendorf and H. Rabitz, 'Random Sampling-High Dimensional Model Representation (RS-HDMR) and Orthogonality of Its Different Order Component Functions', The Journal of Physical Chemistry A **110**, 2474 (2006) (cit. on pp. 12, 30, 39).

[16] J. Behler and M. Parrinello, 'Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces', Physical Review Letters **98**, 146401 (2007) (cit. on pp. 12, 30, 39, 58, 60).

[17] A. P. Bartók, M. C. Payne, R. Kondor and G. Csányi, 'Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons', Physical Review Letters **104**, 136403 (2010) (cit. on pp. 12, 30, 39, 58, 62).

[18] O. T. Unke, D. Koner, S. Patra, S. Käser and M. Meuwly, 'High-Dimensional Potential Energy Surfaces for Molecular Simulations: From Empiricism to Machine Learning', Machine Learning: Science and Technology **1**, 013001 (2020) (cit. on pp. 12, 30, 39).

[19] O. T. Unke, S. Chmiela, H. E. Sauceda, M. Gastegger, I. Poltavsky, K. T. Schütt, A. Tkatchenko and K.-R. Müller, 'Machine Learning Force Fields', Chemical Reviews **121**, 10142 (2021) (cit. on pp. 12, 30, 39).

[20] I. Poltavsky and A. Tkatchenko, 'Machine Learning Force Fields: Recent Advances and Remaining Challenges', Journal of Physical Chemistry Letters **12**, 6551 (2021) (cit. on pp. 12, 30, 39).

[21] M. Rupp, A. Tkatchenko, K.-R. Müller and O. A. von Lilienfeld, 'Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning', Physical Review Letters **108**, 058301 (2012) (cit. on pp. 12, 40, 58, 63).

[22] B. Huang and O. A. von Lilienfeld, 'Ab Initio Machine Learning in Chemical Compound Space', Chemical Reviews **121**, 10001 (2021) (cit. on pp. 12, 40).

[23] J. Kirkpatrick, B. McMorrow, D. H. P. Turban, A. L. Gaunt, J. S. Spencer, A. G. D. G. Matthews, A. Obika, L. Thiry, M. Fortunato, D. Pfau, L. R. Castellanos, S. Petersen, A. W. R. Nelson, P. Kohli, P. Mori-Sánchez, D. Hassabis and A. J. Cohen, 'Pushing the frontiers of density functionals by solving the fractional electron problem', Science **374**, 1385 (2021) (cit. on pp. 12, 23).

[24] Z. M. Sparrow, B. G. Ernst, T. K. Quady and R. A. DiStasio, 'Uniting Nonempirical and Empirical Density Functional Approximation Strategies Using Constraint-Based Regularization', Journal of Physical Chemistry Letters **13**, 6896 (2022) (cit. on p. 12).

[25] J. Hermann, J. Spencer, K. Choo, A. Mezzacapo, W. M. C. Foulkes, D. Pfau, G. Carleo and F. Noé, *Ab-initio quantum chemistry with neural-network wavefunctions*, 26th Aug. 2022, eprint: `arXiv:2208.12590` (cit. on pp. 12, 21).

[26] T. Mueller, A. G. Kusne and R. Ramprasad, 'Machine Learning in Materials Science', in *Reviews in Computational Chemistry* (Wiley, 6th May 2016), pp. 186–273 (cit. on p. 12).

[27] B. Huang, N. O. Symonds and O. A. von Lilienfeld, 'Quantum Machine Learning in Chemistry and Materials', in *Handbook of Materials Modeling. Methods: Theory and Modeling* (Springer, 22nd June 2018) (cit. on p. 12).

[28] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto and L. Zdeborová, 'Machine learning and the physical sciences', Reviews of Modern Physics **91**, 045002 (2019) (cit. on p. 12).

[29] F. Noé, A. Tkatchenko, K.-R. Müller and C. Clementi, 'Machine Learning for Molecular Simulation', Annual Review of Physical Chemistry **71**, 361 (2020) (cit. on p. 12).

[30] K. T. Schütt, S. Chmiela, O. A. von Lilienfeld, A. Tkatchenko, K. Tsuda and K.-R. Müller, eds., *Machine Learning Meets Quantum Physics*, Vol. 968, Lecture Notes in Physics (Springer, 1st Jan. 2020) (cit. on p. 12).

[31] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals and G. E. Dahl, 'Neural Message Passing for Quantum Chemistry', in Proceedings of the 34th International Conference on Machine Learning (ICML 2017), Sydney, Australia, August 6–11 (2017), pp. 1263–1272 (cit. on pp. 12, 39, 79, 81–83).

[32] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard and L. Jackel, 'Handwritten Digit Recognition with a Back-Propagation Network', in Advances in Neural Information Processing Systems 2 (NIPS 1989), Denver, Colorado, USA, Nov. 27–30 (1989) (cit. on pp. 12, 42).

[33] A. Krizhevsky, I. Sutskever and G. E. Hinton, 'ImageNet Classification with Deep Convolutional Neural Networks', in Advances in Neural Information Processing Systems 25 (NIPS 2012), Lake Tahoe, USA, Dec 3–Dec 8 (2012) (cit. on pp. 12, 42).

[34] J. Schmidhuber, 'Deep learning in neural networks: An overview', Neural Networks **61**, 85 (2015) (cit. on p. 12).

[35] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Y. Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu and Xiaoqiang Zheng, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, Software available from tensorflow.org, 2015 (cit. on p. 12).

[36] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala, *PyTorch: An Imperative Style, High-Performance Deep Learning Library*, 3rd Dec. 2019, eprint: `arXiv:1912.01703` (cit. on p. 12).

[37] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne and Q. Zhang, *JAX: composable transformations of Python+NumPy programs*, 2018 (cit. on pp. 12, 87, 90, 130, 216).

[38] D. E. Rumelhart, G. E. Hinton and R. J. Williams, 'Learning representations by back-propagating errors', Nature **323**, 533 (1986) (cit. on pp. 12, 42).

[39] A. Griewank and A. Walther, *Evaluating Derivatives, Principles and Techniques of Algorithmic Differentiation, Second Edition* (Society for Industrial and Applied Mathematics, Jan. 2008) (cit. on pp. 12, 42, 130).

[40] A. G. Baydin, B. A. Pearlmutter, A. A. Radul and J. M. Siskind, 'Automatic Differentiation in Machine Learning: A Survey', Journal of Machine Learning Research **18**, 5595 (2017) (cit. on pp. 12, 42).

[41] K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko and K.-R. Müller, 'SchNet: A continuous-filter convolutional neural network for modeling quantum interactions', in Advances in Neural Information Processing Systems 30 (NIPS 2017), Los Angeles, California, December 4–9 (2017), pp. 992–1002 (cit. on pp. 13, 79, 82, 84, 101, 103, 125, 128).

[42] K. T. Schütt, H. E. Sauceda, P.-J. Kindermans, A. Tkatchenko and K.-R. Müller, 'SchNet—a deep learning architecture for molecules and materials', The Journal of Chemical Physics **148**, 241722 (2018) (cit. on pp. 13, 79, 82, 84, 101, 128, 178).

[43] T. Frank, O. Unke and K.-R. Müller, 'So3krates: Equivariant attention for interactions on arbitrary length-scales in molecular systems', in Advances in Neural Information Processing Systems 35 (NeurIPS 2022), New Orleans, Louisiana, USA, Nov 28–Dec 9 (2022), pp. 29400–29413 (cit. on pp. 13, 39, 77, 84, 87, 99, 101, 119, 125, 128, 131).

[44] M. F. Langer, A. Goeßmann and M. Rupp, 'Representations of molecules and materials for interpolation of quantum-mechanical simulations via machine learning', Nature Partner Journal Computational Materials **8**, 41 (2022) (cit. on pp. 15, 41, 52, 70, 71, 173).

[45] M. F. Langer, F. Knoop, C. Carbogno, M. Scheffler and M. Rupp, *Heat flux for semi-local machine-learning potentials*, 25th Mar. 2023, eprint: `arXiv:2303.14434` (cit. on pp. 15, 80, 102).

[46] M. F. Langer, J. T. Frank and F. Knoop, *Stress and heat flux via automatic differentiation*, 2nd May 2023, eprint: `arXiv:2305.01401` (cit. on pp. 15, 80, 87, 102).

[47] P. Dirac, *The Principles of Quantum Mechanics*, Fourth Edition (Oxford University Press, 1953) (cit. on p. 19).

[48] J. J. Sakurai and J. Napolitano, *Modern Quantum Mechanics*, Second Edition (Addison-Wesley, San Francisco, 2011) (cit. on pp. 19, 20, 23, 183).

[49] J. Hermann, 'Towards unified density-functional model of van der Waals interactions', PhD thesis (Humboldt-Universität zu Berlin, 2018) (cit. on p. 19).

[50] S. Kokott, 'First-principles Investigation of Small Polarons in Metal Oxides', PhD thesis (Humboldt-Universität zu Berlin, 2018) (cit. on p. 19).

[51] E. van Lenthe, E. J. Baerends and J. G. Snijders, 'Relativistic total energy using regular approximations', The Journal of Chemical Physics **101**, 9783 (1994) (cit. on p. 20).

[52] W. P. Huhn and V. Blum, 'One-hundred-three compound band-structure benchmark of post-self-consistent spin-orbit coupling treatments in density functional theory', Physical Review Materials **1**, 033803 (2017) (cit. on p. 20).

[53] R. Zhao, V. W.-z. Yu, K. Zhang, Y. Xiao, Y. Zhang and V. Blum, 'Quasi-four-component method with numeric atom-centered orbitals for relativistic density functional simulations of molecules and solids', Physical Review B **103**, 245144 (2021) (cit. on p. 20).

[54] W. Pauli, 'The Connection Between Spin and Statistics', Physical Review **58**, 716 (1940) (cit. on pp. 20, 21).

[55] M. Born and R. Oppenheimer, 'Zur Quantentheorie der Molekeln', Annalen der Physik **389**, 457 (1927) (cit. on p. 20).

[56] N. E. Ashcrof and N. D. Mermin, *Solid State Physics*, College Edition (Harcourt College Publishers, 1976) (cit. on pp. 20, 31, 33).

[57] A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory* (Dover Publications, 1989) (cit. on p. 20).

[58] W. Foulkes, 'Variational Wave Functions for Molecules and Solids', in *Topology, Entanglement, and Strong Correlations* (Forschungszentrum Jülich Verlag, 2020) (cit. on p. 20).

[59] M. J. Gander and G. Wanner, 'From Euler, Ritz, and Galerkin to Modern Computing', SIAM Review **54**, 627 (2012) (cit. on p. 20).

[60] D. R. Hartree, 'The Wave Mechanics of an Atom with a Non-Coulomb Central Field. Part II. Some Results and Discussion', Mathematical Proceedings of the Cambridge Philosophical Society **24**, 111 (1928) (cit. on p. 21).

[61] V. Fock, 'Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems', Zeitschrift für Physik **61**, 126 (1930) (cit. on p. 21).

[62] R. M. Martin, *Electronic Structure* (Cambridge University Press, Aug. 2020) (cit. on pp. 22–24, 39).

[63] P. Jakobsen and F. Jensen, 'Representing Exact Electron Densities by a Single Slater Determinant in Finite Basis Sets', Journal of Chemical Theory and Computation **17**, 269 (2020) (cit. on p. 22).

[64] W. Kohn, 'Density Functional and Density Matrix Method Scaling Linearly with the Number of Atoms', Physical Review Letters **76**, 3168 (1996) (cit. on p. 23).

[65] S. Goedecker, 'Linear scaling electronic structure methods', Reviews of Modern Physics **71**, 1085 (1999) (cit. on p. 23).

[66] D. R. Bowler and T. Miyazaki, 'O(N) methods in electronic structure calculations', Reports on Progress in Physics **75**, 036503 (2012) (cit. on p. 23).

[67] S. Grimme, 'Semiempirical hybrid density functional with perturbative second-order correlation', The Journal of Chemical Physics **124**, 034108 (2006) (cit. on p. 23).

[68] D. M. Ceperley and B. J. Alder, 'Ground State of the Electron Gas by a Stochastic Method', Physical Review Letters **45**, 566 (1980) (cit. on p. 23).

[69] J. P. Perdew, K. Burke and M. Ernzerhof, 'Generalized Gradient Approximation Made Simple', Physical Review Letters **77**, 3865 (1996) (cit. on pp. 23, 68).

[70] J. P. Perdew, A. Ruzsinszky, G. I. Csonka, O. A. Vydrov, G. E. Scuseria, L. A. Constantin, X. Zhou and K. Burke, 'Restoring the Density-Gradient Expansion for Exchange in Solids and Surfaces', Physical Review Letters **100**, 136406 (2008) (cit. on pp. 23, 103, 122).

[71] J. C. Snyder, M. Rupp, K. Hansen, K.-R. Müller and K. Burke, 'Finding Density Functionals with Machine Learning', Physical Review Letters **108**, 253002 (2012) (cit. on p. 23).

[72] H. J. Kulik, T. Hammerschmidt, J. Schmidt, S. Botti, M. A. L. Marques, M. Boley, M. Scheffler, M. Todorović, P. Rinke, C. Oses, A. Smolyanyuk, S. Curtarolo, A. Tkatchenko, A. P. Bartók, S. Manzhos, M. Ihara, T. Carrington, J. Behler, O. Isayev, M. Veit, A. Grisafi, J. Nigam, M. Ceriotti, K. T. Schütt, J. Westermayr, M. Gastegger, R. J. Maurer, B. Kalita, K. Burke, R. Nagai, R. Akashi, O. Sugino, J. Hermann, F. Noé, S. Pilati, C. Draxl, M. Kuban, S. Rigamonti, M. Scheidgen, M. Esters, D. Hicks, C. Toher, P. V. Balachandran, I. Tamblyn, S. Whitelam, C. Bellinger and L. M. Ghiringhelli, 'Roadmap on Machine learning in electronic structure', Electronic Structure **4**, 023004 (2022) (cit. on p. 23).

[73] R. P. Feynman, 'Forces in Molecules', Physical Review **56**, 340 (1939) (cit. on p. 24).

[74] P. Pernot, *The long road to calibrated prediction uncertainty in computational chemistry*, 5th Jan. 2022, eprint: `arXiv:2201.01511` (cit. on p. 24).

[75] P. Pulay, 'Ab initio calculation of force constants and equilibrium geometries in polyatomic molecules', Molecular Physics **17**, 197 (1969) (cit. on p. 24).

[76] F. Knuth, C. Carbogno, V. Atalla, V. Blum and M. Scheffler, 'All-electron formalism for total energy strain derivatives and stress tensor components for numeric atom-centered orbitals', Computer Physics Communications **190**, 33 (2015) (cit. on pp. 24, 86).

[77] M. E. Tuckerman, B. J. Berne, G. J. Martyna and M. L. Klein, 'Efficient molecular dynamics and hybrid Monte Carlo algorithms for path integrals', The Journal of Chemical Physics **99**, 2796 (1993) (cit. on p. 25).

[78] M. E. Tuckerman, *Statistical Mechanics: Theory and Molecular Simulation* (Oxford University Press, 2010) (cit. on pp. 25, 26, 31).

[79] T. Schneider and E. Stoll, 'Molecular-dynamics study of a three-dimensional one-component model for distortive phase transitions', Physical Review B **17**, 1302 (1978) (cit. on p. 26).

[80] E. Vanden-Eijnden and G. Ciccotti, 'Second-order integrators for Langevin equations with holonomic constraints', Chemical Physics Letters **429**, 310 (2006) (cit. on p. 26).

[81] G. Bussi and M. Parrinello, 'Accurate sampling using Langevin dynamics', Physical Review E **75**, 056707 (2007) (cit. on p. 26).

[82] S. Nosé, 'A unified formulation of the constant temperature molecular dynamics methods', The Journal of Chemical Physics **81**, 511 (1984) (cit. on p. 26).

[83] W. G. Hoover, 'Canonical dynamics: Equilibrium phase-space distributions', Physical Review A **31**, 1695 (1985) (cit. on p. 26).

[84] G. J. Martyna, M. L. Klein and M. Tuckerman, 'Nosé-Hoover chains: The canonical ensemble via continuous dynamics', The Journal of Chemical Physics **97**, 2635 (1992) (cit. on p. 26).

[85] G. Bussi, D. Donadio and M. Parrinello, 'Canonical sampling through velocity rescaling', The Journal of Chemical Physics **126**, 014101 (2007) (cit. on pp. 26, 106).

[86] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola and J. R. Haak, 'Molecular dynamics with coupling to an external bath', The Journal of Chemical Physics **81**, 3684 (1984) (cit. on pp. 26, 103).

[87] M. R. Shirts, 'Simple Quantitative Tests to Validate Sampling from Thermodynamic Ensembles', Journal of Chemical Theory and Computation **9**, 909 (2013) (cit. on p. 26).

[88] A. H. Larsen, J. J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. B. Jensen, J. Kermode, J. R. Kitchin, E. L. Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. B. Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng and K. W. Jacobsen, 'The atomic simulation environment—a Python library for working with atoms', Journal of Physics: Condensed Matter **29**, 273002 (2017) (cit. on pp. 26, 104).

[89] F. Knoop, M. Scheffler, C. Carbogno et al., 'FHI-vibes: Ab Initio Vibrational Simulations', Journal of Open Source Software **5**, 2671 (2020) (cit. on pp. 26, 104).

[90] G. Bussi, T. Zykova-Timan and M. Parrinello, 'Isothermal-isobaric molecular dynamics using stochastic velocity rescaling', The Journal of Chemical Physics **130**, 074101 (2009) (cit. on p. 26).

[91] M. Bernetti and G. Bussi, 'Pressure control using stochastic cell rescaling', The Journal of Chemical Physics **153**, 114107 (2020) (cit. on p. 26).

[92] V. Del Tatto, P. Raiteri, M. Bernetti and G. Bussi, 'Molecular Dynamics of Solids at Constant Pressure and Stress Using Anisotropic Stochastic Cell Rescaling', Applied Sciences **12**, 1139 (2022) (cit. on p. 26).

[93] G. J. Martyna, D. J. Tobias and M. L. Klein, 'Constant pressure molecular dynamics algorithms', The Journal of Chemical Physics **101**, 4177 (1994) (cit. on pp. 26, 106).

[94] P. Raiteri, J. D. Gale and G. Bussi, 'Reactive force field simulation of proton diffusion in BaZrO3using an empirical valence bond approach', Journal of Physics: Condensed Matter **23**, 334213 (2011) (cit. on p. 26).

[95] R. Iftimie, P. Minary and M. E. Tuckerman, 'Ab initio molecular dynamics: Concepts, recent developments, and future trends', Proceedings of the National Academy of Sciences of the United States of America **102**, 6654 (2005) (cit. on p. 27).

[96] B. J. Alder and T. E. Wainwright, 'Phase Transition for a Hard Sphere System', The Journal of Chemical Physics **27**, 1208 (1957) (cit. on pp. 27, 28).

[97] B. J. Alder and T. E. Wainwright, 'Studies in Molecular Dynamics. I. General Method', The Journal of Chemical Physics **31**, 459 (1959) (cit. on pp. 27, 28).

[98] V. Ballenegger, 'Communication: On the origin of the surface term in the Ewald formula', The Journal of Chemical Physics **140**, 161102 (2014) (cit. on p. 28).

[99] P. P. Ewald, 'Die Berechnung optischer und elektrostatischer Gitterpotentiale', Annalen der Physik **369**, 253 (1921) (cit. on pp. 28, 82, 94, 131).

[100] T. Darden, D. York and L. Pedersen, 'Particle mesh Ewald: AnN·log(N) method for Ewald sums in large systems', The Journal of Chemical Physics **98**, 10089 (1993) (cit. on pp. 28, 82, 131).

[101] L. Greengard and V. Rokhlin, 'A fast algorithm for particle simulations', Journal of Computational Physics **73**, 325 (1987) (cit. on pp. 28, 82, 131).

[102] C. A. White and M. Head-Gordon, 'Derivation and efficient implementation of the fast multipole method', The Journal of Chemical Physics **101**, 6593 (1994) (cit. on pp. 28, 82, 131).

[103] J. E. Lennard-Jones, 'On the determination of molecular fields: I. From the variation of the viscosity of a gas with temperature', Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character **106**, 441 (1924) (cit. on p. 28).

[104] R. A. Buckingham and J. E. Lennard-Jones, 'The classical equation of state of gaseous helium, neon and argon', Proceedings of the Royal Society of London, Series A **168**, 264 (1938) (cit. on p. 28).

[105] P. M. Morse, 'Diatomic Molecules According to the Wave Mechanics. II. Vibrational Levels', Physical Review **34**, 57 (1929) (cit. on p. 28).

[106] M. S. Daw and M. I. Baskes, 'Semiempirical, Quantum Mechanical Calculation of Hydrogen Embrittlement in Metals', Physical Review Letters **50**, 1285 (1983) (cit. on p. 28).

[107] M. S. Daw and M. I. Baskes, 'Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals', Physical Review B **29**, 6443 (1984) (cit. on p. 28).

[108] F. H. Stillinger and T. A. Weber, 'Computer simulation of local order in condensed phases of silicon', Physical Review B **31**, 5262 (1985) (cit. on pp. 28, 89).

[109] J. Tersoff, 'New empirical approach for the structure and energy of covalent systems', Physical Review B **37**, 6991 (1988) (cit. on pp. 28, 89).

[110] D. W. Brenner, 'Empirical potential for hydrocarbons for use in simulating the chemical vapor deposition of diamond films', Physical Review B **42**, 9458 (1990) (cit. on pp. 28, 89).

[111] P. Cieplak, F.-Y. Dupradeau, Y. Duan and J. Wang, 'Polarization effects in molecular mechanical force fields', Journal of Physics: Condensed Matter **21**, 333102 (2009) (cit. on p. 29).

[112] W. J. Mortier, S. K. Ghosh and S. Shankar, 'Electronegativity-equalization method for the calculation of atomic charges in molecules', Journal of the American Chemical Society **108**, 4315 (1986) (cit. on p. 29).

[113] A. K. Rappe and W. A. Goddard, 'Charge equilibration for molecular dynamics simulations', The Journal of Chemical Physics **95**, 3358 (1991) (cit. on p. 29).

[114] J. Behler, 'Four Generations of High-Dimensional Neural Network Potentials', Chemical Reviews **121**, 10037 (2021) (cit. on pp. 29, 39, 83, 132).

[115] A. C. T. van Duin, S. Dasgupta, F. Lorant and W. A. Goddard, 'ReaxFF: A Reactive Force Field for Hydrocarbons', The Journal of Physical Chemistry A **105**, 9396 (2001) (cit. on p. 29).

[116] T. P. Senftle, S. Hong, M. M. Islam, S. B. Kylasa, Y. Zheng, Y. K. Shin, C. Junkermeier, R. Engel-Herbert, M. J. Janik, H. M. Aktulga, T. Verstraelen, A. Grama and A. C. T. van Duin, 'The ReaxFF reactive force-field: development, applications and future directions', Nature Partner Journal Computational Materials **2**, 1 (2016) (cit. on p. 29).

[117] J. Yu, S. B. Sinnott and S. R. Phillpot, 'Charge optimized many-body potential for the SiO system', Physical Review B **75**, 085311 (2007) (cit. on p. 29).

[118] T. Liang, B. Devine, S. R. Phillpot and S. B. Sinnott, 'Variable Charge Reactive Potential for Hydrocarbons to Simulate Organic-Copper Interactions', The Journal of Physical Chemistry A **116**, 7976 (2012) (cit. on p. 29).

[119] M. I. Baskes, 'Application of the Embedded-Atom Method to Covalent Materials: A Semiempirical Potential for Silicon', Physical Review Letters **59**, 2666 (1987) (cit. on p. 29).

[120] C. M. Daksha, J. Yeon, S. C. Chowdhury and J. W. Gillespie Jr., 'Automated ReaxFF parametrization using machine learning', Computational Materials Science **187**, 110107 (2021) (cit. on p. 30).

[121] M. C. Kaymak, A. Rahnamoun, K. A. O'Hearn, A. C. T. van Duin, K. M. Merz and H. M. Aktulga, 'JAX-ReaxFF: A Gradient-Based Framework for Fast Optimization of Reactive Force Fields', Journal of Chemical Theory and Computation **18**, 5181 (2022) (cit. on p. 30).

[122] J. M. Bowman, J. S. Bittman and L. B. Harding, '*Ab initio* calculations of electronic and vibrational energies of HCO and HOC', The Journal of Chemical Physics **85**, 911 (1986) (cit. on p. 30).

[123] J. A. Darsey, D. W. Noid and B. R. Upadhyaya, 'Application of neural network computing to the solution for the ground-state eigenenergy of two-dimensional harmonic oscillators', Chemical Physics Letters **177**, 189 (1991) (cit. on p. 30).

[124] H. Heo, T.-S. Ho, K. K. Lehmann and H. Rabitz, 'Regularized inversion of diatomic vibration-rotation spectral data: A functional sensitivity analysis approach', The Journal of Chemical Physics **97**, 852 (1992) (cit. on p. 30).

[125] T. Hollebeek, T.-S. Ho and H. Rabitz, 'Constructing multidimensional molecular potential energy surfaces from ab initio data', Annual Review of Physical Chemistry **50**, 537 (1999) (cit. on p. 30).

[126] X. Fu, Z. Wu, W. Wang, T. Xie, S. Keten, R. Gomez-Bombarelli and T. Jaakkola, *Forces are not Enough: Benchmark and Critical Evaluation for Machine Learning Force Fields with Molecular Simulations*, 13th Oct. 2022, eprint: `arXiv:2210.07237` (cit. on pp. 30, 105, 130, 173).

[127] S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt and K.-R. Müller, 'Machine Learning of Accurate Energy-Conserving Molecular Force Fields', Science Advances **3**, e1603015 (2017) (cit. on pp. 30, 39, 99, 131).

[128] S. Chmiela, H. E. Sauceda, K.-R. Müller and A. Tkatchenko, 'Towards Exact Molecular Dynamics Simulations with Machine-Learned Force Fields', Nature Communications **9**, 3887 (2018) (cit. on pp. 30, 55, 99).

[129] S. Chmiela, H. E. Sauceda, I. Poltavsky, K.-R. Müller and A. Tkatchenko, 'sGDML: Constructing Accurate and Data Efficient Molecular Force Fields Using Machine Learning', Computer Physics Communications **240**, 38 (2019) (cit. on pp. 30, 39, 99).

[130] O. T. Unke, S. Chmiela, M. Gastegger, K. T. Schütt, H. E. Sauceda and K.-R. Müller, 'SpookyNet: Learning force fields with electronic degrees of freedom and nonlocal effects', Nature Communications **12**, 7273 (2021) (cit. on pp. 30, 39, 79, 82, 84, 99, 131).

[131] O. T. Unke, M. Stöhr, S. Ganscha, T. Unterthiner, H. Maennel, S. Kashubin, D. Ahlin, M. Gastegger, L. Medrano Sandonas, A. Tkatchenko and K.-R. Müller, *Accurate Machine Learned Quantum-Mechanical Force Fields for Biomolecular Simulations*, 17th May 2022, eprint: `arXiv:2205.08306` (cit. on pp. 30, 39, 131).

[132] S. R. Xie, M. Rupp and R. G. Hennig, *Ultra-fast interpretable machine-learning potentials*, 1st Oct. 2021, eprint: `arXiv:2110.00624` (cit. on p. 30).

[133] A. Johansson, Y. Xie, C. J. Owen, J. S. (Lim, L. Sun, J. Vandermause and B. Kozinsky, *Micron-scale heterogeneous catalysis with Bayesian force fields from first principles and active learning*, 26th Apr. 2022, eprint: `arXiv:2204.12573` (cit. on pp. 30, 31).

[134] S. Pozdnyakov, A. R. Oganov, E. Mazhnik, A. Mazitov and I. Kruglov, 'Fast general two- and three-body interatomic potential', Physical Review B **107**, 125160 (2023) (cit. on p. 30).

[135] L.-Y. Xue, F. Guo, Y.-S. Wen, S.-Q. Feng, X.-N. Huang, L. Guo, H.-S. Li, S.-X. Cui, G.-Q. Zhang and Q.-L. Wang, 'ReaxFF-MPNN machine learning potential: a combination of reactive force field and message passing neural networks', Physical Chemistry Chemical Physics **23**, 19457 (2021) (cit. on p. 30).

[136] E. V. Podryabinkin, A. G. Kvashnin, M. Asgarpour, I. I. Maslenikov, D. A. Ovsyannikov, P. B. Sorokin, M. Y. Popov and A. V. Shapeev, 'Nanohardness from First Principles with Active Learning on Atomic Environments', Journal of Chemical Theory and Computation **in press**, in press (2022) (cit. on p. 31).

[137] R. C. Dennis, V. F. Hagh and E. I. Corwin, 'Emergence of zero modes in disordered solids under periodic tiling', Physical Review E **106**, 044901 (2022) (cit. on p. 31).

[138] F. Bloch, 'Über die Quantenmechanik der Elektronen in Kristallgittern', Zeitschrift für Physik **52**, 555 (1929) (cit. on p. 33).

[139] G. Floquet, 'Sur les équations différentielles linéaires à coefficients périodiques', Annales scientifiques de l'École normale supérieure **12**, 47 (1883) (cit. on p. 33).

[140] D. J. Chadi and M. L. Cohen, 'Special Points in the Brillouin Zone', Physical Review B **8**, 5747 (1973) (cit. on p. 34).

[141] H. J. Monkhorst and J. D. Pack, 'Special points for Brillouin-zone integrations', Physical Review B **13**, 5188 (1976) (cit. on p. 34).

[142] F. Knoop, 'Heat transport in strongly anharmonic solids from first principles', PhD thesis (Humbold-Universiät zu Berlin, 2021) (cit. on pp. 34, 47, 102, 119, 189).

[143] C. Carbogno, R. Ramprasad and M. Scheffler, 'Ab Initio Green-Kubo Approach for the Thermal Conductivity of Solids', Physical Review Letters **118**, 175901 (2017) (cit. on pp. 34, 46, 79, 90, 91, 94, 101, 103, 104, 117, 121–123, 185).

[144] F. Knoop, M. Scheffler and C. Carbogno, 'Ab initio Green-Kubo simulations of heat transport in solids: Method and implementation', Physical Review B **107**, 224304 (2023) (cit. on pp. 34, 46, 49, 101, 109, 112, 204).

[145] W. Smith, *The Minimum Image Convention in Non-Cubic MD Cells*, 1989 (cit. on p. 35).

[146] M. P. Allen, 'Back to Basics', in *Computer Simulation in Chemical Physics* (Springer Netherlands, 1993), pp. 49–92 (cit. on pp. 35, 48, 93).

[147] J. J. Erpenbeck, 'Einstein-Kubo-Helfand and McQuarrie relations for transport coefficients', Physical Review E **51**, 4296 (1995) (cit. on pp. 35, 93).

[148] R. Resta and D. Vanderbilt, 'Theory of Polarization: A Modern Approach', in *Physics of Ferroelectrics: A Modern Perspective* (Springer Berlin Heidelberg, 2007), pp. 31–68 (cit. on p. 36).

[149] E. Valença Ferreira de Arag ao, D. Moreno, S. Battaglia, G. L. Bendazzoli, S. Evangelisti, T. Leininger, N. Suaud and J. A. Berger, 'A simple position operator for periodic systems', Physical Review B **99**, 205144 (2019) (cit. on p. 36).

