Optimising machine-learning interatomic potentials for the study of disordered functional materials

Daniel Thomas du Toit

St. Anne's College

A thesis submitted to the University of Oxford for the degree of