[150] G. W. Forbes and M. A. Alonso, 'Measures of spread for periodic distributions and the associated uncertainty relations', American Journal of Physics **69**, 340 (2001) (cit. on p. 36).

[151] A. Marcolongo, P. Umari and S. Baroni, 'Microscopic theory and quantum simulation of atomic heat transport', Nature Physics **12**, 80 (2016) (cit. on pp. 36, 46, 48, 79, 90, 91, 187).

[152] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning* (MIT Press, Cambridge, 2006) (cit. on pp. 38, 41).

[153] C. Shannon, 'Communication in the Presence of Noise', Proceedings of the IRE **37**, 10 (1949) (cit. on p. 38).

[154] A. N. Tikhonov, A. V. Goncharsky, V. V. Stepanov and A. G. Yagola, *Numerical Methods for the Solution of Ill-Posed Problems* (Springer Netherlands, 1995) (cit. on p. 38).

[155] J. Behler, 'Constructing High-Dimensional Neural Network Potentials: A Tutorial Review', International Journal of Quantum Chemistry **115**, 1032 (2015) (cit. on p. 39).

[156] A. A. Peterson, R. Christensen and A. Khorshidi, 'Addressing uncertainty in atomistic machine learning', Physical Chemistry Chemical Physics **19**, 10978 (2017) (cit. on p. 39).

[157] A. R. Tan, S. Urata, S. Goldman, J. C. B. Dietschreit and R. Gómez-Bombarelli, *Single-model uncertainty quantification in neural network potentials does not consistently outperform model ensembles*, 2nd May 2023, eprint: `arXiv:2305.01754` (cit. on p. 39).

[158] G. Imbalzano, Y. Zhuang, V. Kapil, K. Rossi, E. A. Engel, F. Grasselli and M. Ceriotti, 'Uncertainty estimation for molecular dynamics and sampling', The Journal of Chemical Physics **154**, 074102 (2021) (cit. on p. 39).

[159] D. Schwalbe-Koda, A. R. Tan and R. Gómez-Bombarelli, 'Differentiable sampling of molecular geometries with uncertainty-based adversarial attacks', Nature Communications **12**, 5104 (2021) (cit. on p. 39).

[160] G. Palmer, S. Du, A. Politowicz, J. P. Emory, X. Yang, A. Gautam, G. Gupta, Z. Li, R. Jacobs and D. Morgan, 'Calibration after bootstrap for accurate uncertainty quantification in regression models', Nature Partner Journal Computational Materials **8**, 115 (2022) (cit. on p. 39).

[161] G. Csányi, T. Albaret, M. C. Payne and A. De Vita, '"Learn on the Fly": A Hybrid Classical and Quantum-Mechanical Molecular Dynamics Simulation', Physical Review Letters **93**, 175503 (2004) (cit. on p. 39).

[162] R. Jinnouchi, K. Miwa, F. Karsai, G. Kresse and R. Asahi, 'On-the-Fly Active Learning of Interatomic Potentials for Large-Scale Atomistic Simulations', Journal of Physical Chemistry Letters **11**, 6946 (2020) (cit. on p. 39).

[163] C. Verdi, F. Karsai, P. Liu, R. Jinnouchi and G. Kresse, 'Thermal transport and phase transitions of zirconia by on-the-fly machine-learned interatomic potentials', Nature Partner Journal Computational Materials **7**, 156 (2021) (cit. on pp. 39, 79, 101, 106, 108, 116, 117, 125, 129, 208).

[164] C. van der Oord, M. Sachs, D. P. Kovács, C. Ortner and G. Csányi, *Hyperactive Learning (HAL) for Data-Driven Interatomic Potentials*, 9th Oct. 2022, eprint: `arXiv:2210.04225` (cit. on p. 39).

[165] Y. Xie, J. Vandermause, S. Ramakers, N. H. Protik, A. Johansson and B. Kozinsky, *Uncertainty-aware molecular dynamics from Bayesian active learning: Phase Transformations and Thermal Transport in SiC*, 7th Mar. 2022, eprint: `arXiv:2203.03824` (cit. on p. 39).

[166] J. Carrete, H. Montes-Campos, R. Wanzenböck, E. Heid and G. K. H. Madsen, *Deep Ensembles vs. Committees for Uncertainty Estimation in Neural-Network Force Fields: Comparison and Application to Active Learning*, 17th Feb. 2023, eprint: arXiv:2302.08805 (cit. on p. 39).

[167] E. Prodan and W. Kohn, 'Nearsightedness of electronic matter', Proceedings of the National Academy of Sciences of the United States of America **102**, 11635 (2005) (cit. on p. 39).

[168] A. Grisafi and M. Ceriotti, 'Incorporating long-range physics in atomic-scale machine learning', The Journal of Chemical Physics **151**, 204105 (2019) (cit. on pp. 39, 132).

[169] H. Yu, L. Hong, S. Chen, X. Gong and H. Xiang, *Capturing long-range interaction with reciprocal space neural network*, 29th Nov. 2022, eprint: arXiv:2211.16684 (cit. on pp. 39, 132).

[170] H. E. Sauceda, L. E. Gálvez-González, S. Chmiela, L. O. Paz-Borbón, K.-R. Müller and A. Tkatchenko, 'BIGDML—Towards accurate quantum machine learning force fields for materials', Nature Communications **13**, 3733 (2022) (cit. on p. 39).

[171] V. L. Deringer, N. Bernstein, G. Csányi, C. Ben Mahmoud, M. Ceriotti, M. Wilson, D. A. Drabold and S. R. Elliott, 'Origins of structural and electronic transitions in disordered silicon', Nature **589**, 59 (2021) (cit. on p. 39).

[172] P. G. Polishchuk, T. I. Madzhidov and A. Varnek, 'Estimation of the size of drug-like chemical space based on GDB-17 data', Journal of Computer-Aided Molecular Design **27**, 675 (2013) (cit. on p. 40).

[173] E. V. Podryabinkin, E. V. Tikhonov, A. V. Shapeev and A. R. Oganov, 'Accelerating crystal structure prediction by machine-learning interatomic potentials with active learning', Physical Review B **99**, 064114 (2019) (cit. on p. 40).

[174] R. Ouyang, S. Curtarolo, E. Ahmetcik, M. Scheffler and L. M. Ghiringhelli, 'SISSO: A compressed-sensing method for identifying the best low-dimensional descriptor in an immensity of offered candidates', Physical Review Materials **2**, 083802 (2018) (cit. on p. 40).

[175] L. Foppa, T. A. R. Purcell, S. V. Levchenko, M. Scheffler and L. M. Ghiringhelli, 'Hierarchical Symbolic Regression for Identifying Key Physical Parameters Correlated with Bulk Properties of Perovskites', Physical Review Letters **129**, 055301 (2022) (cit. on p. 40).

[176] D. Schwalbe-Koda and R. Gómez-Bombarelli, 'Generative Models for Automatic Chemical Design', in *Machine Learning Meets Quantum Physics* (Springer, 4th June 2020), pp. 445–467 (cit. on p. 40).

[177] N. W. A. Gebauer, M. Gastegger, S. S. P. Hessmann, K.-R. Müller and K. T. Schütt, 'Inverse design of 3d molecular structures with conditional generative neural networks', Nature Communications **13**, 973 (2022) (cit. on p. 40).

[178] H. Türk, E. Landini, C. Kunkel, J. T. Margraf and K. Reuter, 'Assessing Deep Generative Models in Chemical Composition Space', Chemistry of Materials **34**, 9455 (2022) (cit. on p. 40).

[179] J. Westermayr, J. Gilkes, R. Barrett and R. J. Maurer, *High-throughput property-driven generative design of functional organic molecules*, 4th July 2022, eprint: arXiv:2207.01476 (cit. on p. 40).

[180] C. Bilodeau, W. Jin, T. Jaakkola, R. Barzilay and K. F. Jensen, 'Generative models for molecular discovery: Recent advances and challenges', Wiley Interdisciplinary Reviews: Computational Molecular Science **in press**, e1608 (2022) (cit. on p. 40).

[181] D. Menon and R. Ranganathan, 'A Generative Approach to Materials Discovery, Design, and Optimization', ACS Omega **7**, 25958 (2022) (cit. on p. 40).

[182] M. Rupp, 'Machine Learning for Quantum Mechanics in a Nutshell', International Journal of Quantum Chemistry **115**, 1058 (2015) (cit. on pp. 40, 41).

[183] T. Hofmann, B. Schölkopf and A. J. Smola, 'Kernel methods in machine learning', The Annals of Statistics **36**, 1171 (2008) (cit. on pp. 40, 41).

[184] N. Aronszajn, 'Theory of reproducing kernels', Transactions of the American Mathematical Society **68**, 337 (1950) (cit. on p. 40).

[185] G. Kimeldorf and G. Wahba, 'Some results on Tchebycheffian spline functions', Journal of Mathematical Analysis and Applications **33**, 82 (1971) (cit. on p. 40).

[186] B. Schölkopf, R. Herbrich and A. J. Smola, 'A Generalized Representer Theorem', Computational Learning Theory, 416 (2001) (cit. on p. 40).

[187] A. J. Smola, B. Schölkopf and K.-R. Müller, 'The connection between regularization operators and support vector kernels', Neural Networks **11**, 637 (1998) (cit. on p. 41).

[188] B. Schölkopf, A. Smola and K.-R. Müller, 'Nonlinear Component Analysis as a Kernel Eigenvalue Problem', Neural Computation **10**, 1299 (1998) (cit. on p. 41).

[189] K.-R. Müller, S. Mika, G. Ratsch, K. Tsuda and B. Schölkopf, 'An introduction to kernel-based learning algorithms', IEEE Transactions on Neural Networks **12**, 181 (2001) (cit. on p. 41).

[190] A. P. Bartók and G. Csányi, 'Gaussian Approximation Potentials: A Brief Tutorial Introduction', International Journal of Quantum Chemistry **116**, 1051 (2015) (cit. on pp. 41, 55).

[191] S. Mathias, 'A Kernel-based Learning Method for an Efficient Approximation of the High-Dimensional Born-Oppenheimer Potential Energy Hypersurface', MA thesis (Institute for Numerical Simulation, Mathematisch-Naturwissenschaftliche Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn, Germany, 2015) (cit. on p. 41).

[192] J. Görtler, R. Kehlbeck and O. Deussen, 'A Visual Exploration of Gaussian Processes', Distill **4**, e17 (2019) (cit. on p. 41).

[193] T. Bayes, 'An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S', Philosophical Transactions of the Royal Society of London **53**, 370 (1763) (cit. on p. 41).

[194] G. Cybenko, 'Approximation by superpositions of a sigmoidal function', Mathematics of Control, Signals, and Systems **2**, 303 (1989) (cit. on p. 42).

[195] G. Gripenberg, 'Approximation by neural networks with a bounded number of nodes at each level', Journal of Approximation Theory **122**, 260 (2003) (cit. on p. 42).

[196] S. Hochreiter and J. Schmidhuber, 'Long Short-Term Memory', Neural Computation **9**, 1735 (1997) (cit. on p. 42).

[197] R. T. Q. Chen, Y. Rubanova, J. Bettencourt and D. Duvenaud, *Neural Ordinary Differential Equations*, 19th June 2018, eprint: `arXiv:1806.07366` (cit. on pp. 42, 132).

[198] D. P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*, 22nd Dec. 2014, eprint: `arXiv:1412.6980` (cit. on pp. 42, 104, 120).

[199] A. Radul, A. Paszke, R. Frostig, M. Johnson and D. Maclaurin, *You Only Linearize Once: Tangents Transpose to Gradients*, 22nd Apr. 2022, eprint: `arXiv:2204.10923` (cit. on p. 43).

[200] N. F. Schmitz, K.-R. Müller and S. Chmiela, 'Algorithmic Differentiation for Automated Modeling of Machine Learned Force Fields', Journal of Physical Chemistry Letters **13**, 10183 (2022) (cit. on pp. 43, 130).

[201] B. J. Alder, D. M. Gass and T. E. Wainwright, 'Studies in Molecular Dynamics. VIII. The Transport Coefficients for a Hard-Sphere Fluid', The Journal of Chemical Physics **53**, 3813 (1970) (cit. on pp. 45, 89).

[202] A. Antidormi and L. Colombo, 'Lattice Thermal Boundary Resistance', in *Handbook of Materials Modeling: Applications: Current and Emerging Materials* (Springer, 2020), pp. 845–863 (cit. on p. 45).

[203] A. Evans, D. Clarke and C. Levi, 'The influence of oxides on the performance of advanced gas turbines', Journal of the European Ceramic Society **28**, 1405 (2008) (cit. on pp. 45, 128).

[204] G. J. Snyder and E. S. Toberer, 'Complex thermoelectric materials', Nature Materials **7**, 105 (2008) (cit. on p. 45).

[205] S. Stackhouse, L. Stixrude and B. B. Karki, 'Thermal Conductivity of Periclase (MgO) from First Principles', Physical Review Letters **104**, 208501 (2010) (cit. on p. 45).

[206] F. Müller-Plathe, 'A simple nonequilibrium molecular dynamics method for calculating the thermal conductivity', The Journal of Chemical Physics **106**, 6082 (1997) (cit. on p. 45).

[207] T. M. Gibbons, B. Kang, S. K. Estreicher and C. Carbogno, 'Thermal conductivity of Si nanostructures containing defects: Methodology, isotope effects, and phonon trapping', Physical Review B **84**, 035317 (2011) (cit. on p. 45).

[208] P. Debye, 'Zur Theorie der spezifischen Wärmen', Annalen der Physik **344**, 789 (1912) (cit. on p. 45).

[209] R. Peierls, 'Zur kinetischen Theorie der Wärmeleitung in Kristallen', Annalen der Physik **395**, 1055 (1929) (cit. on p. 45).

[210] D. A. Broido, M. Malorny, G. Birner, N. Mingo and D. A. Stewart, 'Intrinsic lattice thermal conductivity of semiconductors from first principles', Applied Physics Letters **91**, 231922 (2007) (cit. on pp. 45, 121, 123).

[211] T. Feng and X. Ruan, 'Quantum mechanical prediction of four-phonon scattering rates and reduced thermal conductivity of solids', Physical Review B **93**, 045202 (2016) (cit. on p. 45).

[212] L. Isaeva, G. Barbalinardo, D. Donadio and S. Baroni, 'Modeling heat transport in crystals and glasses from a unified lattice-dynamical approach', Nature Communications **10**, 3853 (2019) (cit. on pp. 45, 94, 186, 187).

[213] M. Simoncelli, N. Marzari and F. Mauri, 'Unified theory of thermal transport in crystals and glasses', Nature Physics **15**, 809 (2019) (cit. on p. 45).

[214] N. K. Ravichandran and D. Broido, 'Unified first-principles theory of thermal properties of insulators', Physical Review B **98**, 085205 (2018) (cit. on p. 46).

[215] L. Onsager, 'Reciprocal relations in irreversible processes. I.', Physical Review **37**, 405 (1931) (cit. on p. 46).

[216] L. Onsager, 'Reciprocal relations in irreversible processes. II.', Physical Review **38**, 2265 (1931) (cit. on pp. 46, 47).

[217] M. S. Green, 'Markoff random processes and the statistical mechanics of time-dependent phenomena', The Journal of Chemical Physics **20**, 1281 (1952) (cit. on pp. 46, 79).

[218] R. Kubo, 'Statistical-mechanical theory of irreversible processes. I. General theory and simple applications to magnetic and conduction problems', Journal of the Physical Society of Japan **12**, 570 (1957) (cit. on pp. 46, 79).

[219] R. Kubo, M. Yokota and S. Nakajima, 'Statistical-mechanical theory of irreversible processes. II. Response to thermal disturbance', Journal of the Physical Society of Japan **12**, 1203 (1957) (cit. on pp. 46, 79).

[220] A. Marcolongo, L. Ercole and S. Baroni, 'Gauge Fixing for Heat-Transport Simulations', Journal of Chemical Theory and Computation **16**, 3352 (2020) (cit. on pp. 46, 91).

[221] N. Chetty and R. M. Martin, 'First-principles energy density and its applications to selected polar surfaces', Physical Review B **45**, 6074 (1992) (cit. on pp. 46, 48, 91).

[222] H. B. Callen and T. A. Welton, 'Irreversibility and Generalized Noise', Physical Review **83**, 34 (1951) (cit. on p. 47).

[223] G. W. Ford and R. F. O'Connell, 'There is No Quantum Regression Theorem', Physical Review Letters **77**, 798 (1996) (cit. on p. 47).

[224] A. Marcolongo, 'Theory and ab initio simulation of atomic heat transport', PhD thesis (SISSA, 2014) (cit. on p. 47).

[225] S. Baroni, R. Bertossa, L. Ercole, F. Grasselli and A. Marcolongo, 'Heat Transport in Insulators from Ab Initio Green-Kubo Theory', in *Handbook of Materials Modeling: Applications: Current and Emerging Materials* (Springer, 2020), pp. 809–844 (cit. on p. 47).

[226] E. Helfand, 'Transport coefficients from dissipation in a canonical ensemble', Physical Review **119**, 1 (1960) (cit. on p. 48).

[227] S. Viscardy, J. Servantie and P. Gaspard, 'Transport and Helfand moments in the Lennard-Jones fluid. I. Shear viscosity', The Journal of Chemical Physics **126**, 184512 (2007) (cit. on p. 48).

[228] S. Viscardy, J. Servantie and P. Gaspard, 'Transport and Helfand moments in the Lennard-Jones fluid. II. Thermal conductivity', The Journal of Chemical Physics **126**, 184513 (2007) (cit. on p. 48).

[229] L. Ercole, A. Marcolongo, P. Umari and S. Baroni, 'Gauge invariance of thermal transport coefficients', Journal of Low Temperature Physics **185**, 79 (2016) (cit. on pp. 48, 91).

[230] R. E. Jones and K. K. Mandadapu, 'Adaptive Green-Kubo estimates of transport coefficients from molecular dynamics based on robust error analysis', The Journal of Chemical Physics **136**, 154102 (2012) (cit. on p. 49).

[231] M. Rupp, R. Ramakrishnan and O. A. von Lilienfeld, 'Machine Learning for Quantum Mechanical Properties of Atoms in Molecules', Journal of Physical Chemistry Letters **6**, 3309 (2015) (cit. on pp. 53, 58, 64, 65).

[232] J. Nigam, S. Pozdnyakov, G. Fraux and M. Ceriotti, 'Unified theory of atom-centered representations and message-passing machine-learning schemes', The Journal of Chemical Physics **156**, 204115 (2022) (cit. on pp. 53, 59, 64).

[233] I. Batatia, S. Batzner, D. P. Kovács, A. Musaelian, G. N. C. Simm, R. Drautz, C. Ortner, B. Kozinsky and G. Csányi, *The Design Space of E(3)-Equivariant Atom-Centered Interatomic Potentials*, 13th May 2022, eprint: `arXiv:2205.06643` (cit. on pp. 53, 64, 79, 82, 84).

[234] A. Glielmo, P. Sollich and A. De Vita, 'Accurate Interatomic Force Fields via Machine Learning with Covariant Kernels', Physical Review B **95**, 214302 (2017) (cit. on pp. 54, 66).

[235] A. Grisafi, D. M. Wilkins, G. Csányi and M. Ceriotti, 'Symmetry-Adapted Machine-Learning for Tensorial Properties of Atomistic Systems', Physical Review Letters **120**, 036002 (2018) (cit. on pp. 54, 64).

[236] M. Ceriotti, M. J. Willatt and G. Csányi, 'Machine Learning of Atomic-Scale Properties Based on Physical Principles', in *Handbook of Materials Modeling. Methods: Theory and Modeling* (Springer, 1st Sept. 2018) (cit. on pp. 54, 58, 64).

[237] R. Drautz, 'Atomic cluster expansion of scalar, vectorial, and tensorial properties including magnetism and charge transfer', Physical Review B **102**, 024104 (2020) (cit. on pp. 54, 55, 58, 63–65).

[238] B. Anderson, T.-S. Hy and R. Kondor, 'Cormorant: Covariant Molecular Neural Networks', in Advances in Neural Information Processing Systems 32 (NeurIPS 2019), Montréal, Canada, December 8–14 (2019), pp. 14537–14546 (cit. on pp. 54, 84).

[239] R. Kondor, Z. Li and S. Trivedi, 'Clebsch-Gordan Nets: a Fully Fourier Space Spherical Convolutional Neural Network', in Advances in Neural Information Processing Systems 31 (NeurIPS 2018), Montréal, Canada, December 2–8 (2018), pp. 10117–10126 (cit. on p. 54).

[240] N. Thomas, T. Smidt, S. Kearnes, L. Yang, L. Li, K. Kohlhoff and P. Riley, 'Tensor Field Networks: Rotation- and Translation-Equivariant Neural Networks for 3D Point Clouds', in NeurIPS 2018 Workshop on Machine Learning for Molecules and Materials, Montréal, Canada, December 8 (2018) (cit. on pp. 54, 77, 84).

[241] T. S. Hy, S. Trivedi, H. Pan, B. M. Anderson and R. Kondor, 'Covariant Compositional Networks for Learning Graphs', in *Proceedings of the International Workshop on Mining and Learning with Graphs (MLG 2019), Anchorage, Alaska, August 5* (5th Aug. 2019) (cit. on p. 54).

[242] S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt and B. Kozinsky, 'E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials', Nature Communications **13**, 1 (2022) (cit. on pp. 54, 59, 75, 77, 79, 82, 84, 87).

[243] J. E. Moussa, 'Comment on "Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning"', Physical Review Letters **109**, 059801 (2012) (cit. on pp. 55, 58, 63).

[244] O. A. von Lilienfeld, R. Ramakrishnan, M. Rupp and A. Knoll, 'Fourier series of atomic radial distribution functions: A molecular fingerprint for machine learning models of quantum chemical properties', International Journal of Quantum Chemistry **115**, 1084 (2015) (cit. on p. 55).

[245] S. N. Pozdnyakov, M. J. Willatt, A. P. Bartók, C. Ortner, G. Csányi and M. Ceriotti, 'Incompleteness of Atomic Structure Representations', Physical Review Letters **125**, 166001 (2020) (cit. on pp. 55, 66, 75).

[246] A. P. Bartók, R. Kondor and G. Csányi, 'On representing chemical environments', Physical Review B **87**, 184115 (2013) (cit. on pp. 55, 58, 61, 66).

[247] E. Kocer, J. K. Mason and H. Erturk, 'Continuous and optimally complete description of chemical environments using Spherical Bessel descriptors', AIP Advances **10**, 015021 (2020) (cit. on pp. 55, 58, 62).

[248] H. M. Le, S. Huynh and L. M. Raff, 'Molecular dissociation of hydrogen peroxide (HOOH) on a neural network *ab initio* potential surface with a new configuration sampling method involving gradient fitting', The Journal of Chemical Physics **131**, 014107 (2009) (cit. on p. 55).

[249] A. Sadeghi, S. A. Ghasemi, B. Schaefer, S. Mohr, M. A. Lill and S. Goedecker, 'Metrics for measuring distances in configuration spaces', The Journal of Chemical Physics **139**, 184118 (2013) (cit. on pp. 55, 58, 64).

[250] M. Welborn, L. Cheng and T. F. Miller III, 'Transferability in Machine Learning for Electronic Structure via the Molecular Orbital Basis', Journal of Chemical Theory and Computation **14**, 4772 (2018) (cit. on pp. 55, 58, 64).

[251] R. Ramakrishnan, P. O. Dral, M. Rupp and O. A. von Lilienfeld, 'Big Data Meets Quantum Chemistry Approximations: The Δ-Machine Learning Approach', Journal of Chemical Theory and Computation **11**, 2087 (2015) (cit. on pp. 55, 58, 68).

[252] G. Schmitz, I. H. Godtliebsen and O. Christiansen, 'Machine learning for potential energy surfaces: An extensive database and assessment of methods', The Journal of Chemical Physics **150**, 244113 (2019) (cit. on pp. 55, 67).

[253] S. A. Ghasemi, A. Hofstetter, S. Saha and S. Goedecker, 'Interatomic potentials for ionic systems with density functional accuracy based on charge densities obtained by a neural network', Physical Review B **92**, 045131 (2015) (cit. on p. 55).

[254] B. Nebgen, N. Lubbers, J. S. Smith, A. E. Sifain, A. Lokhov, O. Isayev, A. E. Roitberg, K. Barros and S. Tretiak, 'Transferable Dynamic Molecular Charge Assignment Using Deep Neural Networks', Journal of Chemical Theory and Computation **14**, 4687 (2018) (cit. on p. 55).

[255] S. Rostami, M. Amsler and S. A. Ghasemi, 'Optimized symmetry functions for machine-learning interatomic potentials of multicomponent systems', The Journal of Chemical Physics **149**, 124106 (2018) (cit. on pp. 55, 58, 61).

[256] I. S. Novikov and A. V. Shapeev, 'Improving accuracy of interatomic potentials: more physics or more data? A case study of silica', Materials Today Communications **18**, 74 (2018) (cit. on pp. 55, 58).

[257] O. T. Unke and M. Meuwly, 'PhysNet: A Neural Network for Predicting Energies, Forces, Dipole Moments, and Partial Charges', Journal of Chemical Theory and Computation **15**, 3678 (2019) (cit. on pp. 55, 79, 82).

[258] T. W. Ko, J. A. Finkler, S. Goedecker and J. Behler, 'General-Purpose Machine Learning Potentials Capturing Nonlocal Charge Transfer', Accounts of Chemical Research **54**, 808 (2021) (cit. on p. 55).

[259] B. Parsaeifard, D. S. De, J. A. Finkler and S. Goedecker, 'Fingerprint-Based Detection of Non-Local Effects in the Electronic Structure of a Simple Single Component Covalent System', Condensed Matter **6**, 9 (2021) (cit. on p. 55).

[260] J. Westermayr and P. Marquetand, 'Machine learning and excited-state molecular dynamics', Machine Learning: Science and Technology **1**, 043001 (2020) (cit. on p. 55).

[261] J. Westermayr and P. Marquetand, 'Deep learning for UV absorption spectra with SchNarc: First steps toward transferability in chemical compound space', The Journal of Chemical Physics **153**, 154112 (2020) (cit. on p. 55).

[262] J. Westermayr, M. Gastegger and P. Marquetand, 'Combining SchNet and SHARC: The SchNarc Machine Learning Approach for Excited-State Dynamics', Journal of Physical Chemistry Letters **11**, 3828 (2020) (cit. on p. 55).

[263] J. Behler, B. Delley, S. Lorenz, K. Reuter and M. Scheffler, 'Dissociation of O_2 at Al(111): The Role of Spin Selection Rules', Physical Review Letters **94**, 036104 (2005) (cit. on p. 55).

[264] J. Westermayr, F. A. Faber, A. S. Christensen, O. A. von Lilienfeld and P. Marquetand, 'Neural networks and kernel ridge regression for excited states dynamics of $CH_2NH_2^+$: From single-state to multi-state representations and multi-property machine learning models', Machine Learning: Science and Technology **1**, 025009 (2020) (cit. on p. 55).

[265] M. Gastegger, K. T. Schütt and K.-R. Müller, 'Machine learning of solvent effects on molecular spectra and reactions', Chemical Science **12**, 11473 (2021) (cit. on p. 55).

[266] A. S. Christensen, F. A. Faber and O. A. von Lilienfeld, 'Operators in quantum machine learning: Response properties in chemical space', The Journal of Chemical Physics **150**, 064105 (2019) (cit. on p. 55).

[267] A. Goscinski, G. Fraux, G. Imbalzano and M. Ceriotti, 'The role of feature space in atomistic learning', Machine Learning: Science and Technology **2**, 025028 (2021) (cit. on pp. 55, 67).

[268] R. Todeschini and V. Consonni, *Handbook of molecular descriptors, 2nd edition* (Wiley, 2009) (cit. on p. 56).

[269] R. Jinnouchi, F. Karsai, C. Verdi, R. Asahi and G. Kresse, 'Descriptors representing two- and three-body atomic distributions and their effects on the accuracy of machine-learned interatomic potentials', The Journal of Chemical Physics **152**, 234102 (2020) (cit. on pp. 56, 59, 65).

[270] A. Glielmo, C. Zeni, Á. Fekete and A. De Vita, 'Building Nonparametric n-Body Force Fields Using Gaussian Process Regression', in *Machine Learning Meets Quantum Physics* (Springer, 4th June 2020), pp. 67–98 (cit. on pp. 56, 65).

[271] J. Behler, 'Atom-centered symmetry functions for constructing high-dimensional neural network potentials', The Journal of Chemical Physics **134**, 074106 (2011) (cit. on pp. 58, 60).

[272] J. S. Smith, O. Isayev and A. E. Roitberg, 'ANI-1: An extensible neural network potential with DFT accuracy at force field computational cost', Chemical Science **8**, 3192 (2017) (cit. on pp. 58, 61).

[273] M. Gastegger, L. Schwiedrzik, M. Bittermann, F. Berzsenyi and P. Marquetand, 'wACSF—Weighted Atom-Centered Symmetry Functions as Descriptors in Machine Learning Potentials', The Journal of Chemical Physics **148**, 241709 (2018) (cit. on pp. 58, 61).

[274] N. Artrith, A. Urban and G. 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**, 241711 (2018) (cit. on pp. 58, 61).

[275] Available as part of the software `RuNNer` at `http://www.uni-goettingen.de/de/560580.html`, GPL license, per email request). (cit. on p. 58).

[276] L. Himanen, M. O. Jäger, E. V. Morooka, F. F. Canova, Y. S. Ranawat, D. Z. Gao, P. Rinke and A. S. Foster, 'DScribe: Library of descriptors for machine learning in materials science', Computer Physics Communications **247**, 106949 (2020) (cit. on pp. 58, 62, 65, 67, 75, 215).

[277] M. A. Wood and A. P. Thompson, 'Extending the accuracy of the SNAP interatomic potential form', The Journal of Chemical Physics **148**, 241721 (2018) (cit. on pp. 58, 62, 75).

[278] A. Seko, A. Togo and I. Tanaka, 'Group-theoretical high-order rotational invariants for structural representations: Application to linearized machine learning interatomic potential', Physical Review B **99**, 214108 (2019) (cit. on p. 58).

[279] A. Seko, 'Machine learning potentials for multicomponent systems: The Ti-Al binary system', Physical Review B **102**, 174104 (2020) (cit. on p. 58).

[280] Available as part of the software `LAMMPS` (large-scale atomic/molecular massively parallel simulator, `http://lammps.sandia.gov`, GPL license, publicly accessible). (cit. on p. 58).

[281] M. Rupp, A. Tkatchenko, K.-R. Müller and O. A. von Lilienfeld, 'Reply to the Comment by J.E. Moussa', Physical Review Letters **109**, 059802 (2012) (cit. on pp. 58, 63).

[282] J. Barker, J. Bulin, J. Hamaekers and S. Mathias, 'LC-GAP: Localized Coulomb Descriptors for the Gaussian Approximation Potential', in *Scientific Computing and Algorithms in Industrial Simulations* (Springer, 20th July 2017), pp. 25–42 (cit. on pp. 58, 63).

[283] Available as part of the software `qmmlpack` (quantum mechanics machine learning package) at `https://gitlab.com/qmml/qmmlpack`, Apache 2.0 license, publicly accessible. (cit. on p. 58).

[284] S. De, A. P. Bartók, G. Csányi and M. Ceriotti, 'Comparing molecules and solids across structural and alchemical space', Physical Chemistry Chemical Physics **18**, 13754 (2016) (cit. on pp. 58, 65).

[285] A. P. Bartók, S. De, C. Poelking, N. Bernstein, J. R. Kermode, G. Csányi and M. Ceriotti, 'Machine Learning Unifies the Modelling of Materials and Molecules', Science Advances **3**, e1701816 (2017) (cit. on p. 58).

[286] M. A. Caro, 'Optimizing many-body atomic descriptors for enhanced computational performance of machine learning based interatomic potentials', Physical Review B **100**, 024112 (2019) (cit. on pp. 58, 62, 65).

[287] R. Jinnouchi, F. Karsai and G. Kresse, 'On-the-fly machine learning force field generation: Application to melting points', Physical Review B **100**, 014105 (2019) (cit. on pp. 58, 59, 61, 129).

[288] E. Kocer, J. K. Mason and H. Erturk, 'A novel approach to describe chemical environments in high-dimensional neural network potentials', The Journal of Chemical Physics **150**, 154102 (2019) (cit. on pp. 58, 62).

[289] Available as part of the software `libAtoms` (`http://www.libatoms.org`, custom license, per webform request). (cit. on p. 58).

[290] L. Zhu, M. Amsler, T. Fuhrer, B. Schaefer, S. Faraji, S. Rostami, S. A. Ghasemi, A. Sadeghi, M. Grauzinyte, C. Wolverton and S. Goedecker, 'A fingerprint based metric for measuring similarities of crystalline structures', The Journal of Chemical Physics **144**, 034203 (2016) (cit. on pp. 58, 64).

[291] B. Parsaeifard, D. S. De, A. S. Christensen, F. A. Faber, E. Kocer, S. De, J. Behler, A. von Lilienfeld and S. Goedecker, 'An assessment of the structural resolution of various fingerprints commonly used in machine learning', Machine Learning: Science and Technology **2**, 015018 (2021) (cit. on pp. 58, 64, 66, 67).

[292] K. Hansen, F. Biegler, R. Ramakrishnan, W. Pronobis, O. A. von Lilienfeld, K.-R. Müller and A. Tkatchenko, 'Machine Learning Predictions of Molecular Properties: Accurate Many-Body Potentials and Nonlocality in Chemical Space', Journal of Physical Chemistry Letters **6**, 2326 (2015) (cit. on pp. 58, 63).

[293] B. Huang and O. A. von Lilienfeld, 'Communication: Understanding molecular representations in machine learning: the role of uniqueness and target similarity', The Journal of Chemical Physics **145**, 161102 (2016) (cit. on pp. 58, 63, 71).

[294] Available as part of the software `QML` (quantum machine learning, `https://www.qmlcode.org/`, MIT license, publicly accessible). (cit. on p. 58).

[295] M. Hirn, S. Mallat and N. Poilvert, 'Wavelet Scattering Regression of Quantum Chemical Energies', Multiscale Modeling and Simulation **15**, 827 (2017) (cit. on pp. 58, 63).

[296] M. Eickenberg, G. Exarchakis, M. Hirn and S. Mallat, 'Solid Harmonic Wavelet Scattering: Predicting Quantum Molecular Energy from Invariant Descriptors of 3D Electronic Densities', in Advances in Neural Information Processing Systems 30 (NIPS 2017), Los Angeles, California, December 4–9 (2017), pp. 6522–6531 (cit. on pp. 58, 63).

[297] X. Brumwell, P. Sinz, K. J. Kim, Y. Qi and M. Hirn, 'Steerable Wavelet Scattering for 3D Atomic Systems with Application to Li-Si Energy Prediction', in NeurIPS 2018 Workshop on Machine Learning for Molecules and Materials, Montréal, Canada, December 8 (2018) (cit. on pp. 58, 63).

[298] M. Eickenberg, G. Exarchakis, M. Hirn, S. Mallat and L. Thiry, 'Solid harmonic wavelet scattering for predictions of molecule properties', The Journal of Chemical Physics **148**, 241732 (2018) (cit. on pp. 58, 63).

[299] E. R. Homer, D. M. Hensley, C. W. Rosenbrock, A. H. Nguyen and G. L. W. Hart, 'Machine-Learning Informed Representations for Grain Boundary Structures', Frontiers in Materials **6**, 168 (2019) (cit. on pp. 58, 63).

[300] P. Sinz, M. W. Swift, X. Brumwell, J. Liu, K. J. Kim, Y. Qi and M. Hirn, 'Wavelet scattering networks for atomistic systems with extrapolation of material properties', The Journal of Chemical Physics **153**, 084109 (2020) (cit. on pp. 58, 63).

[301] M. Andreux, T. Angles, G. Exarchakis, R. Leonarduzzi, G. Rochette, L. Thiry, J. Zarka, S. Mallat, J. Andén, E. Belilovsky, J. Bruna, V. Lostanlen, M. Chaudhary, M. J. Hirn, E. Oyallon, S. Zhang, C. Cella and M. Eickenberg, 'Kymatio: scattering transforms in Python', Journal of Machine Learning Research **21**, 1 (2020) (cit. on p. 58).

[302] A. V. Shapeev, 'Moment Tensor Potentials: A Class of Systematically Improvable Interatomic Potentials', Multiscale Modeling and Simulation **14**, 1153 (2016) (cit. on pp. 58, 63, 66).

[303] E. V. Podryabinkin and A. V. Shapeev, 'Active learning of linearly parametrized interatomic potentials', Computational Materials Science **140**, 171 (2017) (cit. on p. 58).

[304] K. Gubaev, E. V. Podryabinkin and A. V. Shapeev, 'Machine learning of molecular properties: Locality and active learning', The Journal of Chemical Physics **148**, 241727 (2018) (cit. on p. 58).

[305] A. V. Shapeev, 'Applications of Machine Learning for Representing Interatomic Interactions', in *Computational Materials Discovery* (Royal Society of Chemistry, 2019), pp. 66–86 (cit. on p. 58).

[306] I. S. Novikov, K. Gubaev, E. V. Podryabinkin and A. V. Shapeev, 'The MLIP package: Moment Tensor Potentials with MPI and active learning', Machine Learning: Science and Technology **2**, 025002 (2021) (cit. on p. 58).

[307] H. Huo and M. Rupp, 'Unified representation of molecules and crystals for machine learning', Machine Learning: Science and Technology **3**, 045017 (2022) (cit. on p. 58).

[308] C. Nyshadham, M. Rupp, B. Bekker, A. V. Shapeev, T. Mueller, C. W. Rosenbrock, G. Csányi, D. W. Wingate and G. L. Hart, 'Machine-Learned Multi-System Surrogate Models for Materials Prediction', Nature Partner Journal Computational Materials **5**, 51 (2019) (cit. on pp. 58, 67, 68, 178).

[309] F. A. Faber, L. Hutchison, B. Huang, J. Gilmer, S. S. Schoenholz, G. E. Dahl, O. Vinyals, S. Kearnes, P. F. Riley and O. A. von Lilienfeld, 'Prediction errors of molecular machine learning models lower than hybrid DFT error', Journal of Chemical Theory and Computation **13**, 5255 (2017) (cit. on pp. 58, 63, 67, 178).

[310] Y.-H. Tang, D. Zhang and G. E. Karniadakis, 'An Atomistic Fingerprint Algorithm for Learning *Ab Initio* Molecular Force Fields', The Journal of Chemical Physics **148**, 034101 (2018) (cit. on pp. 58, 64).

[311] A reference implementation in Python can be found at `https://doi.org/10.5281/ZENODO.1054550`, CC BY-SA 4.0 license, publicly accessible. (cit. on p. 58).

[312] F. A. Faber, A. S. Christensen, B. Huang and O. A. von Lilienfeld, 'Alchemical and structural distribution based representation for universal quantum machine learning', The Journal of Chemical Physics **148**, 241717 (2018) (cit. on pp. 58, 63, 75, 178).

[313] A. S. Christensen, L. A. Bratholm, F. A. Faber and O. A. von Lilienfeld, 'FCHL revisited: Faster and more accurate quantum machine learning', The Journal of Chemical Physics **152**, 044107 (2020) (cit. on pp. 58, 63, 178).

[314] W. Pronobis, A. Tkatchenko and K.-R. Müller, 'Many-Body Descriptors for Predicting Molecular Properties with Machine Learning: Analysis of Pairwise and Three-Body Interactions in Molecules', Journal of Chemical Theory and Computation **14**, 2991 (2018) (cit. on pp. 58, 63, 75, 178).

[315] Pseudo-code is available as part of the supporting information at `http://pubs.acs.org/doi/abs/10.1021/acs.jctc.8b00110` (cit. on p. 58).

[316] L. Cheng, M. Welborn, A. S. Christensen and T. F. Miller III, 'A universal density matrix functional from molecular orbital-based machine learning: Transferability across organic molecules', The Journal of Chemical Physics **150**, 131103 (2019) (cit. on pp. 58, 64).

[317] T. Husch, J. Sun, L. Cheng, S. J. R. Lee and T. F. Miller III, 'Improved accuracy and transferability of molecular-orbital-based machine learning: Organics, transition-metal complexes, noncovalent interactions, and transition states', The Journal of Chemical Physics **154**, 064108 (2021) (cit. on p. 58).

[318] S. J. R. Lee, T. Husch, F. Ding and T. F. Miller III, 'Analytical gradients for molecular-orbital-based machine learning', The Journal of Chemical Physics **154**, 124120 (2021) (cit. on p. 58).

[319] R. Drautz, 'Atomic cluster expansion for accurate and transferable interatomic potentials', Physical Review B **99**, 249901 (2019) (cit. on pp. 58, 62, 63, 65).

[320] G. Dusson, M. Bachmayr, G. Csányi, R. Drautz, S. Etter, C. van der Oord and C. Ortner, 'Atomic cluster expansion: Completeness, efficiency and stability', Journal of Computational Physics **454**, 110946 (2022) (cit. on pp. 58, 59, 62, 63, 65, 66).

[321] Y. Lysogorskiy, C. van der Oord, A. Bochkarev, S. Menon, M. Rinaldi, T. Hammerschmidt, M. Mrovec, A. Thompson, G. Csányi, C. Ortner and R. Drautz, 'Performant implementation of the atomic cluster expansion (PACE) and application to copper and silicon', Nature Partner Journal Computational Materials **7**, 97 (2021) (cit. on p. 58).

[322] An implementation in Julia can be found at `https://github.com/ACEsuit/ACE.jl`, ASLv1 license, publicly accessible. (cit. on p. 58).

[323] J. Nigam, S. Pozdnyakov and M. Ceriotti, 'Recursive evaluation and iterative contraction of N-body equivariant features', The Journal of Chemical Physics **153**, 121101 (2020) (cit. on pp. 58, 63, 65).

[324] An implementation in Python can be found at `https://github.com/cosmo-epfl/nice`, MIT license, publicly accessible. (cit. on p. 58).

[325] V. Zaverkin and J. Kästner, 'Gaussian Moments as Physically Inspired Molecular Descriptors for Accurate and Scalable Machine Learning Potentials', Journal of Chemical Theory and Computation **16**, 5410 (2020) (cit. on pp. 58, 63).

[326] M. Uhrin, 'Through the eyes of a descriptor: Constructing complete, invertible descriptions of atomic environments', Physical Review B **104**, 144110 (2021) (cit. on p. 58).

[327] An implementation in Python can be found at `https://github.com/muhrin/milad`, GPLv3 license, publicly accessible. (cit. on p. 58).

[328] F. Musil, A. Grisafi, A. P. Bartók, C. Ortner, G. Csányi and M. Ceriotti, 'Physics-Inspired Structural Representations for Molecules and Materials', Chemical Reviews **121**, 9759 (2021) (cit. on pp. 57, 59).

[329] M. J. Elrod and R. J. Saykally, 'Many-Body Effects in Intermolecular Forces', Chemical Reviews **94**, 1975 (1994) (cit. on p. 58).

[330] T. E. Smidt, 'Euclidean Symmetry and Equivariance in Machine Learning', Trends in Chemistry **3**, 82 (2021) (cit. on pp. 59, 77).

[331] K. T. Schütt, O. T. Unke and M. Gastegger, 'Equivariant message passing for the prediction of tensorial properties and molecular spectra', in Proceedings of the 38th International Conference on Machine Learning (ICML 2021), virtual, July 18–24 (2021), pp. 9377–9388 (cit. on pp. 59, 77).

[332] K. T. Schütt, H. Glawe, F. Brockherde, A. Sanna, K.-R. Müller and E. K. Gross, 'How to represent crystal structures for machine learning: Towards fast prediction of electronic properties', Physical Review B **89**, 205118 (2014) (cit. on p. 61).

[333] The `DScribe` code contains a local MBTR example of this. See `https://github.com/SINGROUP/dscribe`. (cit. on pp. 63–65).

[334] J. Nigam, M. J. Willatt and M. Ceriotti, 'Equivariant representations for molecular Hamiltonians and N-center atomic-scale properties', The Journal of Chemical Physics **156**, 014115 (2022) (cit. on p. 64).

[335] I. Batatia, D. P. Kovács, G. N. C. Simm, C. Ortner and G. Csányi, 'MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields', in Advances in Neural Information Processing Systems 35 (NeurIPS 2022), New Orleans, Louisiana, USA, Nov 28–Dec 9 (2022), pp. 11423–11436 (cit. on pp. 64, 75, 77, 79, 82, 84, 130).

[336] Y. Zhang, C. Hu and B. Jiang, 'Embedded Atom Neural Network Potentials: Efficient and Accurate Machine Learning with a Physically Inspired Representation', Journal of Physical Chemistry Letters **10**, 4962 (2019) (cit. on p. 64).

[337] Z. Qiao, M. Welborn, A. Anandkumar, F. R. Manby and T. F. Miller III, 'OrbNet: Deep learning for quantum chemistry using symmetry-adapted atomic-orbital features', The Journal of Chemical Physics **153**, 124111 (2020) (cit. on p. 64).

[338] Y. Chen, L. Zhang, H. Wang and W. E, 'Ground State Energy Functional with Hartree-Fock Efficiency and Chemical Accuracy', The Journal of Physical Chemistry A **124**, 7155 (2020) (cit. on p. 64).

[339] Z. Li, J. R. Kermode and A. De Vita, 'Molecular Dynamics with On-The-Fly Machine Learning of Quantum Mechanical Forces', Physical Review Letters **114**, 096405 (2015) (cit. on pp. 64, 103).

[340] A. Glielmo, C. Zeni and A. D. Vita, 'Efficient non-parametric n-body force fields from machine learning', Physical Review B **97**, 184307 (2018) (cit. on pp. 64, 65).

[341] Y. Zuo, C. Chen, X. Li, Z. Deng, Y. Chen, J. Behler, G. Csányi, A. V. Shapeev, A. P. Thompson, M. A. Wood and S. P. Ong, 'Performance and Cost Assessment of Machine Learning Interatomic Potentials', The Journal of Physical Chemistry A **124**, 731 (2020) (cit. on pp. 67, 130).

[342] A. Stuke, M. Todorović, M. Rupp, C. Kunkel, K. Ghosh, L. Himanen and P. Rinke, 'Chemical diversity in molecular orbital energy predictions with kernel ridge regression', The Journal of Chemical Physics **150**, 204121 (2019) (cit. on p. 67).

[343] B. Onat, C. Ortner and J. R. Kermode, 'Sensitivity and dimensionality of atomic environment representations used for machine learning interatomic potentials', The Journal of Chemical Physics **153**, 144106 (2020) (cit. on p. 67).

[344] S. Käser, D. Koner, A. S. Christensen, O. A. von Lilienfeld and M. Meuwly, 'ML Models of Vibrating H_2CO: Comparing Reproducing Kernels, FCHL and PhysNet', The Journal of Physical Chemistry A **124**, 8853 (2020) (cit. on p. 67).

[345] M. O. J. Jäger, E. V. Morooka, F. Federici-Canova, L. Himanen and A. S. Foster, 'Machine learning hydrogen adsorption on nanoclusters through structural descriptors', Nature Partner Journal Computational Materials **4**, 37 (2018) (cit. on pp. 67, 75).

[346] R. Ramakrishnan, P. O. Dral, M. Rupp and O. A. von Lilienfeld, 'Quantum Chemistry Structures and Properties of 134 kilo Molecules', Scientific Data **1**, 140022 (2014) (cit. on p. 68).

[347] L. Ruddigkeit, R. van Deursen, L. C. Blum and J.-L. Reymond, 'Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17', Journal of Chemical Information and Modeling **52**, 2864 (2012) (cit. on p. 68).

[348] P. J. Stephens, F. J. Devlin, C. F. Chabalowski and M. J. Frisch, '*Ab initio* Calculation of Vibrational Absorption and Circular Dichroism Spectra Using Density Functional Force Fields', Journal of Physical Chemistry **98**, 11623 (1994) (cit. on p. 68).

[349] L. Vegard, 'Die Konstitution der Mischkristalle und die Raumfüllung der Atome', Zeitschrift für Physik **5**, 17 (1921) (cit. on p. 68).

[350] A. R. Denton and N. W. Ashcroft, 'Vegard's law', Physical Review A **43**, 3161 (1991) (cit. on p. 68).

[351] G. L. W. Hart and R. W. Forcade, 'Algorithm for generating derivative structures', Physical Review B **77**, 224115 (2008) (cit. on p. 68).

[352] P. Wisesa, K. A. McGill and T. Mueller, 'Efficient generation of generalized Monkhorst-Pack grids through the use of informatics', Physical Review B **93**, 155109 (2016) (cit. on p. 68).

[353] W. S. Morgan, J. J. Jorgensen, B. C. Hess and G. L. Hart, 'Efficiency of Generalized Regular k-point Grids', Computational Materials Science **153**, 424 (2018) (cit. on p. 68).

[354] C. Sutton, L. M. Ghiringhelli, T. Yamamoto, Y. Lysogorskiy, L. Blumenthal, T. Hammerschmidt, J. R. Golebiowski, X. Liu, A. Ziletti and M. Scheffler, 'Crowd-sourcing materials-science challenges with the NOMAD 2018 Kaggle competition', Nature Partner Journal Computational Materials **5**, 111 (2019) (cit. on pp. 68, 178).

[355] Nomad2018 Predicting Transparent Conductors. Predict the key properties of novel transparent semiconductors. Available at `https://www.kaggle.com/c/nomad2018-predict-transparent-conductors` (cit. on p. 68).

[356] V. Blum, R. Gehrke, F. Hanke, P. Havu, V. Havu, X. Ren, K. Reuter and M. Scheffler, 'Ab initio molecular simulations with numeric atom-centered orbitals', Computer Physics Communications **180**, 2175 (2009) (cit. on pp. 68, 103).

[357] J. S. Bergstra, D. Yamins and D. D. Cox, 'Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures', in Proceedings of the 30th International Conference on Machine Learning (ICML) (2013), p. 115 (cit. on p. 70).

[358] J. S. Bergstra, R. Bardenet, Y. Bengio and B. Kégl, 'Algorithms for hyper-parameter optimization', in Advances in Neural Information Processing Systems 24 (NIPS 2011), Granada, Spain, Dec 12–17 (2011), p. 2546 (cit. on p. 70).

[359] S. Amari, N. Fujita and S. Shinomoto, 'Four Types of Learning Curves', Neural Computation **4**, 605 (1992) (cit. on p. 71).

[360] K.-R. Müller, M. Finke, N. Murata, K. Schulten and S. Amari, 'A Numerical Study on Learning Curves in Stochastic Multilayer Feedforward Networks', Neural Computation **8**, 1085 (1996) (cit. on p. 71).

[361] V. L. Deringer, A. P. Bartók, N. Bernstein, D. M. Wilkins, M. Ceriotti and G. Csányi, 'Gaussian Process Regression for Materials and Molecules', Chemical Reviews **121**, 10073 (2021) (cit. on p. 73).

[362] A. Samanta, 'Representing local atomic environment using descriptors based on local correlations', The Journal of Chemical Physics **149**, 244102 (2018) (cit. on p. 75).

[363] S. J. Honrao, S. R. Xie and R. G. Hennig, 'Augmenting machine learning of energy landscapes with local structural information', Journal of Applied Physics **128**, 085101 (2020) (cit. on p. 75).

[364] M. M. Bronstein, J. Bruna, T. Cohen and P. Veličković, *Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges*, 27th Apr. 2021, eprint: `arXiv:2104.13478` (cit. on p. 77).

[365] J. Klicpera, J. Groß and S. Günnemann, 'Directional Message Passing for Molecular Graphs', in Proceedings of the Eighth International Conference on Learning Representations (ICLR 2020), Addis Ababa, Ethiopia, April 26–May 1 (2020) (cit. on pp. 77, 79, 82, 84).

[366] D. Levesque, L. Verlet and J. Kürkijarvi, 'Computer "Experiments" on Classical Fluids. IV. Transport Properties and Time-Correlation Functions of the Lennard-Jones Liquid near Its Triple Point', Physical Review A **7**, 1690 (1973) (cit. on pp. 79, 89).

[367] A. J. Ladd, B. Moran and W. G. Hoover, 'Lattice thermal conductivity: A comparison of molecular dynamics and anharmonic lattice dynamics', Physical Review B **34**, 5058 (1986) (cit. on pp. 79, 89, 94, 186).

[368] S. G. Volz and G. Chen, 'Molecular dynamics simulation of thermal conductivity of silicon nanowires', Applied Physics Letters **75**, 2056 (1999) (cit. on pp. 79, 89).

[369] Y. Chen, 'Local stress and heat flux in atomistic systems involving three-body forces', The Journal of Chemical Physics **124**, 054113 (2006) (cit. on pp. 79, 89).

[370] A. Guajardo-Cuéllar, D. B. Go and M. Sen, 'Evaluation of heat current formulations for equilibrium molecular dynamics calculations of thermal conductivity', The Journal of Chemical Physics **132**, 104111 (2010) (cit. on pp. 79, 89).

[371] Z. Fan, L. F. C. Pereira, H.-Q. Wang, J.-C. Zheng, D. Donadio and A. Harju, 'Force and heat current formulas for many-body potentials in molecular dynamics simulations with applications to thermal conductivity calculations', Physical Review B **92**, 094301 (2015) (cit. on pp. 79, 86, 89, 94, 96, 121, 183, 184, 203).

[372] G. C. Sosso, D. Donadio, S. Caravati, J. Behler and M. Bernasconi, 'Thermal transport in phase-change materials from atomistic simulations', Physical Review B **86**, 104301 (2012) (cit. on p. 79).

[373] C. Mangold, S. Chen, G. Barbalinardo, J. Behler, P. Pochet, K. Termentzidis, Y. Han, L. Chaput, D. Lacroix and D. Donadio, 'Transferability of neural network potentials for varying stoichiometry: Phonons and thermal conductivity of Mn_xGe_y compounds', Journal of Applied Physics **127**, 244901 (2020) (cit. on p. 79).

[374] P. Korotaev, I. Novoselov, A. Yanilkin and A. Shapeev, 'Accessing thermal conductivity of complex compounds by machine learning interatomic potentials', Physical Review B **100**, 144308 (2019) (cit. on p. 79).

[375] H. Liu, X. Qian, H. Bao, C. Zhao and X. Gu, 'High-temperature phonon transport properties of SnSe from machine-learning interatomic potential', Journal of Physics: Condensed Matter **33**, 405401 (2021) (cit. on pp. 79, 119–121, 125).

[376] R. Li, E. Lee and T. Luo, 'A unified deep neural network potential capable of predicting thermal conductivity of silicon in different phases', Materials Today Physics **12**, 100181 (2020) (cit. on pp. 79, 121, 123).

[377] R. Li, Z. Liu, A. Rohskopf, K. Gordiz, A. Henry, E. Lee and T. Luo, 'A deep neural network interatomic potential for studying thermal conductivity of β-Ga_2O_3', Applied Physics Letters **117**, 152102 (2020) (cit. on p. 79).

[378] X. Qian, S. Peng, X. Li, Y. Wei and R. Yang, 'Thermal conductivity modeling using machine learning potentials: application to crystalline and amorphous silicon', Materials Today Physics **10**, 100140 (2019) (cit. on pp. 79, 121–123).

[379] J. Klicpera, F. Becker and S. Günnemann, 'GemNet: Universal Directional Graph Neural Networks for Molecules', in Advances in Neural Information Processing Systems 34 (NeurIPS 2021), virtual, December 6–14 (2021), pp. 6790–6802 (cit. on pp. 79, 82).

[380] A. Bochkarev, Y. Lysogorskiy, C. Ortner, G. Csányi and R. Drautz, 'Multilayer atomic cluster expansion for semilocal interactions', Physical Review Research **4**, L042019 (2022) (cit. on pp. 79, 82).

[381] R. J. Hardy, 'Energy-flux operator for a lattice', Physical Review **132**, 168 (1963) (cit. on pp. 80, 89, 93, 183).

[382] M. Rupp, E. Proschak and G. Schneider, 'Kernel Approach to Molecular Similarity Based on Iterative Graph Similarity', Journal of Chemical Information and Modeling **47**, 2280 (2007) (cit. on p. 81).

[383] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, C. Gulcehre, F. Song, A. Ballard, J. Gilmer, G. Dahl, A. Vaswani, K. Allen, C. Nash, V. Langston, C. Dyer, N. Heess, D. Wierstra, P. Kohli, M. Botvinick, O. Vinyals, Y. Li and R. Pascanu, *Relational inductive biases, deep learning, and graph networks*, 4th June 2018, eprint: `arXiv:1806.01261` (cit. on p. 81).

[384] T. Xie and J. C. Grossman, 'Crystal Graph Convolutional Neural Networks for an Accurate and Interpretable Prediction of Material Properties', Physical Review Letters **120**, 145301 (2018) (cit. on p. 82).

[385] T. Morawietz, V. Sharma and J. Behler, 'A neural network potential-energy surface for the water dimer based on environment-dependent atomic energies and charges', The Journal of Chemical Physics **136**, 064103 (2012) (cit. on pp. 83, 132).

[386] K. T. Schütt, O. T. Unke and M. Gastegger, *Equivariant message passing for the prediction of tensorial properties and molecular spectra*, 5th Feb. 2021, eprint: `arXiv:2102.03150` (cit. on pp. 84, 178).

[387] A. P. Thompson, S. J. Plimpton and W. Mattson, 'General formulation of pressure and stress tensor for arbitrary many-body interaction potentials under periodic boundary conditions', The Journal of Chemical Physics **131**, 154107 (2009) (cit. on pp. 85–87).

[388] A. Kinaci, J. B. Haskins and T. Çağın, 'On calculation of thermal conductivity from Einstein relation in equilibrium molecular dynamics', The Journal of Chemical Physics **137**, 014106 (2012) (cit. on p. 85).

[389] N. C. Admal and E. B. Tadmor, 'Stress and heat flux for arbitrary multibody potentials: A unified framework', The Journal of Chemical Physics **134**, 184106 (2011) (cit. on p. 86).

[390] M. J. Louwerse and E. J. Baerends, 'Calculation of pressure in case of periodic boundary conditions', Chemical Physics Letters **421**, 138 (2006) (cit. on pp. 86, 87).

[391] K. T. Schütt, P. Kessel, M. Gastegger, K. A. Nicoli, A. Tkatchenko and K.-R. Müller, 'SchNetPack: A Deep Learning Toolbox For Atomistic Systems', Journal of Chemical Theory and Computation **15**, 448 (2019) (cit. on pp. 87, 103, 125, 216).

[392] The `schnetpack` software library is publicly available at `https://github.com/atomistic-machine-learning/schnetpack/` under the MIT license. (cit. on p. 87).

[393] The `nequip` software library is publicly available at `https://github.com/mir-group/nequip` under the MIT license. (cit. on pp. 87, 130).

[394] The `mace` software library is publicly available at `https://github.com/ACEsuit/mace` under the MIT license. (cit. on p. 87).

[395] The `mace-jax` software library is publicly available at `https://github.com/ACEsuit/mace-jax/` under the MIT license. (cit. on p. 87).

[396] J. H. Irving and J. G. Kirkwood, 'The Statistical Mechanical Theory of Transport Processes. IV. The Equations of Hydrodynamics', The Journal of Chemical Physics **18**, 817 (1950) (cit. on pp. 89, 92, 183).

[397] W. Noll, 'Die Herleitung der Grundgleichungen der Thermomechanik der Kontinua aus der statistischen Mechanik', Journal of Rational Mechanics and Analysis **4**, 627 (1955) (cit. on pp. 89, 91, 92, 183, 184).

[398] R. J. Hardy, 'Formulas for determining local properties in molecular-dynamics simulations: Shock waves', The Journal of Chemical Physics **76**, 622 (1982) (cit. on pp. 89, 90, 183).

[399] A. Murdoch and D. Bedeaux, 'On the physical interpretation of fields in continuum mechanics', International Journal of Engineering Science **31**, 1345 (1993) (cit. on p. 89).

[400] N. C. Admal and E. B. Tadmor, 'A Unified Interpretation of Stress in Molecular Systems', Journal of Elasticity **100**, 63 (2010) (cit. on p. 89).

[401] Y. Fu and J.-H. Song, 'Heat flux expressions that satisfy the conservation laws in atomistic system involving multibody potentials', Journal of Computational Physics **294**, 191 (2015) (cit. on pp. 89, 91).

[402] P. Boone, H. Babaei and C. E. Wilmer, 'Heat Flux for Many-Body Interactions: Corrections to LAMMPS', Journal of Chemical Theory and Computation **15**, 5579 (2019) (cit. on pp. 89, 115).

[403] D. Torii, T. Nakano and T. Ohara, 'Contribution of inter- and intramolecular energy transfers to heat conduction in liquids', The Journal of Chemical Physics **128**, 044504 (2008) (cit. on p. 90).

[404] D. Surblys, H. Matsubara, G. Kikugawa and T. Ohara, 'Application of atomic stress to compute heat flux via molecular dynamics for systems with many-body interactions', Physical Review E **99**, 051301 (2019) (cit. on pp. 90, 91, 115).

[405] Y. Chen and A. Diaz, 'Local momentum and heat fluxes in transient transport processes and inhomogeneous systems', Physical Review E **94**, 053309 (2016) (cit. on p. 91).

[406] P. C. Howell, 'Comparison of molecular dynamics methods and interatomic potentials for calculating the thermal conductivity of silicon', The Journal of Chemical Physics **137**, 224111 (2012) (cit. on pp. 91, 121, 123).

[407] L. Ercole, 'Ab Initio Simulation of Heat Transport in Silica Glass', PhD thesis (Scuola Internazionale Superiore di Studi Avanzati, 2018) (cit. on p. 94).

[408] A. McGaughey and M. Kaviany, 'Phonon Transport in Molecular Dynamics Simulations: Formulation and Thermal Conductivity Prediction', Advances in Heat Transfer **39**, 169 (2006) (cit. on p. 94).

[409] N. Galamba, C. A. Nieto de Castro and J. F. Ely, 'Thermal conductivity of molten alkali halides from equilibrium molecular dynamics simulations', The Journal of Chemical Physics **120**, 8676 (2004) (cit. on pp. 94, 129).

[410] The `glp` software library is publicly available at `https://github.com/sirmarcel/glp/` under the MIT license. (cit. on pp. 98, 130, 216).

[411] F. Knoop, T. A. R. Purcell, M. Scheffler and C. Carbogno, 'Anharmonicity in Thermal Insulators: An Analysis from First Principles', Physical Review Letters **130**, 236301 (2023) (cit. on pp. 101, 119–121, 123, 125).

[412] P. Pegolo, S. Baroni and F. Grasselli, 'Temperature- and vacancy-concentration-dependence of heat transport in Li3ClO from multi-method numerical simulations', Nature Partner Journal Computational Materials **8**, 1 (2022) (cit. on p. 101).

[413] F. Grasselli, L. Stixrude and S. Baroni, 'Heat and charge transport in H2O at ice-giant conditions from ab initio molecular dynamics simulations', Nature Communications **11**, 1 (2020) (cit. on p. 101).

[414] S. Fabris, A. T. Paxton and M. W. Finnis, 'Free energy and molecular dynamics calculations for the cubic-tetragonal phase transition in zirconia', Physical Review B **63**, 094101 (2001) (cit. on p. 101).

[415] C. Carbogno, C. G. Levi, C. G. Van de Walle and M. Scheffler, 'Ferroelastic switching of doped zirconia: Modeling and understanding from first principles', Physical Review B **90**, 144109 (2014) (cit. on pp. 101, 128).

[416] L. Momenzadeh, I. V. Belova and G. E. Murch, 'Prediction of the lattice thermal conductivity of zircon and the cubic and monoclinic phases of zirconia by molecular dynamics simulation', Computational Materials Science **176**, 109522 (2020) (cit. on pp. 101, 117).

[417] F. Knoop, T. A. R. Purcell, M. Scheffler and C. Carbogno, 'Anharmonicity measure for materials', Physical Review Materials **4**, 083809 (2020) (cit. on pp. 102, 170, 189).

[418] *Dataset PRL_2017_aiGK on the NOMAD Repository*, `https://doi.org/10.17172/NOMAD/2017.04.13-1` (cit. on p. 104).

[419] C. Draxl and M. Scheffler, 'The NOMAD laboratory: from data sharing to artificial intelligence', Journal of Physics: Materials **2**, 036001 (2019), eprint: `notavailable` (cit. on p. 104).

[420] E. H. Kisi and C. Howard, 'Crystal Structures of Zirconia Phases and their Inter-Relation', Key Engineering Materials **153–154**, 1 (1998) (cit. on pp. 106, 108, 116, 117, 208).

[421] M. Ceriotti, J. More and D. E. Manolopoulos, 'i-PI: A Python interface for ab initio path integral molecular dynamics simulations', Computer Physics Communications **185**, 1019 (2014) (cit. on p. 106).

[422] K. Tolborg and A. Walsh, 'Exploring the High-Temperature Stabilization of Cubic Zirconia from Anharmonic Lattice Dynamics', Crystal Growth & Design **23**, 3314 (2023) (cit. on pp. 107, 128).

[423] R. Patil and E. Subbarao, 'Axial thermal expansion of ZrO_2 and HfO_2 in the range room temperature to 1400°C', Journal of Applied Crystallography **2**, 281 (1969) (cit. on pp. 116, 117, 208).

[424] S. Raghavan, H. Wang, R. B. Dinwiddie, W. D. Porter and M. J. Mayo, 'The effect of grain size, porosity and yttria content on the thermal conductivity of nanocrystalline zirconia', Scripta Materialia **39**, 1119 (1998) (cit. on pp. 117, 125, 210).

[425] J.-F. Bisson, D. Fournier, M. Poulain, O. Lavigne and R. Mévrel, 'Thermal Conductivity of Yttria-Zirconia Single Crystals, Determined with Spatially Resolved Infrared Thermography', Journal of the American Ceramic Society **83**, 1993 (2000) (cit. on pp. 117, 125, 210).

[426] R. Mévrel, J.-C. Laizet, A. Azzopardi, B. Leclercq, M. Poulain, O. Lavigne and D. Demange, 'Thermal diffusivity and conductivity of $Zr_{1-x}Y_xO_{2-x/2}$ ($x = 0$, 0.084 and 0.179) single crystals', Journal of the European Ceramic Society **24**, 3081 (2004) (cit. on pp. 117, 125, 210).

[427] The `mlff` software library is publicly available at `https://github.com/thorben-frank/mlff/`. (cit. on pp. 119, 216).

[428] A. J. Minnich, J. A. Johnson, A. J. Schmidt, K. Esfarjani, M. S. Dresselhaus, K. A. Nelson and G. Chen, 'Thermal Conductivity Spectroscopy Technique to Measure Phonon Mean Free Paths', Physical Review Letters **107**, 095901 (2011) (cit. on p. 119).

[429] Y. He, I. Savić, D. Donadio and G. Galli, 'Lattice thermal conductivity of semiconducting bulk materials: atomistic simulations', Physical Chemistry Chemical Physics **14**, 16209 (2012) (cit. on pp. 119, 121, 123).

[430] *Dataset aiGK_for_anharmonic_solids on the NOMAD Repository*, `https://doi.org/10.17172/NOMAD/2021.11.11-1` (cit. on p. 119).

[431] J. Brorsson, A. Hashemi, Z. Fan, E. Fransson, F. Eriksson, T. Ala-Nissila, A. V. Krasheninnikov, H.-P. Komsa and P. Erhart, 'Efficient Calculation of the Lattice Thermal Conductivity by Atomistic Simulations with Ab Initio Accuracy', Advanced Theory and Simulations **5**, 2100217 (2021) (cit. on pp. 119–121, 125).

[432] P.-C. Wei, S. Bhattacharya, J. He, S. Neeleshwar, R. Podila, Y. Y. Chen and A. M. Rao, 'The intrinsic thermal conductivity of SnSe', Nature **539**, E1 (2016) (cit. on pp. 119–121, 125).

[433] L.-D. Zhao, S.-H. Lo, Y. Zhang, H. Sun, G. Tan, C. Uher, C. Wolverton, V. P. Dravid and M. G. Kanatzidis, 'Ultralow thermal conductivity and high thermoelectric figure of merit in SnSe crystals', Nature **508**, 373 (2014) (cit. on p. 120).

[434] J. J. Plata, P. Nath, D. Usanmaz, J. Carrete, C. Toher, M. de Jong, M. Asta, M. Fornari, M. B. Nardelli and S. Curtarolo, 'An efficient and accurate framework for calculating lattice thermal conductivity of solids: AFLOW—AAPL Automatic Anharmonic Phonon Library', Nature Partner Journal Computational Materials **3**, 1 (2017) (cit. on pp. 121, 123).

[435] H. Dong, Z. Fan, L. Shi, A. Harju and T. Ala-Nissila, 'Equivalence of the equilibrium and the nonequilibrium molecular dynamics methods for thermal conductivity calculations: From bulk to nanowire silicon', Physical Review B **97**, 094305 (2018) (cit. on p. 121).

[436] H. R. Shanks, P. D. Maycock, P. H. Sidles and G. C. Danielson, 'Thermal Conductivity of Silicon from 300 to 1400°K', Physical Review **130**, 1743 (1963) (cit. on pp. 121, 123).

[437] C. J. Glassbrenner and G. A. Slack, 'Thermal Conductivity of Silicon and Germanium from 3°K to the Melting Point', Physical Review **134**, A1058 (1964) (cit. on pp. 121, 123).

[438] W. Fulkerson, J. P. Moore, R. K. Williams, R. S. Graves and D. L. McElroy, 'Thermal Conductivity, Electrical Resistivity, and Seebeck Coefficient of Silicon from 100 to 1300°K', Physical Review **167**, 765 (1968) (cit. on pp. 121, 123).

[439] C. Y. Ho, R. W. Powell and P. E. Liley, 'Thermal Conductivity of the Elements', Journal of Physical and Chemical Reference Data **1**, 279 (1972) (cit. on pp. 121, 123).

[440] T. Ruf, R. Henn, M. Asen-Palmer, E. Gmelin, M. Cardona, H.-J. Pohl, G. Devyatych and P. Sennikov, 'Thermal conductivity of isotopically enriched silicon', Solid State Communications **115**, 243 (2000) (cit. on p. 121).

[441] T. Ruf, R. Henn, M. Asen-Palmer, E. Gmelin, M. Cardona, H.-J. Pohl, G. Devyatych and P. Sennikov, 'Erratum to "Thermal conductivity of isotopically enriched silicon" [Solid State Communications, 115 (2000) 243-247]', Solid State Communications **127**, 257 (2003) (cit. on pp. 121, 123).

[442] R. Kremer, K. Graf, M. Cardona, G. Devyatykh, A. Gusev, A. Gibin, A. Inyushkin, A. Taldenkov and H.-J. Pohl, 'Thermal conductivity of isotopically enriched 28Si: revisited', Solid State Communications **131**, 499 (2004) (cit. on pp. 121, 123).

[443] P. K. Schelling, S. R. Phillpot and P. Keblinski, 'Comparison of atomic-level simulation methods for computing thermal conductivity', Physical Review B **65**, 144306 (2002) (cit. on p. 121).

[444] T. Tadano, Y. Gohda and S. Tsuneyuki, 'Anharmonic force constants extracted from first-principles molecular dynamics: applications to heat transfer simulations', Journal of Physics: Condensed Matter **26**, 225402 (2014) (cit. on p. 121).

[445] N. Shulumba, O. Hellman and A. J. Minnich, 'Intrinsic localized mode and low thermal conductivity of PbSe', Physical Review B **95**, 014302 (2017) (cit. on p. 122).

[446] D. Sangalli and A. Debernardi, 'Exchange-correlation effects in the monoclinic to tetragonal phase stabilization of yttrium-doped ZrO2: A first-principles approach', Physical Review B **84**, 214113 (2011) (cit. on p. 128).

[447] G. Stapper, M. Bernasconi, N. Nicoloso and M. Parrinello, 'Ab initiostudy of structural and electronic properties of yttria-stabilized cubic zirconia', Physical Review B **59**, 797 (1999) (cit. on p. 128).

[448] S. Fabris, 'A stabilization mechanism of zirconia based on oxygen vacancies only', Acta Materialia **50**, 5171 (2002) (cit. on p. 128).

[449] R. Jinnouchi, J. Lahnsteiner, F. Karsai, G. Kresse and M. Bokdam, 'Phase Transitions of Hybrid Perovskites Simulated by Machine-Learning Force Fields Trained on the Fly with Bayesian Inference', Physical Review Letters **122**, 225701 (2019) (cit. on p. 129).

[450] M. Blondel, Q. Berthet, M. Cuturi, R. Frostig, S. Hoyer, F. Llinares-López, F. Pedregosa and J.-P. Vert, *Efficient and Modular Implicit Differentiation*, 31st May 2021, eprint: arXiv:2105.15183 (cit. on pp. 129, 132).

[451] D. M. Anstine and O. Isayev, 'Machine Learning Interatomic Potentials and Long-Range Physics', The Journal of Physical Chemistry A, 10.1021/acs.jpca.2c06778 (2023) (cit. on p. 130).

[452] J. Topping, F. Di Giovanni, B. P. Chamberlain, X. Dong and M. M. Bronstein, *Understanding over-squashing and bottlenecks on graphs via curvature*, 29th Nov. 2021, eprint: arXiv:2111.14522 (cit. on p. 131).

[453] Q. Li, Z. Han and X.-M. Wu, *Deeper Insights into Graph Convolutional Networks for Semi-Supervised Learning*, 22nd Jan. 2018, eprint: arXiv:1801.07606 (cit. on p. 131).

[454] C. Cai and Y. Wang, *A Note on Over-Smoothing for Graph Neural Networks*, 23rd June 2020, eprint: arXiv:2006.13318 (cit. on p. 131).

[455] C. Ortner and Y. Wang, *A framework for a generalisation analysis of machine-learned interatomic potentials*, 12th Sept. 2022, eprint: arXiv:2209.05366 (cit. on p. 131).

[456] J. Behler and G. Csányi, 'Machine learning potentials for extended systems: a perspective', The European Physical Journal B **94**, 1 (2021) (cit. on p. 131).

[457] A. Ziletti, D. Kumar, M. Scheffler and L. M. Ghiringhelli, 'Insightful classification of crystal structures using deep learning', Nature Communications **9**, 2775 (2018) (cit. on p. 132).

[458] A. Kosmala, J. Gasteiger, N. Gao and S. Günnemann, *Ewald-based Long-Range Message Passing for Molecular Graphs*, 8th Mar. 2023, eprint: arXiv:2303.04791 (cit. on p. 132).

[459] T. K. Rusch, B. P. Chamberlain, J. Rowbottom, S. Mishra and M. M. Bronstein, *Graph-Coupled Oscillator Networks*, 4th Feb. 2022, eprint: arXiv:2202.02296 (cit. on p. 132).

[460] A. Tkatchenko, R. A. DiStasio, R. Car and M. Scheffler, 'Accurate and Efficient Method for Many-Body van der Waals Interactions', Physical Review Letters **108**, 236402 (2012) (cit. on p. 132).

[461] M. Gori, P. Kurian and A. Tkatchenko, *Second Quantization Approach to Many-Body Dispersion Interactions*, 23rd May 2022, eprint: arXiv:2205.11549 (cit. on p. 132).

[462] M. Sipser, *Introduction to the Theory of Computation*, Third Edition (Cengage Learning, 2013) (cit. on p. 170).

[463] D. Chicco, M. J. Warrens and G. Jurman, 'The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation', PeerJ Computer Science **7**, e623 (2021) (cit. on p. 171).

[464] R. C. Tolman, 'The Measurable Quantities of Physics', Physical Review **9**, 237 (1917) (cit. on p. 173).

[465] G. N. Hatsopoulos and J. H. Keenan, *Principles of General Thermodynamics* (Wiley, New York, 1965) (cit. on p. 173).

[466] H. Jung, S. Stocker, C. Kunkel, H. Oberhofer, B. Han, K. Reuter and J. T. Margraf, 'Size-Extensive Molecular Machine Learning with Global Representations', ChemSystemsChem **2**, e1900052 (2020) (cit. on p. 173).

[467] C. Poelking, F. A. Faber and B. Cheng, 'BenchML: an extensible pipelining framework for benchmarking representations of materials and molecules at scale', Machine Learning: Science and Technology **3**, 040501 (2022) (cit. on pp. 173, 178).

[468] S. Stocker, J. Gasteiger, F. Becker, S. Günnemann and J. T. Margraf, 'How robust are modern graph neural network potentials in long and hot molecular dynamics simulations?', Machine Learning: Science and Technology **3**, 045010 (2022) (cit. on p. 173).

[469] M. J. Willatt, F. Musil and M. Ceriotti, 'Feature Optimization for Atomistic Machine Learning Yields A Data-Driven Construction of the Periodic Table of the Elements', Physical Chemistry Chemical Physics **20**, 29661 (2018) (cit. on p. 178).

[470] K. T. Schütt, M. Gastegger, A. Tkatchenko and K.-R. Müller, 'Quantum-chemical insights from interpretable atomistic neural networks', in *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* (Springer, 10th Sept. 2019), pp. 311–330 (cit. on p. 178).

[471] J. Gasteiger, S. Giri, J. T. Margraf and S. Günnemann, *Fast and Uncertainty-Aware Directional Message Passing for Non-Equilibrium Molecules*, 28th Nov. 2020, eprint: `arXiv:2011.14115` (cit. on p. 178).

[472] A. Musaelian, S. Batzner, A. Johansson, L. Sun, C. J. Owen, M. Kornbluth and B. Kozinsky, *Learning Local Equivariant Representations for Large-Scale Atomistic Dynamics*, 11th Apr. 2022, eprint: `arXiv:2204.05249` (cit. on p. 178).

[473] A. J. McGaughey and M. Kaviany, 'Thermal conductivity decomposition and analysis using molecular dynamics simulations. Part I. Lennard-Jones argon', International Journal of Heat and Mass Transfer **47**, 1783 (2004) (cit. on pp. 180, 203).

[474] R. B. Lehoucq and A. Von Lilienfeld-Toal, 'Translation of Walter Noll's "Derivation of the Fundamental Equations of Continuum Thermodynamics from Statistical Mechanics"', Journal of Elasticity **100**, 5 (2010) (cit. on p. 183).

[475] J. Che, T. Çağın, W. Deng and W. A. Goddard, 'Thermal conductivity of diamond and related materials from molecular dynamics simulations', The Journal of Chemical Physics **113**, 6888 (2000) (cit. on p. 203).

[476] L. Ercole, A. Marcolongo and S. Baroni, 'Accurate thermal conductivities from optimally short molecular dynamics simulations', Scientific Reports **7**, 1 (2017) (cit. on p. 203).

[477] D. Hasselman, L. Johnson, L. Bentsen, R. Syed, H. Lee and M. Swain, 'Thermal Diffusivity of Conductivity of Dense Polycrystalline ZrO2 Ceramics: A Survey', American Ceramic Society Bulletin **66**, 799 (1987) (cit. on p. 210).

[478] G. E. Youngblood, R. W. Rice and R. P. Ingel, 'Thermal Diffusivity of Partially and Fully Stabilized (Yttria) Zirconia Single Crystals', Journal of the American Ceramic Society **71**, 255 (1988) (cit. on p. 210).

[479] The `cmlkit` Python package is publicly available at `https://marcel.science/cmlkit` under an MIT license and as part of the Nomad Analytics Toolkit (`https://analytics-toolkit.nomad-coe.eu/`). (cit. on p. 215).

[480] The `RuNNer` software (`https://www.uni-goettingen.de/de/560580.html`, GPL license) is available from its author Jörg Behler (`joerg.behler@uni-goettingen.de`) on request. (cit. on p. 215).

[481] The `quippy` software is publicly available at `http://libatoms.github.io/QUIP/` under the GNU General Public license 2. (cit. on p. 215).

[482] The `qmmlpack` (quantum mechanics machine learning package) library is publicly available at `https://gitlab.com/qmml/qmmlpack` under the Apache-2.0 license. (cit. on p. 215).

[483] The `gkx` software library is publicly available at `https://github.com/sirmarcel/gkx/` under the MIT license. (cit. on p. 216).

Glossary

ACE atomic cluster expansion. 58, 62–64, 66

AD automatic differentiation. 12–14, 30, 37, 42, 43, 79–81, 84–87, 90, 95–97, 99, 127–130, 132, 179, 188, 204

ADAM adaptive moment estimation. 42, 104

AE absolute error. 105, 171

aiGK ab initio Green-Kubo. 46, 79, 90, 102, 119, 121, 122, 125, 129

aiMD ab initio molecular dynamics. 11, 13, 27, 101, 103, 105, 119–121, 123, 125, 189, 215

ASE atomic simulation environment. 88, 179, 180

B3LYP Becke 3-parameter Lee-Yang-Parr. 68

BCC body-centred cubic. 68

BO Born-Oppenheimer. 11, 12, 17, 19, 20, 25, 27, 34, 46, 79, 81

BoB bag of bonds. 58, 63, 64

BS bispectrum. 58, 62, 64

BTE Boltzmann transport equation. 45, 121, 123, 125

CM Coulomb matrix. 58, 63–66

DECAF density-encoded canonically-aligned fingerprint. 58, 64

DFT density-functional theory. 11, 12, 17, 19, 22–24, 27, 31, 39, 46, 68, 79, 90, 91, 101, 102, 120, 121, 123, 128, 185

DM21 DeepMind 21. 23

EAM embedded atom method. 28

FCC face-centred cubic. 68

FCHL Faber-Christensen-Huang-von Lilienfeld. 58, 63, 64, 178

FCI full configuration-interaction. 21

FCP force constant potential. 119–121

FF forcefield. 11, 12, 14, 17, 25, 27–31, 39, 45, 46, 59, 79, 81–83, 89, 101, 115, 117, 121, 123, 125, 132, 173, 184

GAP Gaussian approximation potential. 121–123

GDB-17 'generated database 17'. 68

GGA generalised gradient approximation. 23

GK Green-Kubo. 12–14, 17, 45, 46, 48–50, 79, 89–91, 95, 98, 99, 101, 107, 109, 115, 117, 119–123, 125, 127, 128, 130, 203, 215

GLP graph-based machine-learning potential. 13, 79–84, 86, 87, 90, 94–96, 98, 99, 101, 115–117, 119, 120, 125, 127, 128, 130, 173, 180, 184, 185, 204, 209, 216

GM Gaussian moments. 58, 63, 64

GPR Gaussian process regression. 39, 41

GPU graphics processing unit. 12, 120, 122, 216

HCP hexagonal close-packed. 68

HDAD histograms of distances, angles, and dihedral angles. 58, 63, 64, 178

HE Helfand-Einstein. 48, 93

HF Hartree-Fock. 21–23, 64

HFACF heat flux autocorrelation function. 47, 49, 110, 112, 113, 115, 170, 198–201, 203

HP hyperparameter. 38, 41, 42, 51–53, 60–64, 66, 67, 69–71, 73, 76, 77, 104, 106, 127, 173, 215

HPC high-performance computing. 11, 27, 49, 71

i.i.d. independent and identically distributed. 37, 69, 172

IDMBR inverse-distance many-body representation. 58, 63, 64, 178

IV internal vectors. 64

JVP Jacobian-vector product. 43, 85, 97, 188

KRR kernel ridge regression. 37, 38, 40–42, 69–71, 77

LAMMPS Large-scale Atomic/Molecular Massively Parallel Simulator. 89, 90, 130

LDA local-density approximation. 23

MACE multi-layer atomic cluster expansion. 75, 87

MAE mean absolute error. 71, 72, 74, 76, 88, 171, 172, 175–178, 180, 211, 213

MAPE mean absolute percentage error. 88, 172, 180

maxAE maximum absolute error. 88, 171, 180, 211, 213

maxAPE maximum absolute percentage error. 88, 172, 180

MBTR many-body tensor representation. 57–59, 62–64, 67, 69, 73, 75–77, 127, 178, 215

MD molecular dynamics. 11, 12, 17, 25–28, 30, 33, 34, 39, 45, 46, 49, 67, 79, 82, 85, 91, 106, 107, 116, 120, 122, 126, 128–130, 170, 171, 183, 184, 215, 216

MIC minimum image convention. 35, 38, 57, 61, 79, 82, 83, 85, 87, 95, 96, 130, 171, 179, 180, 184

MILAD moment invariants local atomic descriptors. 58, 64

ML machine learning. 11, 12, 17, 23, 30, 37–40, 51, 53–55, 59, 68, 69, 77, 81, 127, 130, 131, 173, 215

MLIP machine-learning interatomic potential. 11–14, 17, 25, 30, 31, 39, 40, 42, 43, 46, 55, 67, 77, 79–83, 87, 90, 91, 99, 101–103, 106–109, 116, 117, 119, 121, 123, 125, 127–131, 173, 185, 192–195, 208, 215

MOB molecular orbital basis. 58, 64

MPNN message-passing neural network. 12–14, 42, 64, 83, 84, 97, 99, 101, 103, 104, 106, 119, 128, 131, 132, 191

MTP moment tensor potential. 58, 63, 64, 66, 119, 178

NICE N-body iterative contraction of equivariants. 58, 63, 64

NN neural network. 12, 21, 37, 38, 41, 42, 51, 54, 59, 75, 77, 83, 132

OMF overlap matrix fingerprint. 58, 64

PAW projector-augmented wave. 68

PBE Perdew, Burke, and Ernzerhof. 23, 68, 120, 123

PES potential energy surface. 11, 12, 17, 19, 20, 25, 27, 30, 34, 39, 46, 58, 79, 81, 85, 101, 102, 111, 123, 126, 185

QM quantum mechanics. 19, 20, 37

RBF radial basis function. 41

RKHS reproducing kernel Hilbert space. 40, 41

rMAE relative mean absolute error. 176, 177

RMSE root mean squared error. 68, 70–72, 74, 76, 172, 176–178, 211, 213

RMSLE root mean square logarithmic error. 68

rRMSE relative root mean squared error. 71, 72, 74, 76, 172, 178

s.c. simulation cell. 32, 93

SCF self-consistent field. 21, 22, 27, 64

SE squared exponential. 41

SF symmetry function. 57–61, 63, 64, 67, 69, 76, 77, 127, 215

SGD stochastic gradient descent. 42

SISSO sure independence screening and sparsifying operator. 40

SNAP Spectral Neighbor Analysis Potential. 62

SOAP smooth overlap of atomic positions. 57–59, 61, 62, 64–67, 69, 70, 73, 77, 101, 127, 178, 215

SW Stillinger-Weber. 28, 121, 123, 184

TPE tree-structured Parzen estimator. 70, 71, 215

TPU tensor processing unit. 12, 31

VASP Vienna Ab initio Simulation Package. 129

VDOS vibrational density of states. 105, 106, 112, 119, 120, 130, 191, 211

VJP vector-Jacobian product. 43, 97

WST wavelet scattering transform. 58, 63, 64

Appendices

Sometimes what goes without saying is best said anyway.

– Ian M. Banks, *The Hydrogen Sonata*

Appendix A

Conventions and Notation

A.1 Conventions

- Latin indices $i, j, k, l, ...$: Such indices are used for atoms.
- Latin indices $p, q, r, s, ...$: Used for electrons.
- Greek indices $\alpha, \beta, ...$: Used for cartesian directions and to enumerate lattice vectors.

A.2 Notation

General Notation

- Calligraphic symbols like $\mathcal{A}$ are used for sets and other collections of objects. Occasionally, also used to denote abstract objects, as opposed to fixed-size array-like representations.
- $\boldsymbol{x}$ indicates a column vector, uppercase $\boldsymbol{M}$ a matrix. In most cases, these objects are restricted to $\mathbf{R}^n$ and $\mathbf{R}^{n\times m}$ respectively, but may occasionally be used for more general vector- and matrix-like objects, such as elements of and operators acting in a Hilbert space.
- $\boldsymbol{M}_{ij}$: Entry in row i and column j of matrix. Can be used to define a matrix via its matrix elements.
- $\cdot$ used for bold symbols indicates a matrix-matrix, vector-matrix, or matrix-vector product; otherwise, it is used to emphasise multiplication, for instance in multi-line expressions. We also use it to signify a placeholder in another expression.
- $\times$ used to indicate the cross product between vectors, or multiples in the context of supercells and k-point grids.
- $[\, a \mid b \,]^\top$ defines a column vector with entries given by a evaluated over b. If the a are column vectors, the result is a matrix where the a define the *rows*.
- $[\, a \mid b \,]$ same as above, but the result is a row vector, or a matrix where a define the columns.
- $O(f(n))$ indicates that the computational cost $c(n)$ (or other quantity) scales with n such that $\exists n_0, k > 0 : c(n) < kf(n) \forall n > n_0$. If

$f(n) = n$, for instance, the cost $c(n)$ is said to be asymptotically linear. [462] In the case of the number of atoms N, the limit is taken at *constant density*, and assuming homogeniety, such that the size of atomic neighbourhoods remains approximately constant with increasing N.

- $|\cdot|$ denotes the absolute value for scalars, the vector norm $\sqrt{\boldsymbol{x}\cdot\boldsymbol{x}}$ for vectors, and the cardinality for sets. Used in other contexts, it denotes a general norm, with the particular type of norm indicated by a subscript.
- $\mathbf{R}$: The real numbers.
- $\mathbf{Z}$: Integers, including negatives and zero.
- $\mathbf{1}$: Matrix with diagonal 1.

Symbols Related to Green-Kubo Method

- $\boldsymbol{\kappa}$: Thermal conductivity tensor. (Scalar: $\kappa = \mathrm{tr}(\boldsymbol{\kappa})/3$).
- $\boldsymbol{C}(t)$: HFACF at t. (Scalar: C.)
- $\boldsymbol{\kappa}(\tau)$: Integral of HFACF at upper integration limit τ. (Scalar: $\kappa(\tau)$.)
- t_0: MD simulation duration.
- n: Number of trajectories.
- T_0: Total simulation time nt_0.
- σ^{A}: Anharmonicity score defined in reference [417].

Atomistic Systems

- N: Number of atoms in a given system.
- $\mathcal{A}$: Abstract notation for an atom.
- $\mathcal{M} = \{\, \mathcal{A}_i \mid i = 1...N \,\}$: Abstract notation for a system.
- $\boldsymbol{P}_i$: General property of atom i.
- Z_i: Atomic number, i.e. the charge of the given nucleus. An example for an atomic property, and the only one explicitly featured in this thesis.
- $\mathcal{Z}$: All N charges for the atoms in a system.
- $\mathcal{B}$: See below; the lattice vectors for a periodic system.
- $\mathcal{R}$: All N positions of the system. In a periodic system, in most cases taken to be $\mathcal{R}_{\mathrm{sc}}$.
- $(\mathcal{R}, \mathcal{Z}, \mathcal{B})$: The main properties of a system relevant to this thesis, as these define the input for electronic structure methods and representations.

- $\mathcal{P} = \{ \boldsymbol{p}_i \mid i = 1...N \}$: All N momenta in a system.
- Γ^t: Phase-space point $(\mathcal{R}(t), \mathcal{P}(t))$ at time t. In the context of MD, this is taken to indicate that the system has evolved from some starting time $t = 0$ to the given t. Used as alternative to the 3-tuple defined above in cases where different position/momentum configurations of an otherwise identical system occur.
- $\boldsymbol{r}_{ij} = \boldsymbol{r}_j - \boldsymbol{r}_i$: Atom-pair vector. Also called 'displacement' vector. In this definition $\boldsymbol{r}_{ij}$ points from i to j. r_{ij} is defined as $|\boldsymbol{r}_{ij}|$. r is used as a general distance if no particular atoms are specified.
- $|\boldsymbol{r}_{ij}^{\text{mic}}| = \min_{\boldsymbol{n}\in\mathbf{Z}^3} |\boldsymbol{r}_{j\boldsymbol{n}} - \boldsymbol{r}_i| \quad i, j \in \mathcal{R}_{\text{sc}}$: Minimum image convention. Note that $\boldsymbol{r}_{ij}^{\text{mic}}$ indicates $\boldsymbol{r}_{j\boldsymbol{n}} - \boldsymbol{r}_i$ with $\boldsymbol{n}$ chosen such that $|\boldsymbol{r}_{ij}^{\text{mic}}|$ is the MIC distance. In this thesis, the MIC is assumed to be implemented such that derivatives match this definition; this does not hold, for instance, if it is implemented via fractional coordinates.

Periodic Systems

Notation for periodic systems is defined in detail in section 2.3. Selected symbols:

- $\mathcal{R}_{\text{sc}}$: Positions in simulation cell.
- $\mathcal{R}_{\text{all}}$: Positions in bulk.
- $\mathcal{B}$: Collection of lattice vectors $\boldsymbol{b}$.
- $\mathcal{A}$: Collection of reciprocal lattice vectors $\boldsymbol{a}$.
- $r_{\text{c}}^{\text{max}}$: Maximum cutoff radius for a given simulation cell that avoids double interactions. In orthorhombic systems, half the smallest lattice constant. For an implementation of the MIC that uses fractional coodinates, neighbours beyond this distance may be miscounted.

A.3 Loss Functions and Error Metrics

An overview of error metrics can be found in [463], which also supplies the definitions collected here.

In this section, y_i denote labels and f_i the corresponding model prediction. $e_i{=}f_i - y_i$ are the *residuals* from which most loss functions are constructed. We take n to be the number of labels and predictions.

Absolute Error The absolute error (AE) is simply

$$\text{AE} = |e_i| \,; \tag{A.1}$$

it is used as the basis for aggregate losses. The mean absolute error (MAE) is

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |e_i| \,. \tag{A.2}$$

The maximum absolute error (maxAE) is

$$\text{maxMAE} = \max_{i=1...n} |e_i| \,. \tag{A.3}$$

Root Mean Squared Error The root mean squared error (RMSE) is

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} e_i^2} \tag{A.4}$$

The relative root mean squared error (rRMSE) is then

$$\text{rRMSE} = \frac{\text{RMSE}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})^2}} = \frac{\text{RMSE}}{\sigma_y} = \frac{\text{RMSE}}{\text{RMSE}^*}, \tag{A.5}$$

where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ is the mean of the ground truth labels and σ_y is their standard deviation. The rRMSE can be seen as RMSE relative to the RMSE of a baseline model RMSE^* that always predicts the mean of the labels.[1]

We note that the RMSE is an upper bound to the MAE; it is more sensitive to outliers, and a large difference between the two metrics can be used to detect the present of predictions with large error.

[1] Note, however, that the baseline model would more naturally predict the mean of the *training* data. As long as the assumption of i.i.d. data holds, either should yield similar results.

Percentage Errors In some situations, it is useful to use error metrics that can be compared across datasets, where the overall scale of labels can differ significantly. One approach to this is to use percentage errors. The absolute percentage error (APE) is defined as

$$\text{APE} = \left|\frac{y_i - f_i}{f_i}\right|. \tag{A.6}$$

The mean absolute percentage error (MAPE) and maximum absolute percentage error (maxAPE) are the mean and maximum over the dataset, respectively. Note that the division by predictions f_i implies that percentage errors suffer from numerical issues for small values, and diverge at 0.

Coefficient of Determination Another scale-independent metric is the coefficient of determination. Denoting $\bar{y}$ as the mean of labels,

$$R^2 = \left(1 - \frac{\sum_{i=1}^{n}(y_i - f_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}\right). \tag{A.7}$$

In this thesis R^2 is always implemented in this form, and usually given in %.

Appendix B
Review and Benchmark of Representations of Molecules and Materials

This appendix collects additions to chapter 3. Additional information, for instance HP search spaces and details of the employed methods, can be found in the supplementary information to reference [44].

Here, only additions, and figures and tables required for discussion, are included.

B.1 Extensive and Intensive Properties

A property whose magnitude is independent of the size of an object is called *intensive*, whereas a property that is additive in size is called *extensive* [464, 465]. For example, internal energy is an extensive property, band gap energy an intensive one.

For finite systems such as molecules, a property p is extensive if for any two *non-interacting* systems A and B, $p(A+B) = p(A) + p(B)$, [466] and intensive if $p(A) = p(A+A)$. For periodic systems such as bulk crystals, we take A and B to be supercells of the same unit cell. In this minimal sense, total and atomisation energy of atomistic systems are extensive.

However, energies are not additive for general changes in a system, such as changes in atomic position, and addition or removal of atoms. Once interactions are included, extensivity is no longer ensured. Nevertheless, with respect to the requirements in section 3.1, ML models for energies should be size-extensive in the (minimal) sense above. For global representations, this can be achieved via normalisation in conjunction with the linear kernel [466], whereas local representations as described in section 3.1, FFs of the form in equation (2.2.10), which includes the GLPs in section 4.1, automatically satisfy this requirement.

B.2 Related Benchmarks

In addition to the studies in table 3.3.1, Poelking et al. [467] introduce a benchmarking framework for representations, including automatic HP optimisation, and report learning curves for a large number of datasets and representations, including `qm9` and `ba10`. A number of recent studies on MLIPs have also appeared. Fu et al. [126] and Stocker et al. [468] probe the limits of MLIPs in practice, finding that stability of simulations is not always correlated with force error.

B.3 Additional Figures and Tables

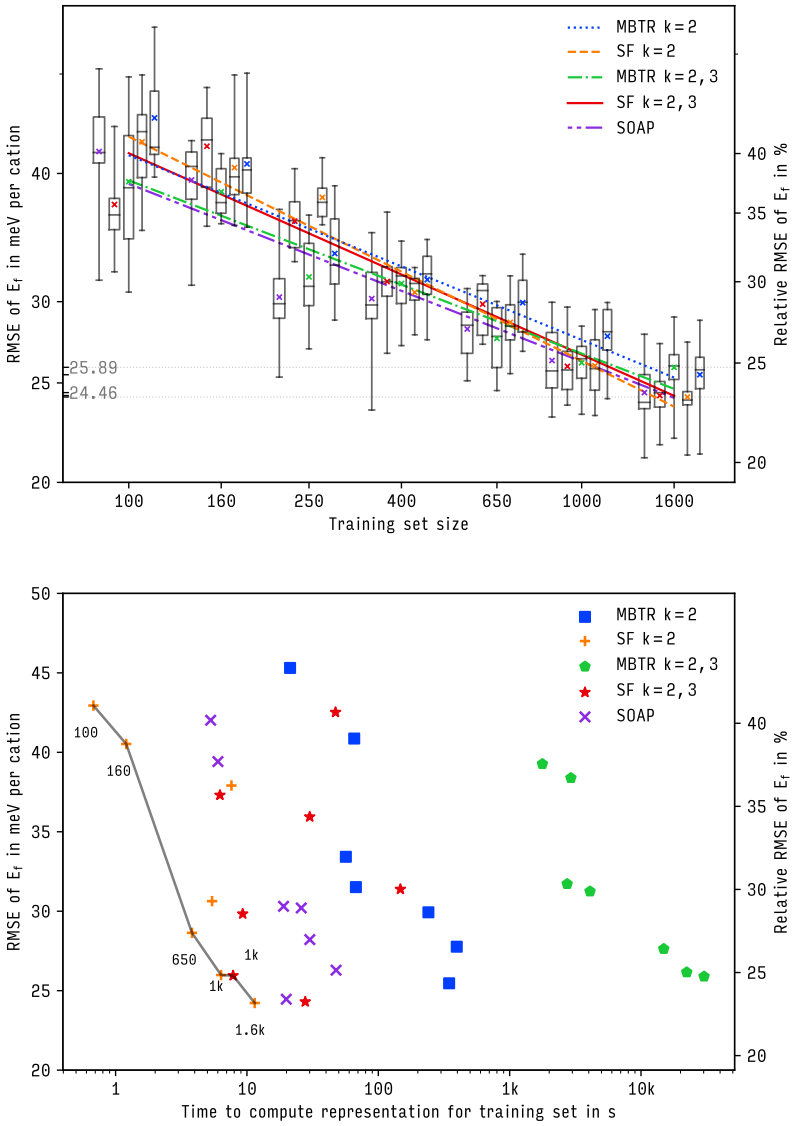


Figure B.1: Learning curve (top, see figure 3.3.2) and error/runtime plot (bottom, see figure 3.3.3) for selected representations on dataset `nmd18u`.

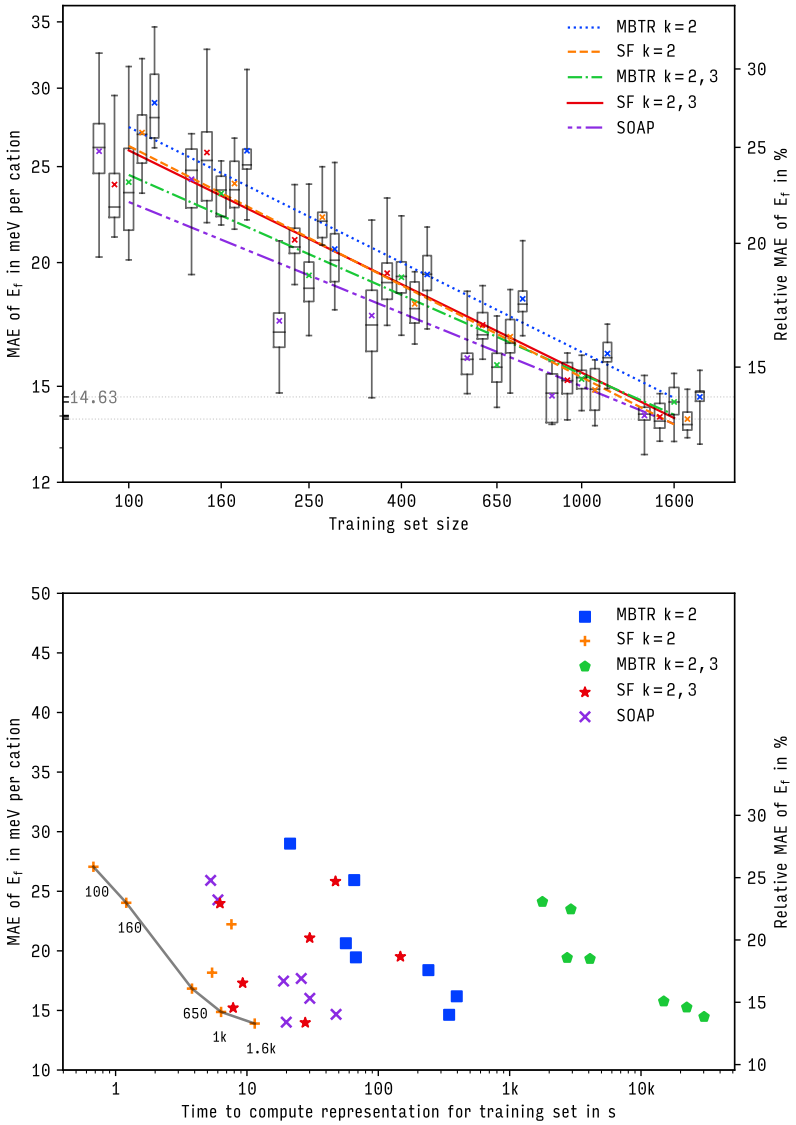


Figure B.2: Equivalent to figure B.1, but using MAE.

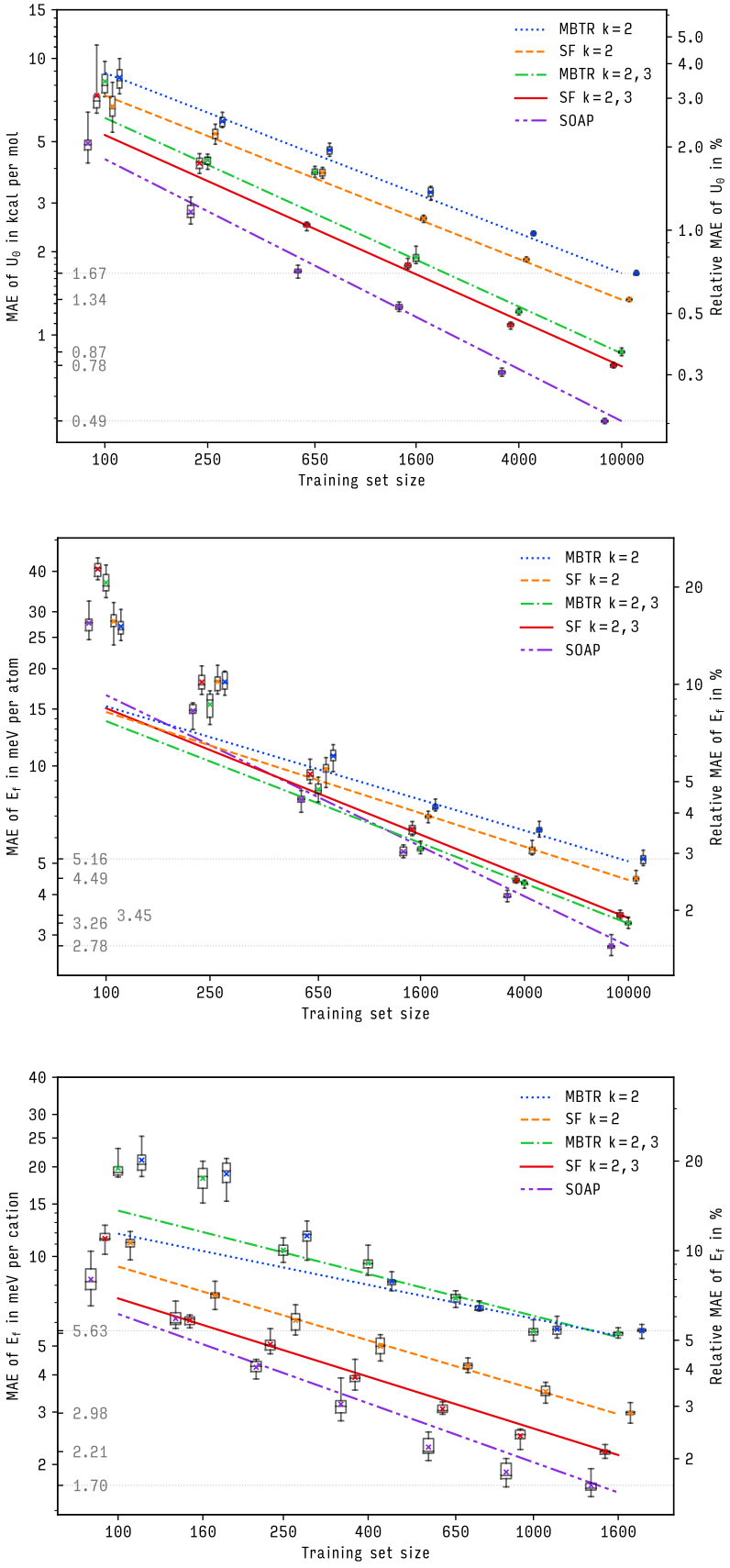
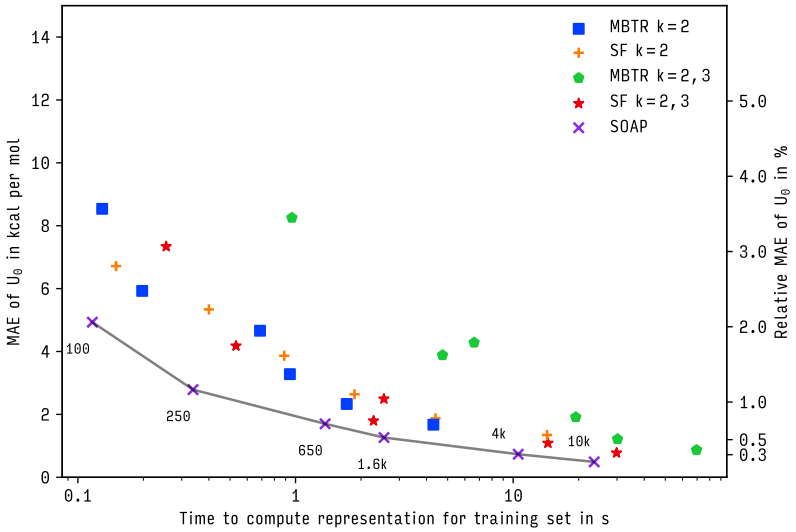
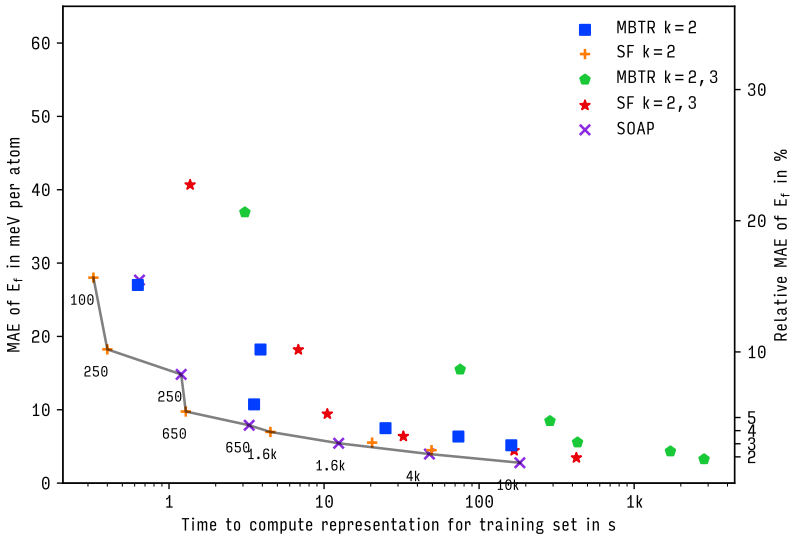


Figure B.3: Learning curves for selected representations on datasets `qm9` (top), `ba10` (centre), and `nmd18r` (bottom). Shown are MAE and rMAE of energy predictions on out-of-sample-data as a function of training set size. Boxes, whiskers, bars, crosses show interquartile range, total range, median, mean, respectively. Lines are fits to theoretical asymptotic MAE.

See figure 3.3.2 for RMSE.





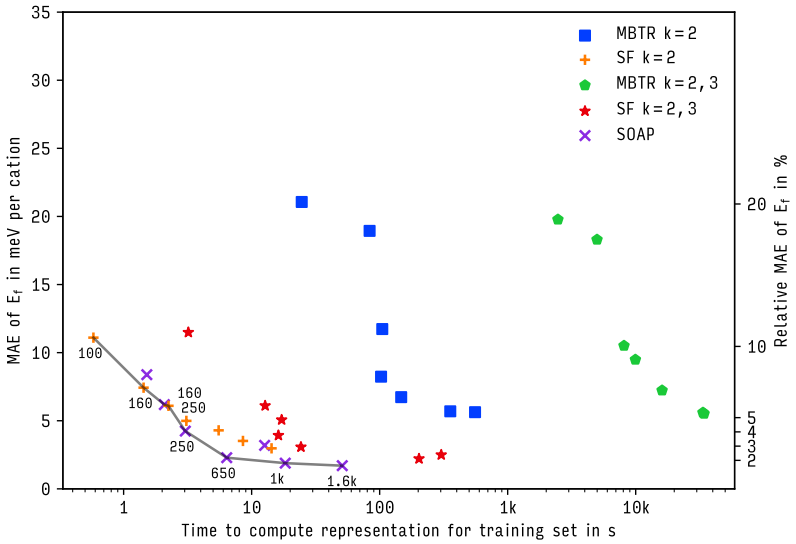


Figure B.4: Compute times of selected representations for datasets `qm9` (top), `ba10` (centre), and `nmd18r` (bottom). Shown are MAE and rMAE of energy predictions on out-of-sample-data as a function of the time needed to compute all representations in a training set. Lines indicate Pareto frontiers; inset numbers show training set sizes.

See figure 3.3.3 for RMSE.

Literature Values

	Error in kcal/mol			
Reference	MAE	RMSE	N	Method
[314]	1.5	2.8	5 k	IDMBR
[313]	0.72	—	10 k	SOAP
[42]	1.27	—	10 k	SchNet
[312]	0.44	—	10 k	FCHL [b]
[313]	0.66	—	10 k	FCHL [c]
[469]	0.14	—	100 k	SOAP [d]
[470]	0.35	0.94	100 k	SchNet
[309]	0.58	—	118 k	HDAD
[467]	—	2.11	12 k	SOAP
[467]	—	1.43	102 k	SOAP
[471]	0.15	—	110 k	DimeNet++
[386]	0.15	—	110 k	PaiNN
[472]	0.11	—	110 k	Allegro
here	0.49	0.90	10 k	SOAP

Table B.1: Selected performance estimates for `qm9` from the literature.
[a] original FCHL18 version [312]
[b] revised FCHL19 version [313]
[c] radial-scaling modification

	Error / meV/atom				
Ref.	MAE	RMSE	rRMSE in %	N	Method
[308]	5.3	—	—	10 k	MBTR
[308]	3.4	—	—	10 k	MTP
[467]	80.0[a]	—	0.53[a]	12 k	SOAP
here	2.8	4.6	2.60	10 k	SOAP

Table B.2: Selected performance estimates for `ba10` from the literature.
[a] report RMSE for total energy, as opposed to formation energy per atom. For rRMSE, we divide by standard deviation of total energy over entire dataset.

	Error / meV/cation			
Ref.	MAE	RMSE	N	Method
[354]	13	—	2 400	SOAP
here	14–15	24–26	1 600	all

Table B.3: Selected performance estimates for `nmd18u` from the literature. Here, all representations performed roughly equally. At the time of printing, no published results existed for the relaxed `nmd18r` version.

Appendix C
Heat Flux for Semi-Local Machine-Learning Potentials

C.1 Testing Forces and Stress

In order to verify the approach described in section 4.1, and to test the `glp` framework, potential energy, forces, and stress for the Lennard-Jones potential were compared with the implementation in ASE, where derivatives are computed analytically, and all operations are performed in double precision arithmetic.

The experiment consists of computing these properties for 100 randomly perturbed simulation cells of Lennard-Jones argon as defined in appendix C.2, starting from an $8 \times 8 \times 8$ supercell of the face-centred cubic primitive cell with lattice parameter 3.72 Å and angle 60°.

Positions are then perturbed slightly, with perturbations drawn from a normal distribution with σ=0.01 Å; a random strain with each component drawn from a uniform distribution over $[-0.1, 0.1]$ is also applied. For each such structure, energy, forces, and stress are computed using a number of different approaches

- The analytical implementation in ASE,
- `glp` implementation where $\mathcal{G}$ is constructed using the MIC in *fractional* coordinates,
- `glp` implementation where $\mathcal{G}$ is constructed using the MIC using explicit offsets,
- `glp` implementation using the 'unfolded' graph construction where no MIC is applied,
- Finite differences using a `glp` calculator, where the displacements have been chosen to minimise the deviation from ASE, see figure C.1.

For the forces, we compare approaches where gradients are computed with respect to positions directly, and those where gradients are compared with respect to edges. For the stress, equations (4.1.15) to (4.1.20) are compared, using the graph construction required to produce the correct stress.

Tables 4.1.1 and C.1 to C.3 show the result. For the forces, all MIC-based implementation approaches are equivalent and in excellent agreement with the analytical implementation. For the stress, all given formulations are found to be equivalent as well. In single precision arithmetic, the AD-based implementations are more accurate than finite differences, in double precision, errors are similar.

Comment	Grads	MAE in eV/Å	maxAE in eV/Å
MIC (offset)	edges	2.72×10^{-7}	2.07×10^{-6}
MIC (frac.)	edges	2.58×10^{-7}	1.88×10^{-6}
MIC (offset)	direct	2.72×10^{-7}	2.07×10^{-6}
MIC (frac.)	direct	2.58×10^{-7}	1.89×10^{-6}
Unfolded	direct	1.76×10^{-7}	1.64×10^{-6}

Table C.1: Comparison of deviations in forces for different GLP implementations of the Lennard-Jones potential, compared with the implementation in ASE. Results are shown for single precision arithmetic.

Comment	Grads	MAE in eV/Å	maxAE in eV/Å
MIC (offset)	edges	3.58×10^{-10}	2.21×10^{-9}
MIC (frac.)	edges	3.58×10^{-10}	2.21×10^{-9}
MIC (offset)	direct	3.58×10^{-10}	2.21×10^{-9}
MIC (frac.)	direct	3.58×10^{-10}	2.21×10^{-9}
Unfolded	direct	3.58×10^{-10}	2.21×10^{-9}

Table C.2: Comparison of deviations in forces for different GLP implementations of the Lennard-Jones potential, compared with the implementation in ASE. Results are shown for double precision arithmetic.

Equation	MAE in eV	maxAE in eV	MAPE in %	maxAPE in %
Fin. diff.	1.13×10^{-6}	7.49×10^{-6}	1.18×10^{-4}	8.55×10^{-3}
4.1.15	3.15×10^{-6}	1.04×10^{-5}	3.69×10^{-4}	2.72×10^{-2}
4.1.16	3.15×10^{-6}	1.04×10^{-5}	3.69×10^{-4}	2.72×10^{-2}
4.1.17	3.15×10^{-6}	1.04×10^{-5}	3.69×10^{-4}	2.72×10^{-2}
4.1.18	3.15×10^{-6}	1.04×10^{-5}	3.69×10^{-4}	2.72×10^{-2}
4.1.19	3.15×10^{-6}	1.04×10^{-5}	3.69×10^{-4}	2.72×10^{-2}
4.1.20	3.15×10^{-6}	1.04×10^{-5}	3.69×10^{-4}	2.72×10^{-2}

Table C.3: Error in stress for Lennard-Jones argon, comparing different formulations, as well as finite differences, with a baseline implementation in ASE. Results are shown for double precision arithmetic, and for $\sigma \cdot V$ in place of σ.

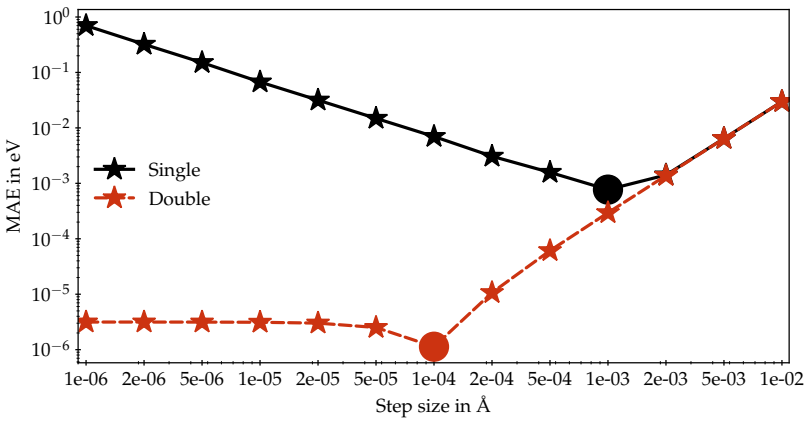


Figure C.1: MAE of stress obtained via finite differences compared to the analytical implementation for different choices of displacement, for single and double precision. The minimum is marked with a dot.

C.2 Lennard-Jones Argon

The following parameters are used to model Lennard-Jones argon, following reference [473]: σ=3.405 Å, ϵ=0.010 42 eV, r_c=10.5 Å, r_o=9.0 Å, where r_o is the 'onset' parameter of the cutoff function f_c that ensures that forces smoothly decay to zero as the cutoff radius r_c is approached.

This cutoff function is multiplied onto $U(r_{ij})$ and is defined as

$$f_c(r) = \begin{cases} 1 & \text{for } r < r_o \\ \frac{(r_c^2 - r^2)^2 (r_c^2 + 2r^2 - 3r_o^2)}{(r_c^2 - r_o^2)^3} & \text{for } r_o \leq r \leq r_c \\ 0 & \text{for } r > r_c \,. \end{cases} \tag{C.1}$$

C.3 Efficient Unfolding

In order to implement the unfolded heat flux, we must be able to efficiently construct $\mathcal{R}_{\text{unf}}$ from $\mathcal{R}_{\text{sc}}$ and $\mathcal{B}$. We present a simple and vectorisable approach.

Consider our objective: For each position $\mathcal{R}_{\text{sc}}$, we must determine *(a)* whether it requires replication at all, and *(b)* if yes, in which directions. Both can be achieved by breaking the task down into a series of steps.

1. For each position $i \in \mathcal{R}_{\text{sc}}$, determine whether it lies within r_c^{eff} of any of the six faces of the simulation cell. If this is the case, i is said to have a collision with that boundary. (See appendix C.4 for a discussion of the non-orthorhombic case.).
2. From the resulting collisions, compute which replicas of each i must be constructed. There are 26 possibilities:
   (a) If i collides with a boundary, it is replicated in the opposite direction of that boundary (6 cases).
   (b) If it collides with two, it collides with an edge and must therefore also be replicated diagonally (12 cases).
   (c) If it collides with three boundaries, it collides with a corner and must also be replicated diagonally (8 cases).
3. Construct the replicas from this prescription.

Separating steps two and three allows the determination of the unfolding outside the computational graph, and also enables the caching of results: by increasing the cutoff used for unfolding by a tolerance r_{tol}, it must only be recomputed if the distance of any position to the boundary changes by more than $r_{\text{tol}}/2$. While the unfolding is typically not a bottleneck in the systems investigated in this work, this caching also avoids oscillations of atoms close to the cutoff boundary, which in turn avoids cache misses in the neighbourlist construction.

C.4 Unfolding for Non-Orthorhombic Systems

The simulation cell is spanned by $\mathcal{B} = (\boldsymbol{b}_1, \boldsymbol{b}_2, \boldsymbol{b}_3)$, forming a parallelpepiped. Our task is to determine the sets of points within[1] this parallelpepiped that lie within r_c^{eff} of each of the six faces. There are two straightforward approaches to solving this problem: We can project all positions onto the surface normals of the faces, directly measuring the distances to the faces, or we can transition to fractional co-

[1] We assume that in a pre-processing step, every positions has been wrapped into the simulation cell.

ordinates, where distances to the faces can be obtained by inspecting the components of the resulting coordinates. We choose the former of these two equivalent approaches.

As a first step, we therefore obtain the surface normals $\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3$, which are simply the normalised reciprocal lattice vectors.[2] The surface normals are oriented inwards, in the direction of, but in general not parallel to, the basis vector that shares the same index.

Then, we compute the distances d_i between opposite faces by projecting the basis vectors onto the surface normals, $d_\alpha = \boldsymbol{b}_\alpha \cdot \boldsymbol{a}_\alpha$. Similarly, for each position $\boldsymbol{r}_i$ we compute the components along the surface normals, yielding projected positions $r_i'^{\alpha} = \boldsymbol{r}_i \cdot \boldsymbol{a}_\alpha$. Now, collisions with the two surfaces in direction α can be detected by checking $r_i'^{\alpha} \leq r_\mathrm{c}^\mathrm{eff}$ or $r_i'^{\alpha} \geq d_\alpha - r_\mathrm{c}^\mathrm{eff}$.

[2] In particular

$$\boldsymbol{a}_1 \propto \boldsymbol{b}_2 \times \boldsymbol{b}_3$$
$$\boldsymbol{a}_2 \propto \boldsymbol{b}_3 \times \boldsymbol{b}_1$$
$$\boldsymbol{a}_3 \propto \boldsymbol{b}_1 \times \boldsymbol{b}_2 .$$

C.5 Size of Unfolded System

We compute the number of replica positions generated during unfolding, assuming a constant number density $\rho = N/V$. For simplicity, we consider a cubic system, but note that the result can be readily generalised to other systems. In such a system $V = L^3$, where $L = |\boldsymbol{b}_a|$. We define $c \coloneqq r_\mathrm{c}^\mathrm{eff}$.

We first calculate the volumes of the three different collision areas:

1. Corners have volume c^3.
2. Edges, excluding corners, have volume $(L-c)c^2$.
3. Remaining collision areas on the surfaces have volume $(L-c)^3$.

The total replicated volume is therefore

$$V_\mathrm{rep} = 8c^3 + 12(L-c)c^2 + 6(L-c)^2 c\,, \tag{C.2}$$

the number of replicated atoms is ρV_rep. It scales quadratically with L, and therefore is proportional to $V^{2/3}$ or, equivalently, $N^{2/3}$. This is asymptotically dominated by N, and therefore, the number of total positions after unfolding is $O(N)$. Note that the additional number of atoms scales *cubically* with c.

Table C.4 shows real-world numbers for the size of unfolded simulation cells for materials used in this work. As N increases, at constant density, the relative increase in positions to be considered decreases.

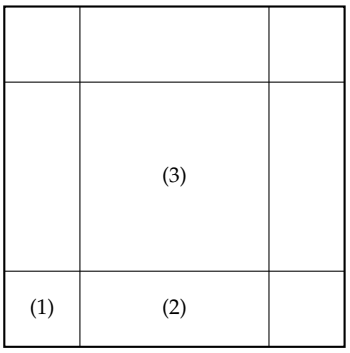


Figure C.2: Sketch of collision areas, viewing the face of a cubic simulation cell.

Si (400 K)			ZrO_2 (300 K)		
N	N_unf	% increase	N	N_unf	% increase
512	3736	630	768	5888	667
1728	7430	330	1500	8424	462
4096	12 996	217	4116	15 488	276
8000	20 818	160	8748	25 688	194

Table C.4: Number of atoms N in the simulation cell, number of atoms in the unfolded system N_unf, and increase in % for Si and ZrO_2 simulation cells used in this work, for $r_\mathrm{c}^\mathrm{eff}$=10 Å.

C.6 Heat Flux in the Literature

We provide a brief overview of selected heat flux formulations in the literature, focusing on those that provide a closed form suitable for implementation with MD, which excludes, for instance, the work by Irving and Kirkwood [396].

Noll (1955)

Noll [397][3] provides a reformulation of the work by Irving and Kirkwood, avoiding infinite series. In the notation of this thesis,[4] the heat current density [397, eq. 2.13-2.16] for a pair potential $U = 1/2 \sum_{ij} U_{ij}$ is given by

$$\boldsymbol{j}(\boldsymbol{r}) = \boldsymbol{j}_{\text{kinetic}}(\boldsymbol{r}) + \boldsymbol{j}_{\text{transport}}(\boldsymbol{r}) + \boldsymbol{j}_{\text{interaction}}(\boldsymbol{r}) \tag{C.3}$$

$$\boldsymbol{j}_{\text{kinetic}}(\boldsymbol{r}) = \sum_i T_i \boldsymbol{v}_i \Delta(\boldsymbol{r} - \boldsymbol{r}_i) \tag{C.4}$$

$$\boldsymbol{j}_{\text{transport}}(\boldsymbol{r}) = \sum_i U_i \boldsymbol{v}_i \Delta(\boldsymbol{r} - \boldsymbol{r}_i) \tag{C.5}$$

$$\boldsymbol{j}_{\text{interaction}}(\boldsymbol{r}) = -\sum_i \boldsymbol{r}_{ij} U'_{ij} \left(\overline{\boldsymbol{r}}_{ij} \cdot \frac{\boldsymbol{v}_i + \boldsymbol{v}_j}{2} \right) \Lambda_{ij}(\boldsymbol{r}) \,; \tag{C.6}$$

for $\boldsymbol{j}_{\text{interaction}}$ we have used that only terms in [397, eq. 2.16] where $\boldsymbol{z} = \boldsymbol{r}_{ij}$ and $\boldsymbol{x}$ on the line segment between i and j contribute. $\boldsymbol{j}_{\text{kinetic}} + \boldsymbol{j}_{\text{transport}}$ yield the second term in equation (4.2.11). $\boldsymbol{j}_{\text{interaction}}$, describes the exchange of potential (not total) energy, which gives rise to the different expression.

The work by Noll provides the foundation for the later work by Hardy [398] that section 4.2 relies on. However, it only considers additive pairwise potentials, and does not discuss periodicity, and is therefore not sufficient for the present work.

[3] An English translation can be found in reference [474].

[4] We drop the velocity density, as it is not relevant for solids. We also replace Noll's notation for expectation values with Hardy's [398] localisation functions.

Hardy (1963)

Hardy, in his earlier work in 1963, reference [381] set out to derive a quantum-mechanical operator for heat flux in a periodic quantum system. It reads [381, eq. 2.14]:

$$\frac{1}{2} \sum_i \frac{\hat{\boldsymbol{p}}_i}{m_i} \left(\frac{\hat{\boldsymbol{p}}_i \cdot \hat{\boldsymbol{p}}_i}{2m_i} + \hat{U}_i \right) + \sum_{ij} (\hat{\boldsymbol{r}}_i - \hat{\boldsymbol{r}}_j) \frac{1}{i\hbar} \left[\frac{\hat{\boldsymbol{p}}_i \cdot \hat{\boldsymbol{p}}_i}{2m_i}, \hat{U}_j \right] + \text{H.c.}\,. \tag{C.7}$$

'H.c.' denotes the Hermitian conjugate and square brackets a commutator [48].

The classical equivalent is shown [371] to be

$$\sum_i E_i \boldsymbol{v}_i + \sum_{ij} \boldsymbol{r}_{ij} \frac{\partial U_j}{\partial \boldsymbol{r}_i} \cdot \boldsymbol{v}_i \,, \tag{C.8}$$

the 'Hardy' heat flux in equation (4.2.14), once i and j are exchanged.

However, equation (C.8) does not yet immediately provide information of the range of the sum over i and j in the second, potential, term, which hinders its unambiguous implementation for periodic systems: If they run over the simulation cell, the second term in equation (C.8) is a total time derivate and contributes no thermal conductivity due to

the gauge principle. Introducing the MIC to the $\boldsymbol{r}_{ij}$ prefactors provides an ad-hoc solution, provided the range of the involved potentials admits treatment with the MIC, see section 4.2.7. The general solution, however, is to restrict one sum to $\mathcal{R}_{\text{sc}}$ and let one run over the bulk $\mathcal{R}_{\text{all}}$, which is obtained in section 4.2.4.

Apart from this difficulty, as discussed in section 4.2.6, a direct implementation of this heat flux requires explicit access to the Jacobian, in other words, all derivatives $\partial U_i / \partial \boldsymbol{r}_j$. Without making use of the structure of this Jacobian, evaluation scales linearly with either the number of input or putput dimensions (see section 2.4.6), and therefore quadratically with N.

Hardy (1983)

The later work by Hardy, from 1983, aimed to provide a formulation of the heat current density, as well as other quantities, suitable for MD. Using similar techniques as Noll [397], two contributions to the heat current density are given, also under the assumption of a pairwise additive potential:[5]

$$\boldsymbol{j}_{\text{kinetic}}(\boldsymbol{r}) = \sum_i \frac{1}{2} E_i \boldsymbol{v}_i \Delta(\boldsymbol{r} - \boldsymbol{r}_i) \tag{C.9}$$

$$\boldsymbol{j}_{\text{potential}}(\boldsymbol{r}) = \sum_{ij} \frac{1}{2} \boldsymbol{r}_{ji} \left(-\frac{\partial U_{ij}}{\partial \boldsymbol{r}_i} \cdot \boldsymbol{v}_i \right) \Lambda_{ij}(\boldsymbol{r}) \,, \tag{C.10}$$

[5] As before, we discard velocity (or momentum) density.

with renaming of indices, equation (4.2.11) restricted to additive pairwise potentials is obtained. This work also provides the necessary argument for integrating the heat current density in the periodic case, used in section 4.2.4 and was therefore used as starting point for section 4.2, which extends it to non-additive pairwise potentials.

Fan et al. (2015)

Fan et al. [371] aim to unify previous formulations of the heat flux for many-body FFs, in particular the Tersoff and SW FFs. Arguing that all such potentials can be written as a sum over atomic potential energy contributions that depend only on atom-pair vectors,

$$U_i = U_i(\{ \boldsymbol{r}_{ij} \mid j \in \mathcal{N}(i) \}) \tag{C.11}$$

they obtain

$$\sum_i E_i \boldsymbol{v}_i + \sum_{ij} \boldsymbol{r}_{ji} \frac{\partial U_i}{\partial \boldsymbol{r}_{ij}} \cdot \boldsymbol{v}_j \,, \tag{C.12}$$

which is equation (C.8) with indices renamed and $\partial U_i / \partial \boldsymbol{r}_j = \partial U_i / \partial \boldsymbol{r}_{ij}$, which is a consequence of the definition of the functional dependence of U_i in equation (C.11).

In this form, the ambiguity in periodic systems has been resolved: The sum over j is restricted to neighbours of i, while the sum over i runs over the simulation cell. However, this form is not directly suitable for GLPs, as equation (C.11) does not hold for $M{>}1$. In principle,

neighbourhoods could be extended to include all interacting neighbours, recovering equation (C.11), but this negates the computational advantages of a GLP architecture, and is therefore not pursued.

Carbogno et al. (2017)

Carbogno et al. [143] provide a heat flux for DFT. In essence, their approach is based on the observation that the electronic Hamiltonian in equation (2.1.6), which consists of one- and two-body terms involving atomic and electronic degrees of freedom, together with the Hellmann-Feynman theorem, lends an all-to-all pairwise structure to the forces in DFT. This structure can then be used to derive a form of $\boldsymbol{J}_{\text{pot}}$ composed of terms that appear in the definition of the stress, and are therefore readily computed in DFT, without requiring an explicit partitioning of the potential energy into atomic contributions.

In the context of GLPs, this approach cannot be used: It relies on all-to-all pairwise interactions, which is combined with a many-body electron density to yield a many-body PES. In the MLIPs investigated in this thesis, the PES is approximated as local, potentially many-body, function of positions, which does not feature all-to-all interactions.

C.7 Identities for Localisation Functions

We start by defining

$$\boldsymbol{x} = \lambda \boldsymbol{r}_i + (1-\lambda)\boldsymbol{r}_j - \boldsymbol{r}\,. \tag{C.13}$$

We then compute the derivative of $\Delta(\cdot)$ with respect to λ:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\lambda}\Delta(\boldsymbol{x}) &= (\boldsymbol{r}_i - \boldsymbol{r}_j)\cdot\boldsymbol{\nabla}_{\boldsymbol{x}}\Delta(\boldsymbol{x}) && \text{(C.14)}\\
&= -(\boldsymbol{r}_i - \boldsymbol{r}_j)\cdot\boldsymbol{\nabla}_{\boldsymbol{r}}\Delta(\lambda\boldsymbol{r}_i + (1-\lambda)\boldsymbol{r}_j - \boldsymbol{r}) && \text{(C.15)}\\
&= \boldsymbol{r}_{ij}\cdot\boldsymbol{\nabla}_{\boldsymbol{r}}\Delta(\lambda\boldsymbol{r}_i + (1-\lambda)\boldsymbol{r}_j - \boldsymbol{r})\,, && \text{(C.16)}
\end{aligned}$$

where the last line can be easily verified by comparing $\boldsymbol{\nabla}_{\boldsymbol{r}}\Delta(\cdot)$ and $\boldsymbol{\nabla}_{\boldsymbol{x}}\Delta(\cdot)$. Then, we simply substitute this relation into

$$\begin{aligned}
\boldsymbol{r}_{ij}\boldsymbol{\nabla}_{\boldsymbol{r}}\Lambda_{ij}(\boldsymbol{r}) &= \boldsymbol{r}_{ij}\boldsymbol{\nabla}_{\boldsymbol{r}}\left(\int_0^1 \mathrm{d}\lambda\,\Delta(\lambda\boldsymbol{r}_i + (1-\lambda)\boldsymbol{r}_j - \boldsymbol{r})\right) && \text{(C.17)}\\
&= \int_0^1 \mathrm{d}\lambda\,\frac{\mathrm{d}}{\mathrm{d}\lambda}\Delta(\lambda\boldsymbol{r}_i + (1-\lambda)\boldsymbol{r}_j - \boldsymbol{r}) && \text{(C.18)}\\
&= \Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \Delta(\boldsymbol{r}_j - \boldsymbol{r})\,. && \text{(C.19)}
\end{aligned}$$

C.8 Time-Derivative of the Energy Density

We begin by considering the time derivative of E_i.

$$\frac{\mathrm{d}}{\mathrm{d}t}E_i = \frac{\mathrm{d}}{\mathrm{d}t}U_i + \frac{\mathrm{d}}{\mathrm{d}t}T_i \tag{C.20}$$

$$= \sum_j \frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j + \sum_{i=1}^{N} F_i \cdot \boldsymbol{v}_i \tag{C.21}$$

$$= \sum_j \frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j - \sum_{i=1}^{N} \frac{\partial U}{\partial \boldsymbol{r}_i} \cdot \boldsymbol{v}_i \,. \tag{C.22}$$

Then, we re-arrange it in a pair-wise form

$$\sum_{i=1}^{N} \left(\frac{\mathrm{d}}{\mathrm{d}t}E_i\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) \tag{C.23}$$

$$= \sum_{i=1}^{N} \left(\sum_j \frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \sum_{i=1}^{N} \left(\frac{\partial U}{\partial \boldsymbol{r}_i} \cdot \boldsymbol{v}_i\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) \tag{C.24}$$

$$= \sum_{i,j=1}^{N} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \sum_{i=1}^{N} \left(\frac{\partial (\sum_j U_j)}{\partial \boldsymbol{r}_i} \cdot \boldsymbol{v}_i\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) \tag{C.25}$$

$$= \sum_{i,j=1}^{N} \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \sum_{i,j=1}^{N} \left(\frac{\partial U_j}{\partial \boldsymbol{r}_i} \cdot \boldsymbol{v}_i\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) \tag{C.26}$$

$$= \sum_{i,j=1}^{N} \left[\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \left(\frac{\partial U_j}{\partial \boldsymbol{r}_i} \cdot \boldsymbol{v}_i\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r})\right] \tag{C.27}$$

$$= \sum_{i,j=1}^{N} \left[\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \Delta(\boldsymbol{r}_j - \boldsymbol{r})\right] \tag{C.28}$$

$$= \sum_{i,j=1}^{N} \left[\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j} \cdot \boldsymbol{v}_j\right) \left(\Delta(\boldsymbol{r}_i - \boldsymbol{r}) - \Delta(\boldsymbol{r}_j - \boldsymbol{r})\right)\right] , \tag{C.29}$$

C.9 Heat Flux in Solids

Similar arguments appear in [212, 367].

In solids, atomic positions are bounded over time. In other words, atomic positions $\boldsymbol{r}_i(t)$ can be split into a fixed *reference* position $\boldsymbol{r}_i^0$ and a time-dependent displacement from that position $\boldsymbol{u}_i(t)$:

$$\boldsymbol{r}_i(t) \coloneqq \boldsymbol{r}_i^0 + \boldsymbol{u}_i(t) \quad i \in \mathcal{R}_{\text{all}} \tag{C.30}$$

If we choose $\boldsymbol{r}_i^0$ such that it contains the information about which replica i belongs to,[6] we obtain:

[6] For instance by choosing positions in the pristine supercell, or $\boldsymbol{r}_i^0 = \boldsymbol{r}(t=0)$, or $\boldsymbol{r}_i^0 = \langle \boldsymbol{r}(t) \rangle$ over the simulation run.

$$\boldsymbol{r}_{i\boldsymbol{n}}(t) = \boldsymbol{r}_{i\boldsymbol{n}}^0 + \boldsymbol{u}_i(t) \tag{C.31}$$

$$\Rightarrow \dot{\boldsymbol{r}}_{i\boldsymbol{n}}(t) = \boldsymbol{v}_i(t) \,. \tag{C.32}$$

In other words, the displacements and velocities are shared between all replicas of i. Substituting into equation (4.2.15), we obtain:

$$\boldsymbol{J} = \sum_{\substack{i\in\mathcal{R}_{\mathrm{sc}}\\ j\in\mathcal{R}_{\mathrm{all}}}} \left(\boldsymbol{r}_{ji}^{0}\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right) + \boldsymbol{u}_{ji}\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) + \sum_{i\in\mathcal{R}_{\mathrm{sc}}} E_i\boldsymbol{v}_i \tag{C.33}$$

$$= \sum_{\substack{i\in\mathcal{R}_{\mathrm{sc}}\\ j\in\mathcal{R}_{\mathrm{all}}}} \left(\boldsymbol{r}_{ji}^{0}\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) + \sum_{i\in\mathcal{R}_{\mathrm{sc}}} \boldsymbol{u}_i\dot{E}_i + \sum_{i\in\mathcal{R}_{\mathrm{sc}}} E_i\boldsymbol{v}_i \tag{C.34}$$

$$= \sum_{\substack{i\in\mathcal{R}_{\mathrm{sc}}\\ j\in\mathcal{R}_{\mathrm{all}}}} \left(\boldsymbol{r}_{ji}^{0}\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) + \frac{\mathrm{d}}{\mathrm{d}t}\sum_{i\in\mathcal{R}_{\mathrm{sc}}} \boldsymbol{u}_i E_i \tag{C.35}$$

$$=: \boldsymbol{J}_{\mathrm{int}} + \boldsymbol{J}_{\mathrm{disp}} =: \boldsymbol{J}_{\mathrm{full}}\,. \tag{C.36}$$

The details of rewriting the middle term are given in appendix C.10.

Since the displacements are, by definition, bounded over time, and E_i is bounded by the total energy, $\boldsymbol{J}_{\mathrm{disp}}$ is the time-derivative of a bounded quantity. By the gauge invariance principle, it does not contribute to the thermal conductivity, and can therefore be neglected. The remaining term, $\boldsymbol{J}_{\mathrm{int}}$, solely determines the thermal conductivity, which occurs through the *interaction*, i.e. energy exchange, between atoms, rather than through *displacement* of the atoms themselves [151, 212].

C.10 Displacement Term in Heat Flux for Solids

Our task is to re-arrange:

$$\sum_{\substack{i\in\mathcal{R}_{\mathrm{sc}}\\ j\in\mathcal{R}_{\mathrm{all}}}} \left(\boldsymbol{u}_{ji}\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) \tag{C.37}$$

into the form

$$\sum_{i\in\boldsymbol{r}_u c} \boldsymbol{u}_i\left(\dot{U}_i + \dot{T}_i\right)\,. \tag{C.38}$$

To achieve this, we first decompose into two terms:

$$\sum_{\substack{i\in\mathcal{R}_{\mathrm{sc}}\\ j\in\mathcal{R}_{\mathrm{all}}}} \left(\boldsymbol{u}_i\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) - \sum_{\substack{i\in\mathcal{R}_{\mathrm{sc}}\\ j\in\mathcal{R}_{\mathrm{all}}}} \left(\boldsymbol{u}_j\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) \tag{C.39}$$

The sum over j in the first term is simply the time-derivative of U_i. In the second term, we can move the sum over i inside the derivative,

$$= \sum_{i\in\mathcal{R}_{\mathrm{sc}}} \boldsymbol{u}_i\dot{U}_i - \sum_{j\in\mathcal{R}_{\mathrm{all}}} \left(\boldsymbol{u}_j\left(\frac{\partial U}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) \tag{C.40}$$

We are done with the first term. In the second term, we split the sum over $j \in \mathcal{R}_{\mathrm{all}}$ into $j \in \mathcal{R}_{\mathrm{sc}}$ and its replicas $j\boldsymbol{n}$. For a fixed j in the unit cell, the sum will collect contributions from all replicas that interact with the unit cell. $\boldsymbol{u}_j$ and $\boldsymbol{v}_j$ are identical across replicas, so therefore:

$$-\sum_{j\in\mathcal{R}_{\mathrm{sc}}} \left(\boldsymbol{u}_j\left(\sum_{\boldsymbol{n}\in\mathbf{Z}}\left[\frac{\partial U}{\partial \boldsymbol{r}_{j\boldsymbol{n}}}\right]\cdot\boldsymbol{v}_j\right)\right) \tag{C.41}$$

$$= \sum_{j\in\mathcal{R}_{\mathrm{sc}}} \boldsymbol{u}_j\left(\boldsymbol{F}_j\cdot\boldsymbol{v}_j\right) \tag{C.42}$$

$$= \sum_{i\in\mathcal{R}_{\mathrm{sc}}} \boldsymbol{u}_i\dot{T}_i\,. \tag{C.43}$$

C.11 Unfolded Heat Flux

The remaining task is to rewrite equation (4.2.25) to take advantage of AD. We begin by splitting the atom-pair vector $\boldsymbol{r}_{ij}$:

$$\boldsymbol{J}_{\text{pot}} = \sum_{\substack{i\in\mathcal{R}_{\text{sc}}\\ j\in\mathcal{R}_{\text{unf}}}} \left(\boldsymbol{r}_{ji}\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) \tag{C.44}$$

$$= \sum_{\substack{i\in\mathcal{R}_{\text{sc}}\\ j\in\mathcal{R}_{\text{unf}}}} \left(\left[\boldsymbol{r}_i-\boldsymbol{r}_j\right]\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) \tag{C.45}$$

$$= \sum_{\substack{i\in\mathcal{R}_{\text{sc}}\\ j\in\mathcal{R}_{\text{unf}}}} \left(\boldsymbol{r}_i\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) - \sum_{\substack{i\in\mathcal{R}_{\text{sc}}\\ j\in\mathcal{R}_{\text{unf}}}} \left(\boldsymbol{r}_j\left(\frac{\partial U_i}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\right) \tag{C.46}$$

$$=: \boldsymbol{J}_1 - \boldsymbol{J}_2\,. \tag{C.47}$$

To use AD efficiently, we now rewrite the resulting expressions in a way that allows us to execute the sum over i before taking derivatives. For $\boldsymbol{J}_1$, we simply define

$$\boldsymbol{B} := \sum_{i\in\mathcal{R}_{\text{sc}}} \boldsymbol{r}_i^{\text{const}} U_i \tag{C.48}$$

Here $\boldsymbol{r}_i^{\text{const}}$ denotes positions that are numerically identical to $\boldsymbol{r}_i$ but are treated as constants during the calculation of derivatives.

and obtain

$$\boldsymbol{J}_1 = \sum_{j\in\mathcal{R}_{\text{unf}}} \frac{\partial \boldsymbol{B}}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\,. \tag{C.49}$$

Note that the dot product is taken between $\boldsymbol{r}_j$ and $\boldsymbol{v}_j$.

We note that if $\boldsymbol{r}^{\text{const}}$ are replaced with fixed reference positions $\boldsymbol{r}^0$, $\boldsymbol{J}_1$ is a total time derivative. Therefore, in that case, this term does not contribute to the thermal conductivity. We nevertheless retain this term for equivalence with other formulations, and for non-solids.

The case of $\boldsymbol{J}_2$ is straightforward, since the sum over i yields U,

$$\boldsymbol{J}_2 = \sum_{j\in\mathcal{R}_{\text{unf}}} \boldsymbol{r}_j\left(\frac{\partial U}{\partial \boldsymbol{r}_j}\cdot\boldsymbol{v}_j\right)\,. \tag{C.50}$$

Both terms can be obtained efficiently with AD: For $\boldsymbol{J}_1$, we require one JVP, or three JVPs, and for $\boldsymbol{J}_2$, a single evaluation of either is sufficient.

Appendix D
Thermal Conductivity with Message-Passing Neural Networks

D.1 Anharmonicity Scores

To inform the selection of materials in chapter 5, the anharmonicity score by Knoop et al. [417] has been employed. While the value for SnSe at 300 K has been obtained from reference [142], values for Si at 400 K and ZrO_2at 300 K and 1400 K were computed.

For ZrO_2, anharmonicity scores were obtained based on the harmonic sampling technique from reference [417], using `FHI-aims` and 30 samples in the 324-atom supercell used for phonon band structure calculations in figure 5.1.4.

For Si, σ^{A} was computed based on an aiMD trajectory of 4 ps duration, using the same settings as the training data used for Si in section 5.4. Force constants were obtained from the phonon band structure calculation reported in figure D.29.

D.2 Model for Zirconia

Training Data

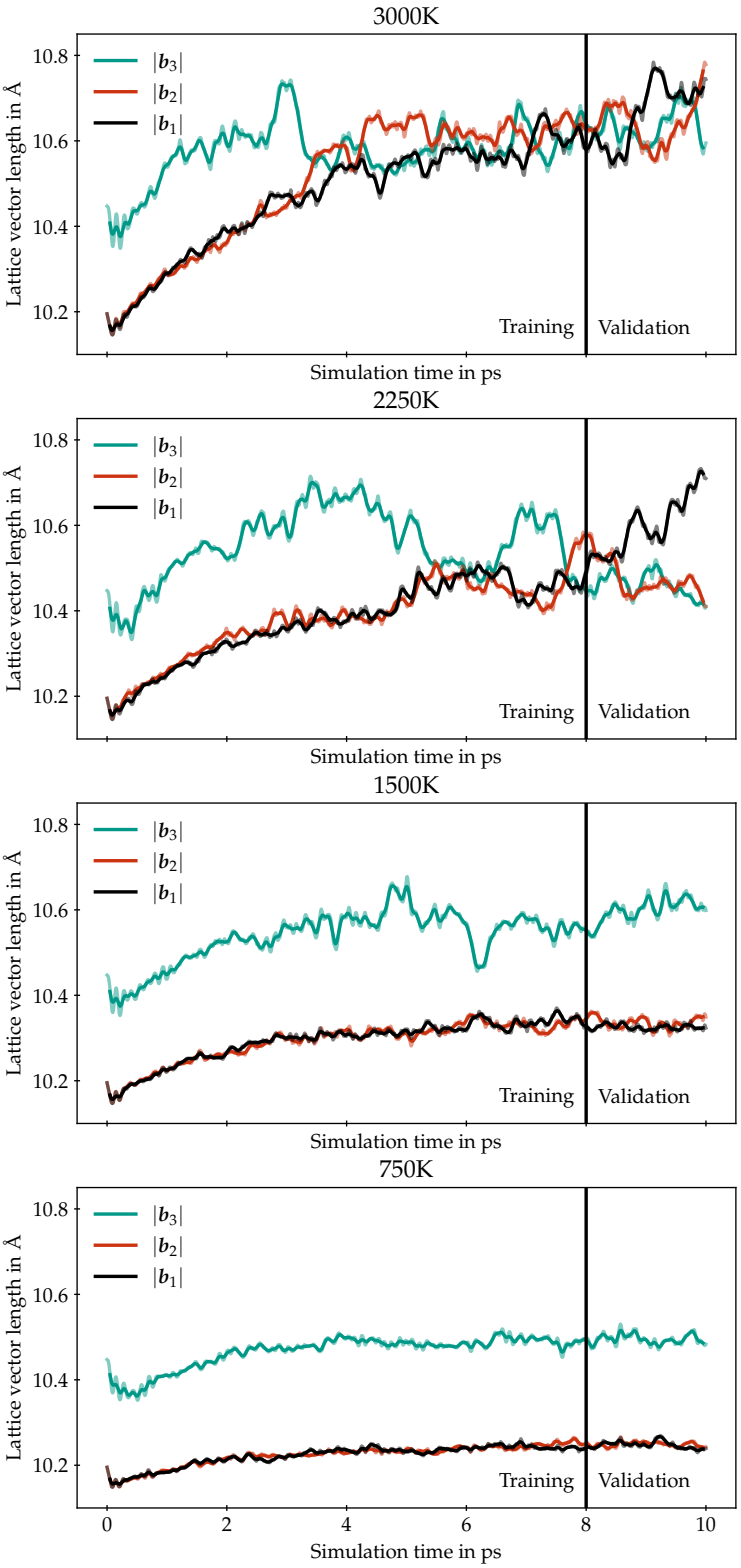


Figure D.1: Magnitude of lattice vectors for the zirconia NpT dataset. The vertical line separates the samples used for training from those used for validation.

Vibrational Density of States

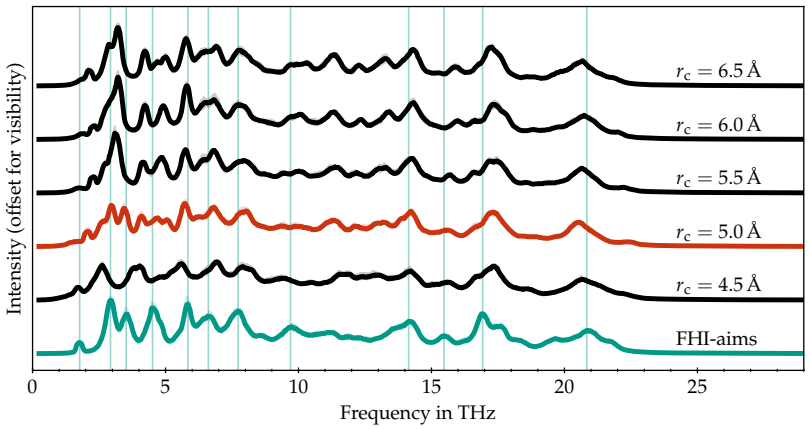
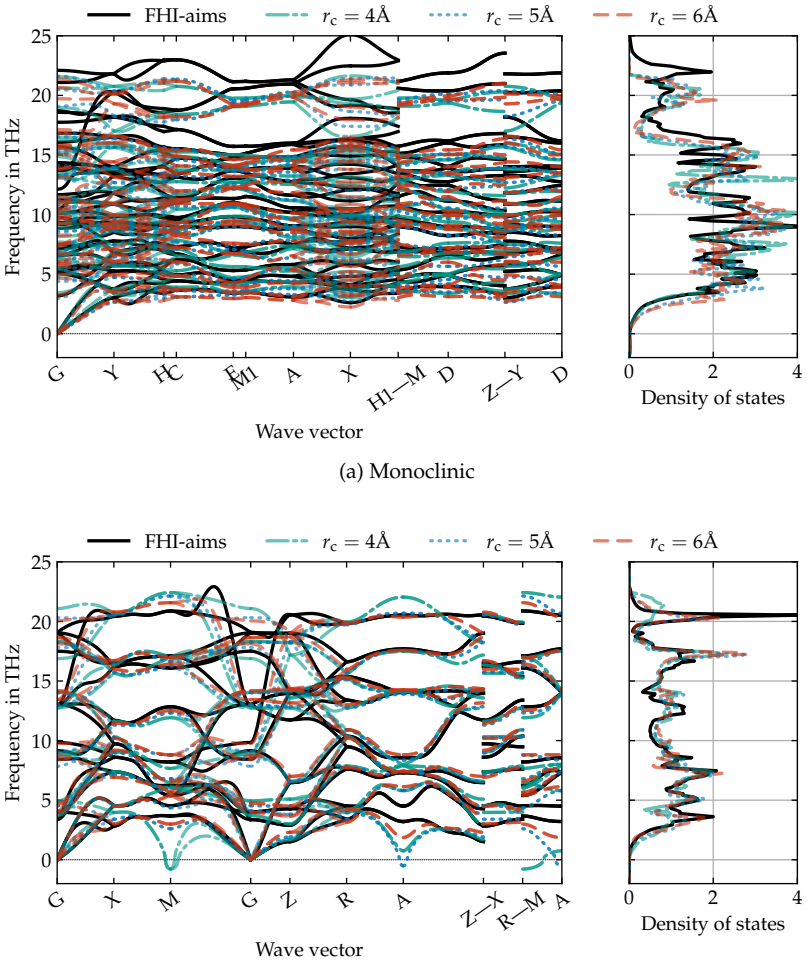


Figure D.2: Comparison of VDOS for MPNNs (M=1) with different cutoff radii compared to a baseline computed with `FHI-aims`. The chosen production cutoff radius is highlighted in red. Vertical lines indicate peaks in the `FHI-aims` result. Constant vertical offsets are applied to distinguish curves. Results are averaged over three trajectories of 60ps (Δt=4 fs), in the tetragonal phase at 300 K, with matching initial configurations. Shaded areas indicate the minimum and maximum.

The corresponding figure with M=2 can be found in figure 5.1.3.

Phonon Band Structures

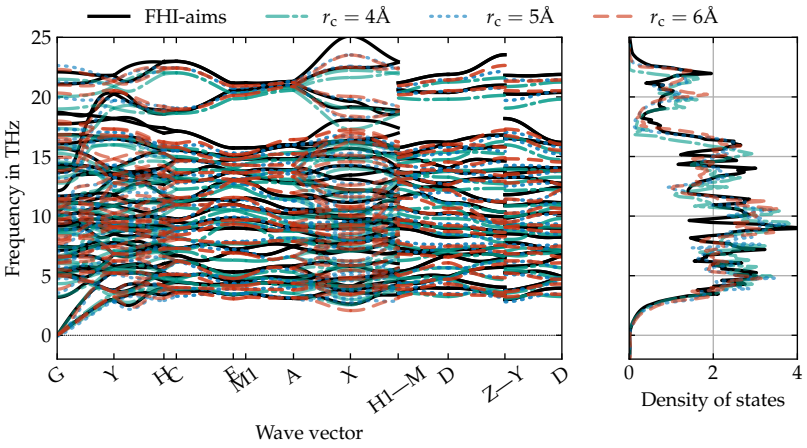


(a) Monoclinic

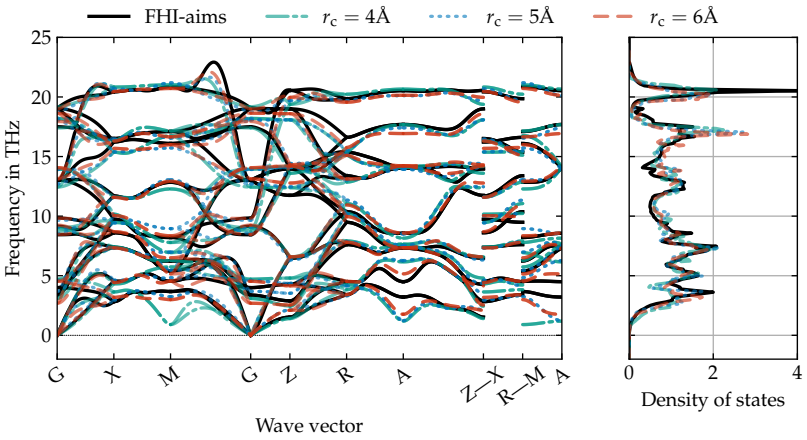
(b) Tetragonal

Figure D.3: Phonon band structure and density of states in the monoclinic (top) and tetragonal (bottom) phases, using alternate SchNet MLIPs with M=1 and different cutoff radii, compared to a `FHI-aims` reference calculation. Results are shown for a 324-atom supercell. Convergence with respect to supercell size was checked.

Production settings, M=2 and r_c=5.0 Å are shown in figure 5.1.4.



(a) Monoclinic



(b) Tetragonal

Figure D.4: Phonon band structure and density of states in the monoclinic (top) and tetragonal (bottom) phases, using alternate SchNet MLIPs with M=2 and different cutoff radii, compared to a `FHI-aims` reference calculation. Results are shown for a 324-atom supercell. Convergence with respect to supercell size was checked.

Production settings, M=2 and r_c=5.0 Å are shown in figure 5.1.4.

Displacements

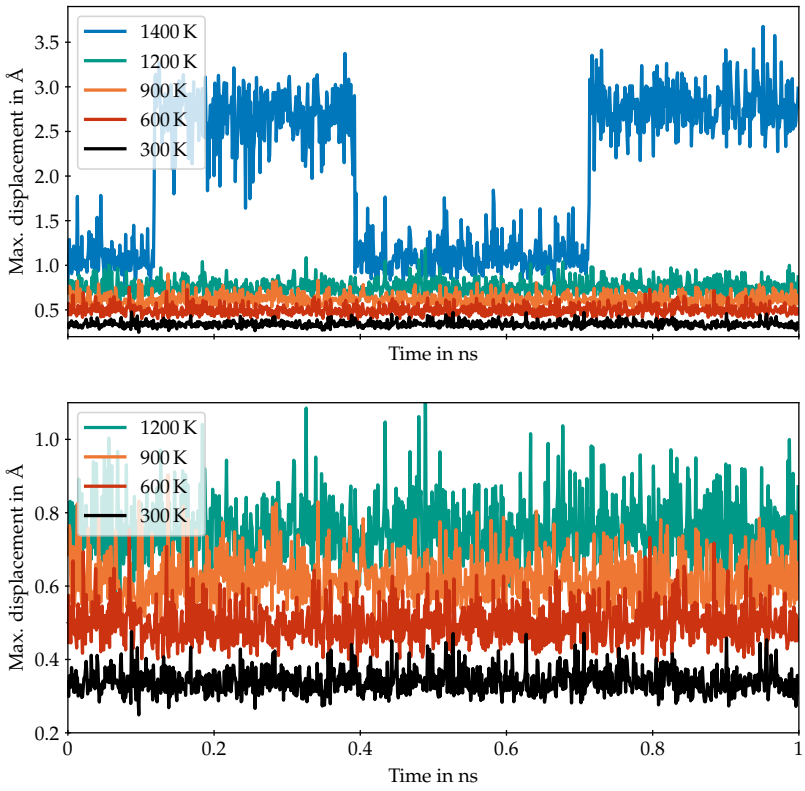


Figure D.5: Maximum displacement across oxygen atoms with respect to average positions over the course of a single trajectory at different temperatures in the monoclinic phase, computed with the M=2 and r_c=5.0 Å production SchNet MLIP. The top plot shows the full range of data, while the lower displays details.

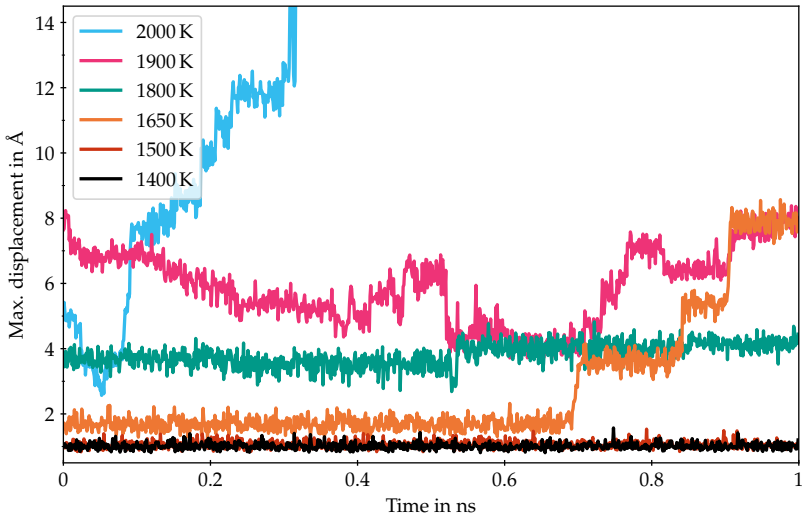


Figure D.6: Maximum displacement across oxygen atoms with respect to average positions over the course of a single trajectory at different temperatures in the tetragonal phase, computed with the M=2 and r_c=5.0 Å production SchNet MLIP.
This figure includes a trajectory at 2000 K where an instability is encountered: After significant oxygen diffusion, the potential becomes unstable and displacements, along with energy and temperature, diverge.

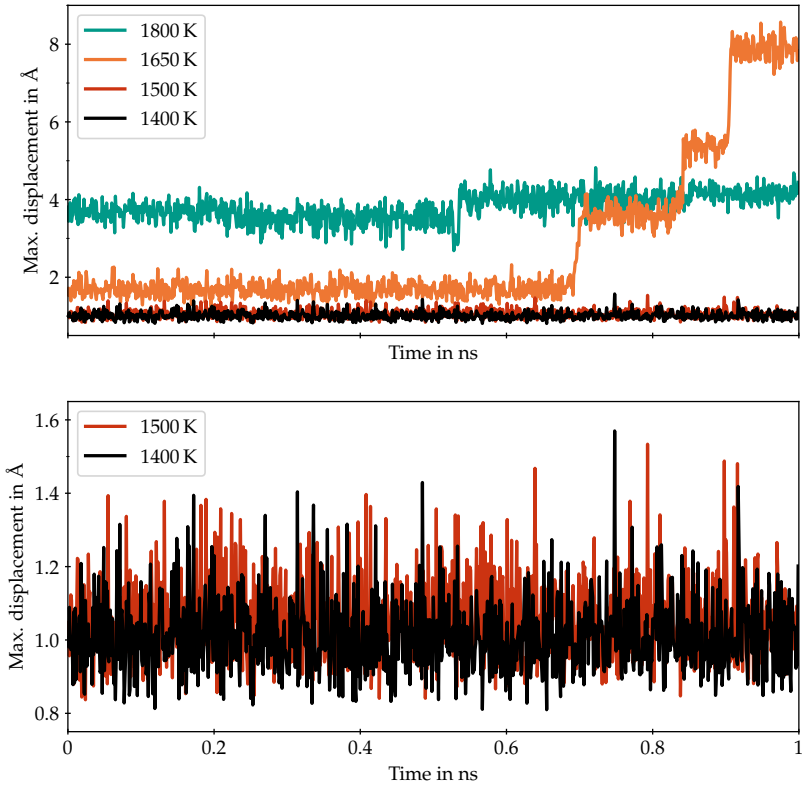


Figure D.7: Maximum displacement across oxygen atoms with respect to average positions over the course of a single trajectory at different temperatures in the tetragonal phase, computed with the M=2 and r_c=5.0 Å production SchNet MLIP. The top plot shows the full range of data, while the lower displays details.

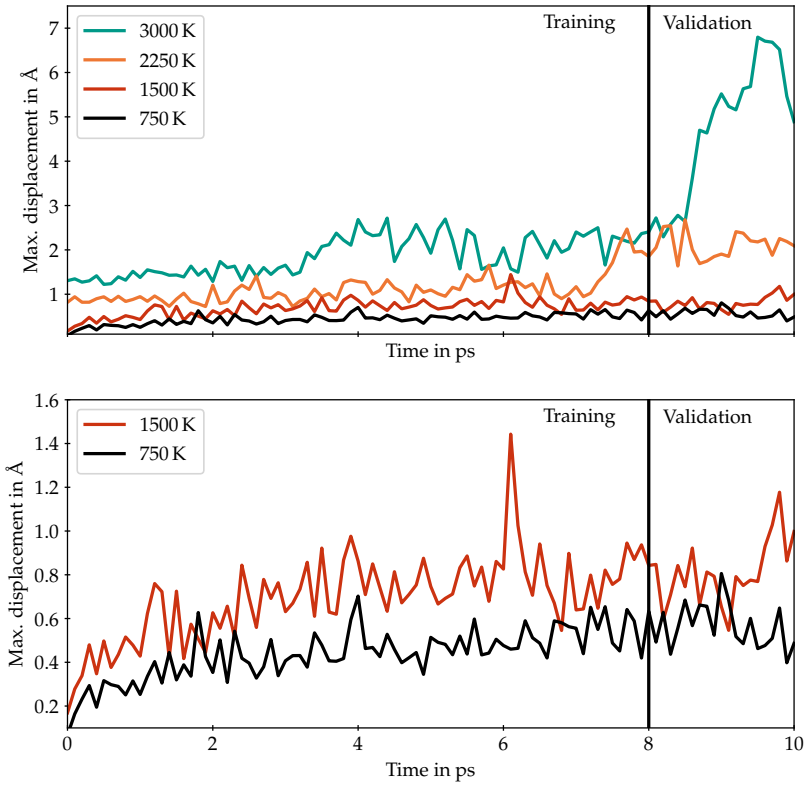


Figure D.8: Maximum displacement across oxygen atoms with respect to average positions for the trajectories in the training data, computed with `FHI-aims`. The top plot shows the full range of data, while the lower displays details.

D.3 Green-Kubo Convergence

Noise Removal

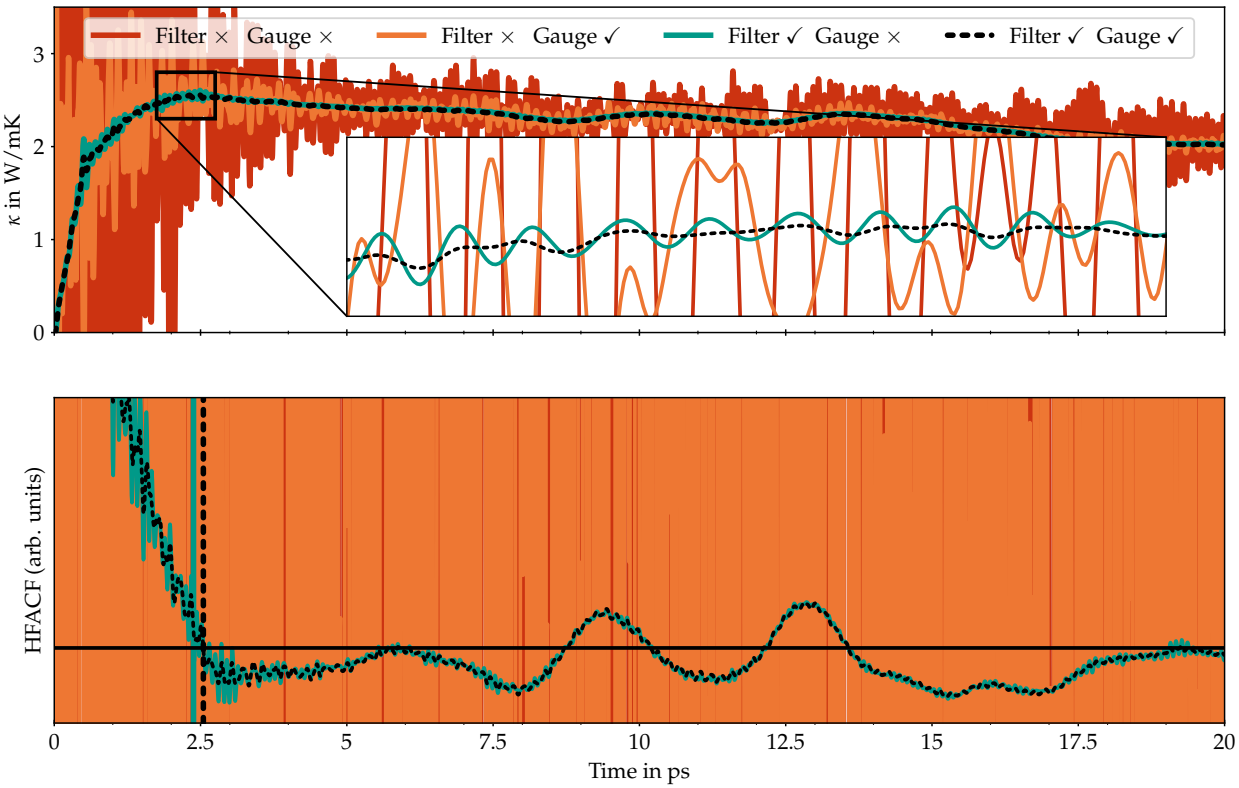


Figure D.9: $\kappa(\tau)$ (top) and $C(\tau)$ (bottom) for monoclinic zirconia at 300 K for 'unconverged' parameter choices (96, 0.1 ns), comparing different components of the noise reduction approach. The vertical lines in the bottom plot indicate the cutoff time.

Figure for 'production' settings in figure 5.2.4.

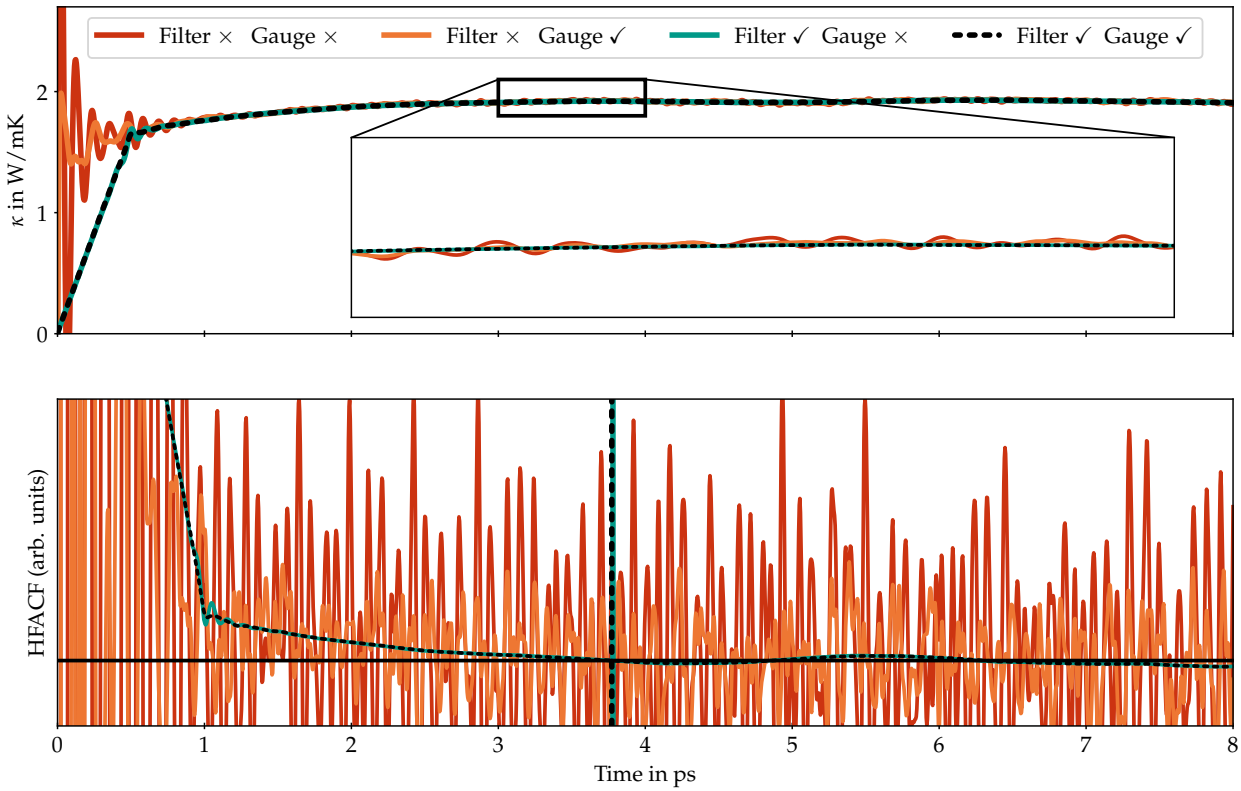


Figure D.10: $\kappa(\tau)$ (top) and $C(\tau)$ (bottom) for monoclinic zirconia at 1400 K for 'production' parameter choices (1500, 1 ns), comparing different components of the noise reduction approach. The vertical lines in the bottom plot indicate the cutoff time.

Filter Frequency

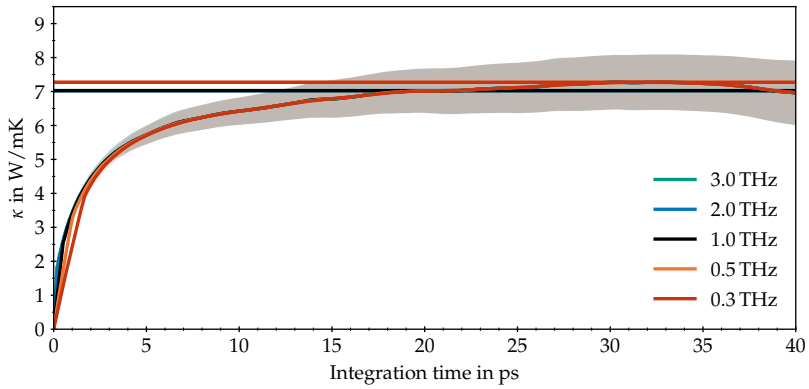


Figure D.11: $\kappa(\tau)$ and κ as determined from the HFACF for different choies of ω_{filter} for monoclinic zirconia at 300 K with 'production' settings (1500, 1 ns). Shaded areas indicate standard error over trajectories, lines the mean.

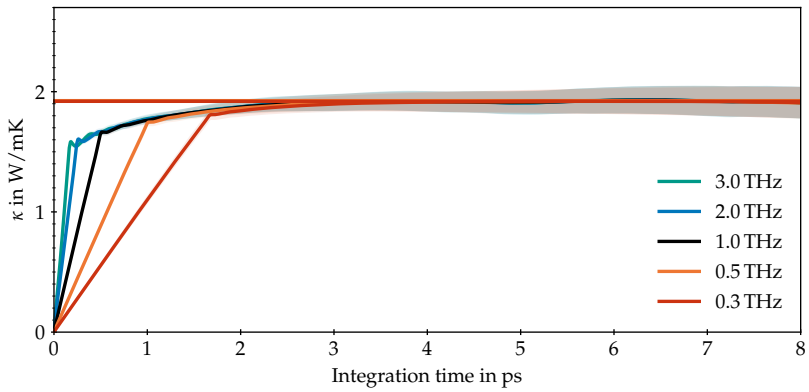


Figure D.12: $\kappa(\tau)$ and κ as determined from the HFACF for different choies of ω_{filter} for tetragonal zirconia at 1400 K with 'production' settings (1500, 1 ns). Shaded areas indicate standard error over trajectories, lines the mean.

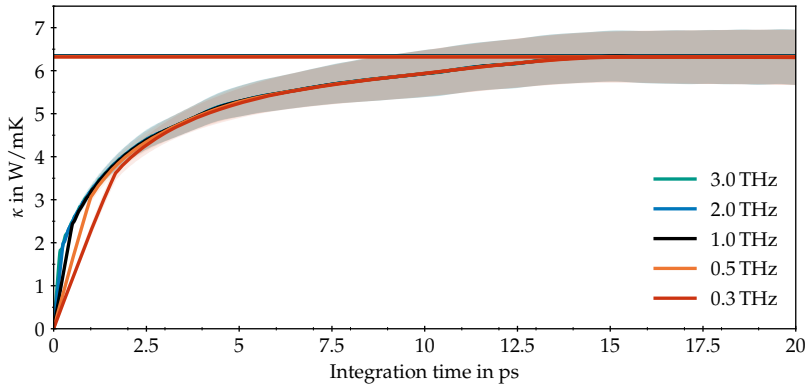


Figure D.13: $\kappa(\tau)$ and κ as determined from the HFACF for different choies of ω_{filter} for monoclinic zirconia at 300 K with 'light' settings (768, 0.5 ns). Shaded areas indicate standard error over trajectories, lines the mean.

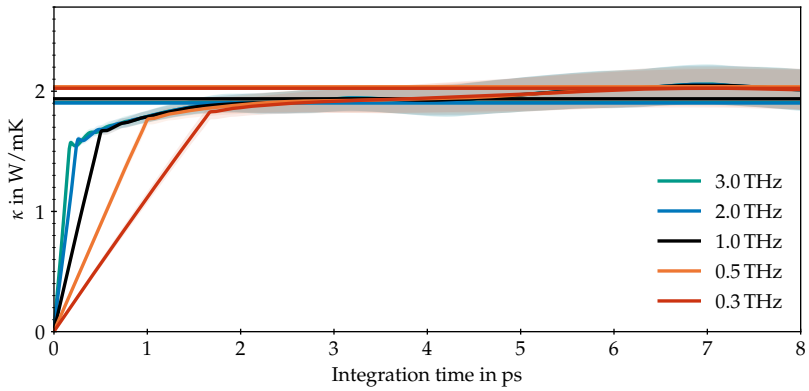


Figure D.14: $\kappa(\tau)$ and κ as determined from the HFACF for different choies of ω_{filter} for tetragonal zirconia at 1400 K with 'light' settings (768, 0.5 ns). Shaded areas indicate standard error over trajectories, lines the mean.

Spacing

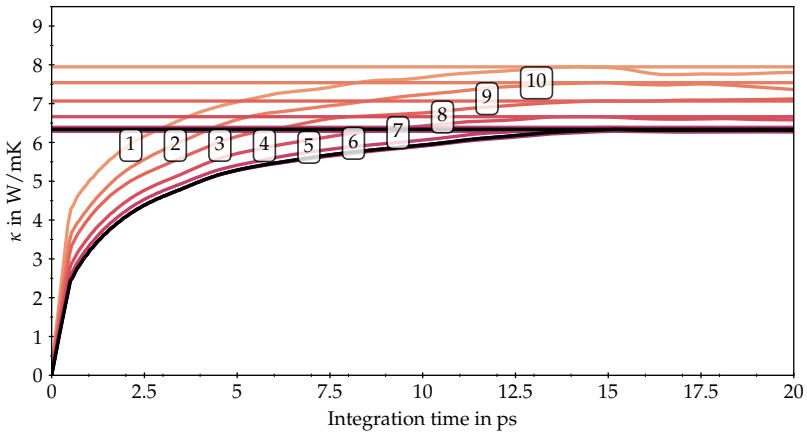


Figure D.15: $\kappa(\tau)$ and final κ for different choies of n_{hf} for monoclinic zirconia at 300 K with 'light' settings (768, 0.5 ns). The upper edges of the labels indicating n_{hf} are aligned with horizontal lines indicating the value for κ chosen by the first dip of the corresponding HFACF.

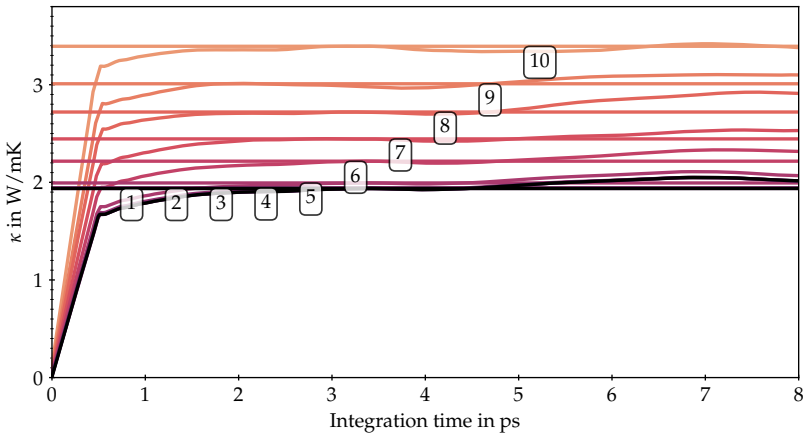


Figure D.16: $\kappa(\tau)$ and final κ for different choies of n_{hf} for monoclinic zirconia at 1400 K with 'light' settings (768, 0.5 ns). The upper edges of the labels indicating n_{hf} are aligned with horizontal lines indicating the value for κ chosen by the first dip of the corresponding HFACF.

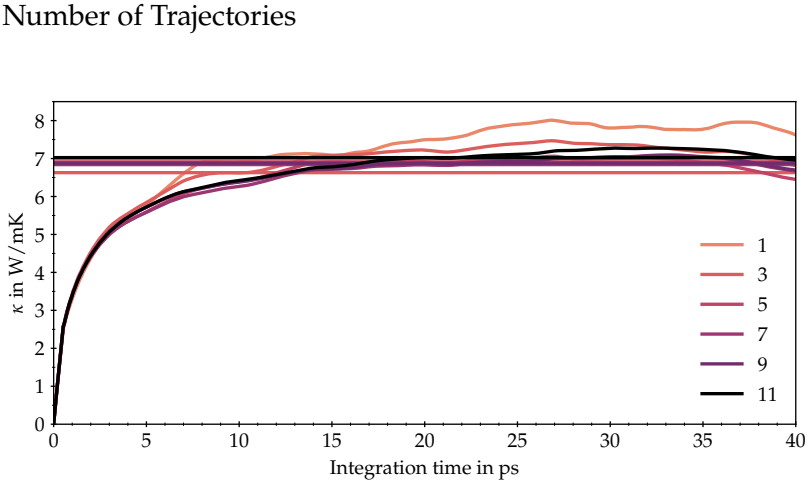


Figure D.17: $\kappa(\tau)$ and final κ for different number of trajectories n for monoclinic zirconia at 300 K with 'production' settings (1500, 1 ns). Horizontal lines indicate the value for κ chosen by the first dip of the corresponding HFACF.

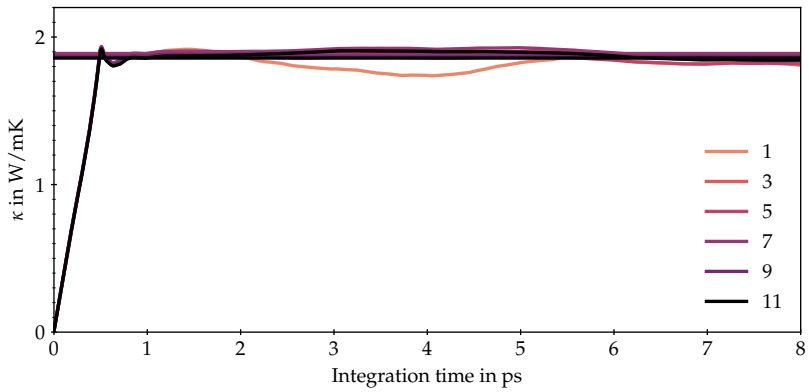


Figure D.18: $\kappa(\tau)$ and final κ for different number of trajectories n for tetragonal zirconia at 1400 K with 'production' settings (1500, 1 ns). Horizontal lines indicate the value for κ chosen by the first dip of the corresponding HFACF.

Size and Time

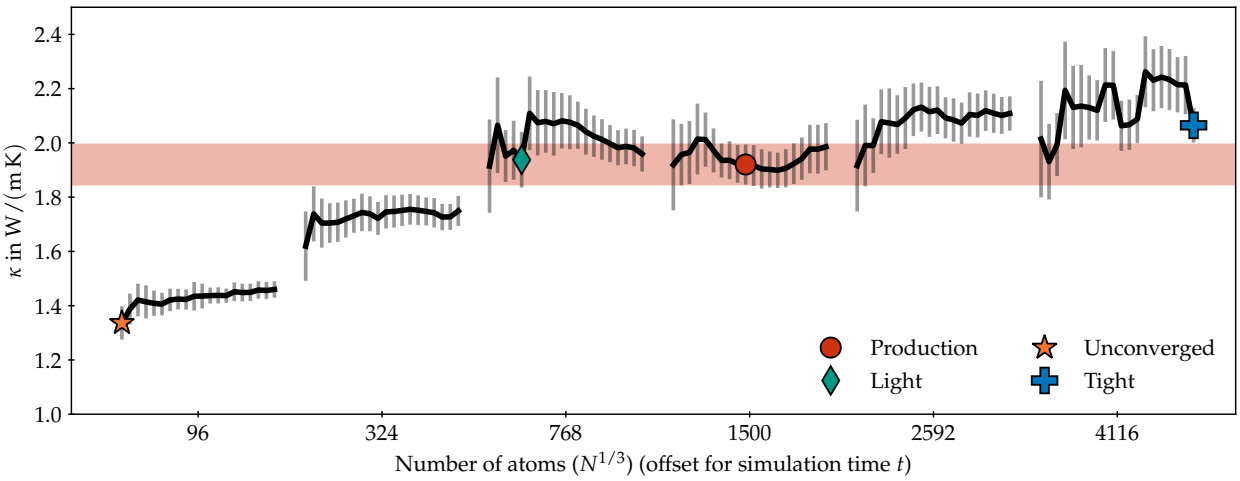


Figure D.19: κ for monoclinic zirconia at 1400 K for different choices of N and t_0. Error bars indicate the standard error across trajectories. N is shown as $N^{1/3}$, which is proportional to the length scale of the simulation cell. For each choice of N, t_0 from 0.1 ns to 2.0 ns are shown with a horizontal offset. 'Production', 'light', 'unconverged' and 'tight' choices are indicated; for the 'production' setting, the associated standard error is also shown as a shaded band.

Figure 5.2.5 shows 300 K.

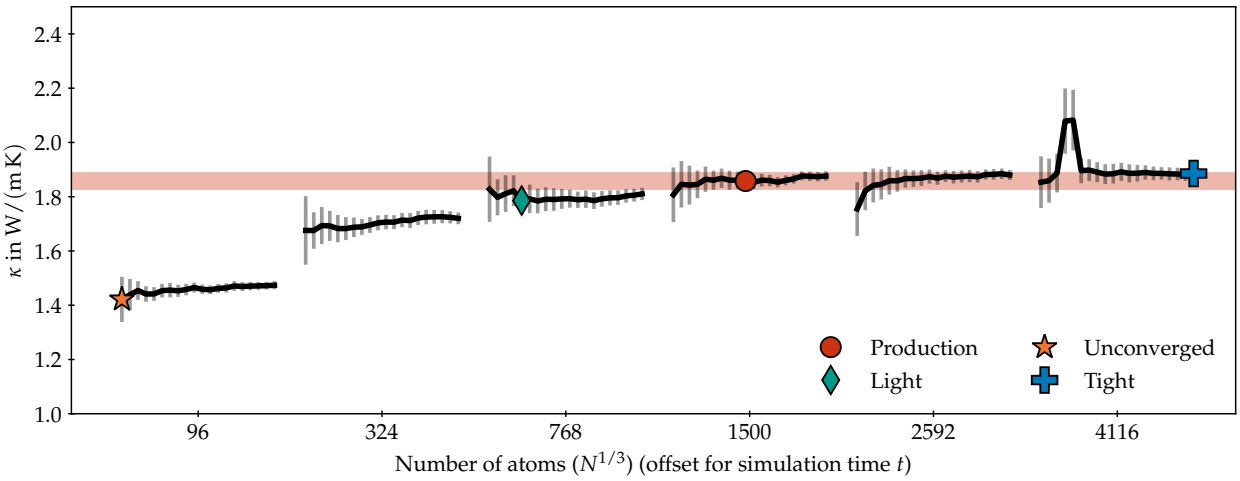


Figure D.20: κ for tetragonal zirconia at 1400 K for different choices of N and t_0. Error bars indicate the standard error across trajectories. N is shown as $N^{1/3}$, which is proportional to the length scale of the simulation cell. For each choice of N, t_0 from 0.1 ns to 2.0 ns are shown with a horizontal offset. 'Production', 'light', 'unconverged' and 'tight' choices are indicated; for the 'production' setting, the associated standard error is also shown as a shaded band.

The outliers at low t_0 for N=4116 in figure D.20 are due to the HFACF narrowly avoiding a zero crossing that emerges at higher t_0. It highlights that results for singular choices of N and t_0 must be carefully investigated, as they may emerge from such an artefact. The impact of such events can be reduced by choosing t_c not as the first zero crossing, $C(t_c) = 0$, but rather at a small positive number, ϵ_c, such that $C(t_c)=\epsilon_c$. In this thesis, this is not done to avoid an additional parameter in the GK method.

Alternatively, more sophicistaced methods for the determination of t_c can be employed, for instance by fitting a functional form [473, 475]. κ can also be determined by averaging over $\kappa(\tau)$ after t_c [371], or via spectral methods [476].

D.4 Gauge Term

As shown by Knoop et al. [144], it is advantageous to subtract a non-diffusive 'gauge term' from the instantaneous heat flux $\boldsymbol{J}(t)$ to reduce noise. This term takes the form

$$\boldsymbol{J}^{\text{gauge}} = \sum_{i \in \mathcal{R}_{\text{sc}}} \boldsymbol{\Omega}_i \cdot \boldsymbol{v}_i \tag{D.1}$$

with a constant 3×3 matrix $\boldsymbol{\Omega}_i$ for each atom, which is computed from the terms preceding the velocities in $\boldsymbol{J}_{\text{pot}}$ in equation (4.2.15). As $\boldsymbol{\Omega}_i$ is constant, $\boldsymbol{J}^{\text{gauge}}$ is a total time derivative, and by the gauge principle, does not contribute to the final thermal conductivity. However, as seen in section 5.2.3, removing this term from $\boldsymbol{J}$ can reduce noise.

In the original formulation, the 'virials'[1] are computed as time averages over the trajectory,

$$\boldsymbol{\Omega}_i^{\text{original}} = \langle \boldsymbol{r}_{ji} \otimes \frac{\partial U_i}{\partial \boldsymbol{r}_j} \rangle \ . \tag{D.2}$$

As has been discussed in section 4.2, an efficient heat flux formulation for GLPs does not necessarily yield $\partial U_i / \partial \boldsymbol{r}_j$ at every timestep, as the heat flux is directly computed with AD. Instead, we compute $\boldsymbol{\Omega}_i$ once for a reference structure, and in particular the supercell for a given temperature and phase as obtained in appendix D.6. In this formulation, $\boldsymbol{J}_{\text{pot}}$ and $\boldsymbol{J}_{\text{int}}$ can be treated interchangeably, assuming that the reference structure used for $\boldsymbol{r}^0$ is the one used for $\boldsymbol{\Omega}_i$.

These two approaches are compared in figures D.21 and D.22. No significant differences between both methods are observed. Indeed, if filtering is applied (not shown), any remaining differences vanish entirely.

[1] We note in passing that in this definition, the terms $\boldsymbol{\Omega}_i$ do not add up to the total stress in all cases.

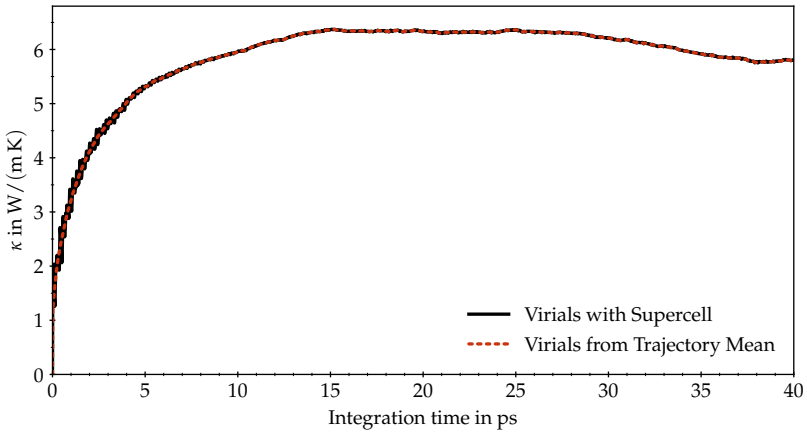


Figure D.21: $\kappa(\tau)$ for differently computed gauge terms for monoclinic zirconia at 300 K with 'light' settings (768, 0.5 ns). No filtering is applied. Heat flux is computed every n_{hf}=2 simulation steps.

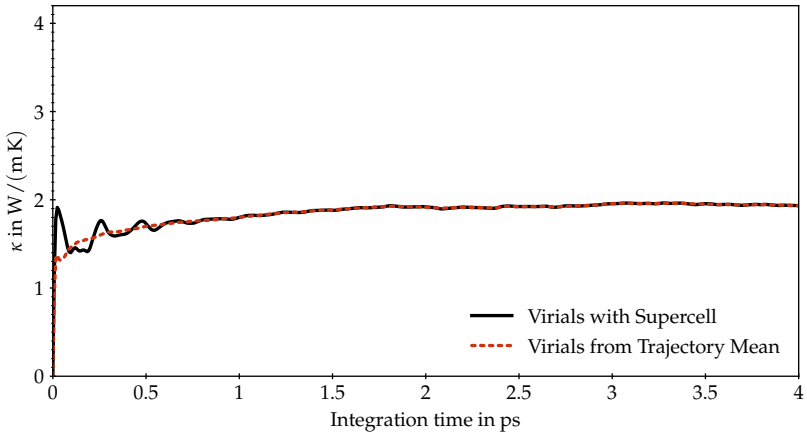


Figure D.22: $\kappa(\tau)$ for differently computed gauge terms for monoclinic zirconia at 1400 K with 'light' settings (768, 0.5 ns). No filtering is applied. Heat flux is computed every n_{hf}=2 simulation steps.

D.5 Potential and Convective Flux

In section 4.2.5, we discussed the distinction between the decomposition of the total heat flux into

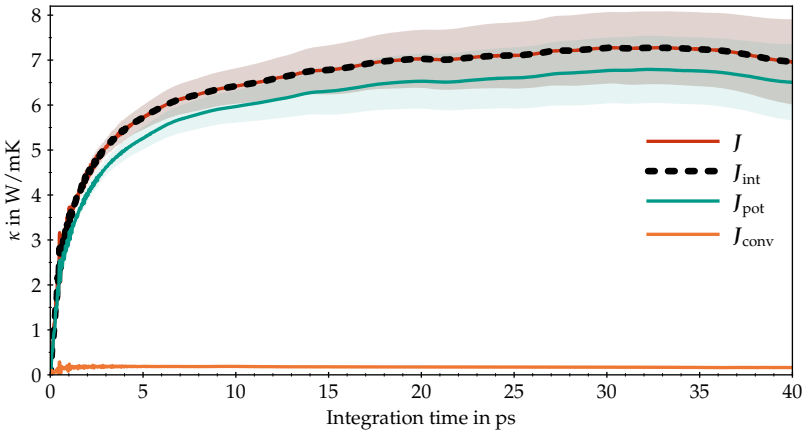
$$\boldsymbol{J} = \boldsymbol{J}_{\text{pot}} + \boldsymbol{J}_{\text{conv}} \tag{D.3}$$

and

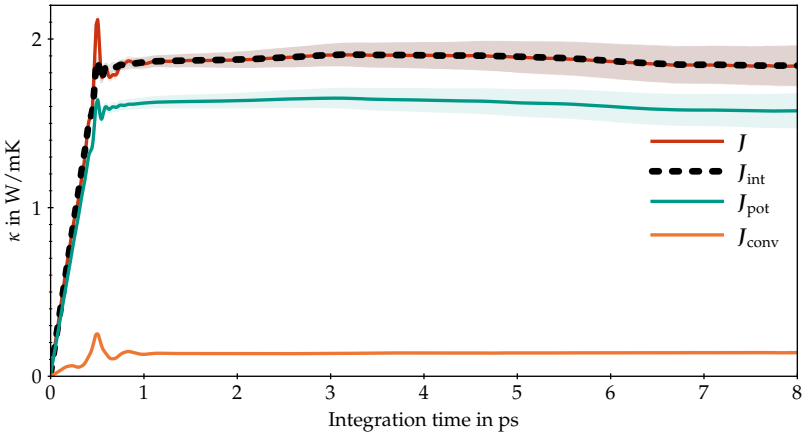
$$\boldsymbol{J} = \boldsymbol{J}_{\text{int}} + \boldsymbol{J}_{\text{disp}}\,, \tag{D.4}$$

finding that $\boldsymbol{J}$ and $\boldsymbol{J}_{\text{int}}$ should be equivalent if atomic positions are bounded over the course of the simulation. This is verified for zirconia in figure D.23; $\boldsymbol{J}_{\text{int}}$ and $\boldsymbol{J}$ are found to be equivalent, while $\boldsymbol{J}_{\text{pot}}$ is not. $\boldsymbol{J}_{\text{conv}}$ is non-vanishing, but yields a significantly smaller thermal conductivity on its own. At increased temperature, where atoms become more mobile, the magnitude of $\boldsymbol{J}_{\text{conv}}$ and hence the difference between $\boldsymbol{J}$ and $\boldsymbol{J}_{\text{pot}}$ becomes more pronounced.

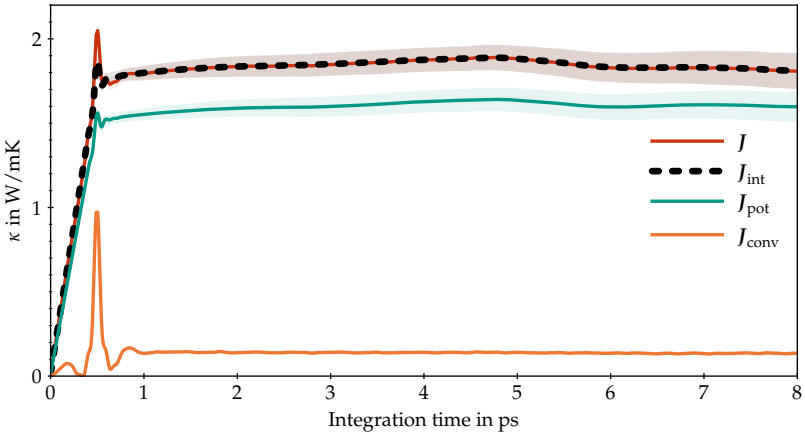
We can therefore conclude that $\boldsymbol{J}_{\text{int}}$ can be used in place of $\boldsymbol{J}$ for zirconia at the investigated range of temperatures and phases.



(a) Monoclinic, 300 K



(b) Tetragonal, 1400 K



(c) Tetragonal, 1800 K

Figure D.23: $\kappa(\tau)$ for different terms in the heat flux for zirconia at different temperatures and in different phases, using 'production' settings (1500, 1 ns). Filtering with ω_{filter}=1 THz is applied; no gauge term is removed.

D.6 Lattice Vectors and Supercells

As discussed in section 5.1, the employed MLIP displays limited accuracy when predicting lattice constants and treating the monoclinic to tetragonal phase transition. Since the focus of the present work is the heat flux, and since results are compared to experimental measurements, experimental lattice parameters are therefore used. The used lattice parameters, as presented by Verdi et al. [163], based on references [420, 423], can be found in tables D.1 and D.2.

The workflow to obtain supercells at a given temperature T and number of atoms N is as follows: The starting point is always a 12-atom unit cell. Lattice parameters are applied to this original cell, leaving atomic positions in fractional coordinates unchanged. From the resulting cell, simple $k \times k \times k$ supercells are constructed such that $N = 12k^3$.

Temperature in K	a in Å	b in Å	c in Å	β in °
300	5.147	5.209	5.311	99.30
350	5.150	5.209	5.315	99.28
400	5.152	5.211	5.318	99.27
450	5.154	5.211	5.322	99.26
600	5.161	5.213	5.330	99.17
750	5.165	5.213	5.340	99.07
900	5.172	5.215	5.349	98.99
1050	5.179	5.216	5.356	98.86
1200	5.193	5.237	5.386	98.74
1400	5.196	5.219	5.388	98.69

Table D.1: Lattice parameters for monoclinic zirconia based on references [163, 420, 423].

Temperature in K	a in Å	b in Å
1400	5.149	5.278
1500	5.155	5.284
1650	5.165	5.295
1800	5.174	5.305

Table D.2: Lattice parameters for tetragonal zirconia based on references [163, 420, 423].

D.7 Additional Results for Zirconia

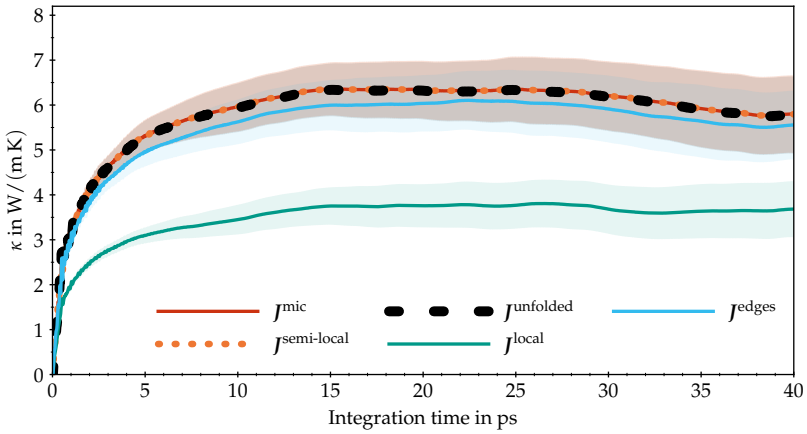


Figure D.24: $\kappa(\tau)$ for all heat flux formulations for monoclinic zirconia at 300 K, using a SchNet GLP ($M=2$, r_c=5 Å). See figure 5.3.1 for details.

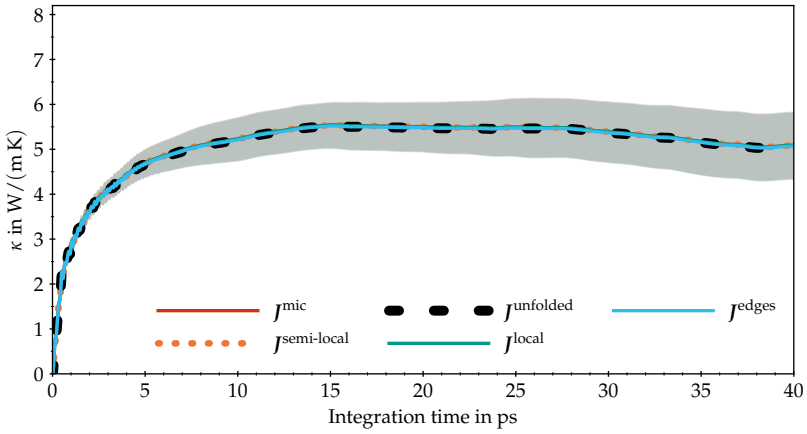


Figure D.25: $\kappa(\tau)$ for all heat flux formulations for monoclinic zirconia at 300 K, using a SchNet GLP ($M=1$, r_c=5 Å). See figure 5.3.3 for details.

D.8 Measurements of Thermal Conductivities of Zirconia

A number of experimental measurements of the thermal conductivity of ZrO_2 have been undertaken. [424–426, 477] In all cases, κ is determined indirectly, based on measurements of the thermal diffusivity α, the specific heat capacity c, and the density ρ, as $\kappa = \alpha\rho c$.

Hasselman et al. [477] investigate polycrystalline samples with varying MgO (magnesia) and Y_2O_3 (yttria) content at room temperature, estimating results for pure samples in different phases with an ansatz based on composite theory. For the tetragonal phase, $\kappa = 4.82\,\mathrm{W/mK}$, and for the monoclinic phase $\kappa = 5.2\,\mathrm{W/mK}$, indicating that different phases, at least at room temperature, display comparable thermal conductivities. As this study does not use pure samples, it has not been included in figure 5.3.4.

Raghavan et al. [424] measure pure and yttria-stabilised ZrO_2 nanoparticles in the monoclinic phase from 100 °C to 1000 °C. They observe a strong, approximately $1/T$, temperature dependence of κ for the pure sample, but not for samples with significant yttria content.

Bisson et al. [425] study pure and yttria-stabilised single crystals at room temperature, measuring $\kappa = 8.2\,\mathrm{W/mK}$ for pure monoclinic zirconia.[2] The higher value for κ obtained here may be attributed to a different methodology for determining α, which is less sensitive to grain boundaries. No significant anisotropy is found for κ.

Finally,[3] Mévrel et al. [426] determine κ for pure (monoclinic) and yttria-stabilised (tetragonal, cubic) single crystals from room temperature to 1100 °C. An amended model[4] for the temperature dependence of κ is also given.

MEASUREMENT RESULTS FOR the thermal conductivity of ZrO_2 are included in figure 5.3.4, experimental setups are tabulated in table D.3. At low temperatures, references [424–426] report similar values. Hasselman et al. provide a lower estimate, which is, however, not based on direct measurement, and therefore not included. As temperature increases, Mévrel et al. measure a more rapid decrease in κ compared to Raghavan et al., despite performing measurements on single crystals. The reason for this behaviour is unclear, as polycrystalline samples are expected to display reduced thermal conductivity due to phonon scattering at grain boundaries.

The standard method for measuring thermal diffusivity, used in references [424, 426, 477], is the 'laser flash technique', where one end of a sample is briefly heated, and the temperature change at the opposite end is observed. However, the method probes a large area of the sample, which may include imperfections that can hinder thermal transport, prompting the development of alternative techniques, such as the 'spatially resolved infrared thermography' method used in [425].

[2] An uncertainty of 'the order of 10 %' is mentioned.

[3] Youngblood et al. [478] is not included in the summary, as the work does not consider undoped zirconia. We also exclude studies of porous samples.

[4] The model, based on the assumption that the phonon mean free path is bounded from below by the minimum neighbour distance, gives $\kappa(T) = A/T\left[2\sqrt{T_1}/3\sqrt{T} + T/3T_1\right]$ with fit parameters A and T_1.

	Phase	Single	Pure	$\kappa(T)$
[477]	t,m	×	×	×
[424]	m	×	✓	✓
[425]	m	✓	✓	×
[426]	m	✓	✓	✓

Table D.3: Experiments measuring κ for pure ZrO_2. *Single* = single crystal samples used, *pure* = undoped sample directly measured, $\kappa(T)$ = measured temperature dependence.

D.9 Tin Selenide

Testing

Property	RMSE	MAE	maxAE	R^2 in %
Energy in meV	94.5029	73.6968	307.2220	95.8476
Forces in meV/Å	52.1557	41.3036	321.0409	97.4306

Table D.4: Test set errors for So3krates for SnSe, with M=1.

Property	RMSE	MAE	maxAE	R^2 in %
Energy in meV	28.2936	22.4426	78.1010	99.6278
Forces in meV/Å	19.7130	15.5923	148.4874	99.6329

Table D.5: Test set errors for So3krates for SnSe, with M=2.

Property	RMSE	MAE	maxAE	R^2 in %
Energy in meV	17.0437	13.4452	47.9379	99.8649
Forces in meV/Å	13.5658	10.7136	85.0041	99.8262

Table D.6: Test set errors for So3krates for SnSe, with M=3.

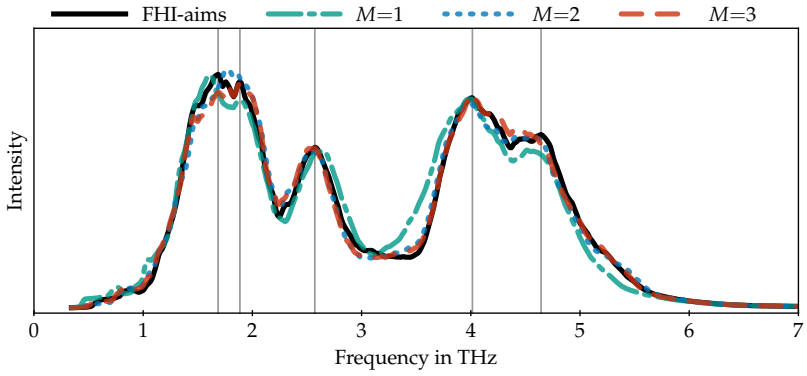


Figure D.26: Comparison of VDOS in SnSe for So3krates with varying values of M compared to a baseline computed with `FHI-aims`. Vertical lines indicate peaks in the `FHI-aims` result. Results are shown for one trajectory of 30ps (Δt=4 fs) at 300 K in a supercell containing 256 atoms, with matching initial configurations.

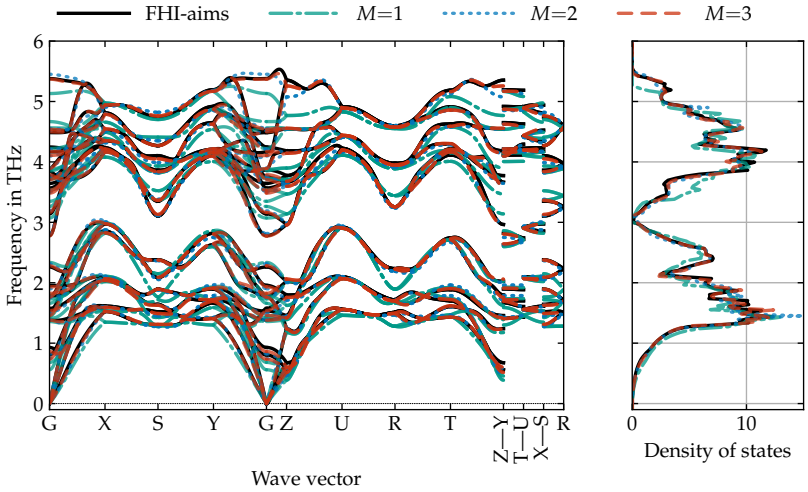


Figure D.27: Comparison of phonon band structure and density of states in SnSe for So3krates with varying values of M compared to a baseline computed with `FHI-aims`. Results are showsn for a supercell containing 256 atoms, with matching lattice and positions relaxed to the 0 K configuration.

Green-Kubo Convergence

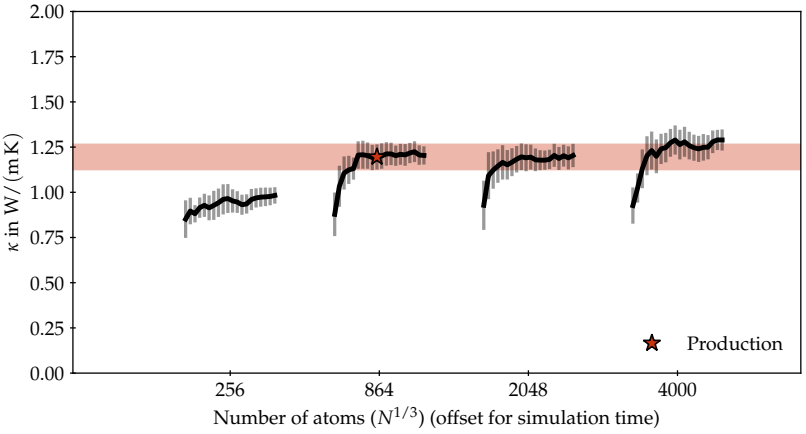


Figure D.28: κ for tin selenide at 300 K for different choices of N and t_0 using a So3krates model with M=2. Error bars indicate the standard error across trajectories. N is shown as $N^{1/3}$, which is proportional to the length scale of the simulation cell. For each choice of N, t_0 from 0.1 ns to 4.0 ns are shown with a horizontal offset. 'Production' settings are indicated, the associated standard error is also shown as a shaded band.

D.10 Silicon

Property	RMSE	MAE	maxAE	R^2 in %
Energy in meV	92.4688	71.8210	295.3312	99.9784
Forces in meV/Å	45.5592	35.1807	330.7921	99.7029
Stress in kbar	0.8973	0.6349	2.5944	99.8320

Table D.7: Test set errors for So3krates for Si, with M=1.

Property	RMSE	MAE	maxAE	R^2 in %
Energy in meV	42.3110	33.1063	153.6074	99.9955
Forces in meV/Å	23.6281	18.0433	274.2255	99.9201
Stress in kbar	0.9403	0.6905	2.2113	99.8155

Table D.8: Test set errors for So3krates for Si, with M=2.

Property	RMSE	MAE	maxAE	R^2 in %
Energy in meV	34.6730	25.9488	150.2103	99.9970
Forces in meV/Å	16.7011	12.6485	142.6790	99.9601
Stress in kbar	0.9720	0.6784	2.9101	99.8029

Table D.9: Test set errors for So3krates for Si, with M=3.

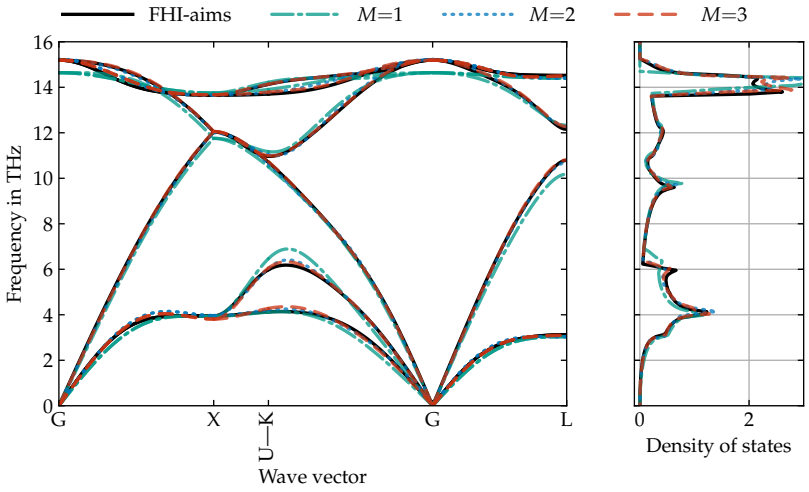


Figure D.29: Comparison of phonon band structure and density of states in Si for So3krates with varying values of M compared to a baseline computed with `FHI-aims`. Results are showsn for a supercell containing 216 atoms, with matching lattice and positions relaxed to the 0 K configuration.

Appendix E
Software

Nothing is yours. It is to use. It is to share.
If you will not share it, you cannot use it.

– Ursula K. LeGuin, *The Dispossessed*

The work presented in this thesis required the development of a number of different software libraries, which are presented briefly in this appendix. All software has been made publicly available under permissive open source licences.

E.1 cmlkit

`cmlkit` [479] is a framework developed for the benchmark presented in section 3.3. It implements a general architecture for constructing ML models for atomistic problems that are composed of a representation and a regressor. Components of such models are represented as stateless objects, which can be descibes as `dicts`. An interface to the `hyperopt` package, which implements TPE-based HP optimisation, was implemented, as well as interfaces to `RuNNer` [480] (for SFs), `quippy` [481] (for SOAP), `qmmlpack` [482] (for MBTR), as well as `dscribe` [276].

E.2 Green-Kubo Pytorch Infrastructure

As we have seen in section 5.2, studying convergence in the GK method requires large-scale simulations, beyond the ability of `FHI-vibes`. In order to enable the simulations performed in this thesis, I consequently adapted `FHI-vibes` and its underlying infrastructure.

In `FHI-vibes`, MD simulations are logged into the `.son` file format, which consists of a series of `json` dictionaries separated by a particular delimiter. For post-processing, this format is then converted into `xarray` datasets backed by `netcdf`. This post-processing backend, originally developed for aiMD, was unable to process the files produced by MLIP MD, which generated hundreds of TiB of data. In order to process datasets of this size, some changes were made:

- `son`, the implementation of the `.son` format, was re-written entirely to allow for stream-based processing, rather than loading full trajectories into memory. This rewrite is used by `FHI-vibes`.

- I developed an alternative post-processing backend for `FHI-vibes`, code-named `stepson`, which relies on `dask` to perform parallel chunked post-processing of `son` trajectories. This allows the processing of datasets far exceeding the available memory. `stepson` is planned be included in a future release of `FHI-vibes`.

Additionally, I developed extensions for `schnetpack` [391] to implement the heat flux formulations from section 4.2, and to enable efficient MD with `FHI-vibes`, available at `https://github.com/sirmarcel/gknet-archive`.

E.3 Green-Kubo JAX Infrastructure

In order to use So3krates models, which are implemented in `mlff` [427], which in turn relies on `jax` [37], new infrastructure was required. I therefore developed `glp` [410], which implements the heat flux formulations from section 4.2, as well as the stress formulas from section 4.1, for any potential of the GLP type, i.e., mapping a graph of atom-pair vectors to a set of atomic potential energy contributions. To efficiently run MD with GLP, I also wrote `gkx` [483], which runs NVE MD entirely on the GPU, or potentially other accelerator devices, and directly emits `netcdf` datasets, rather than requiring `.son` files.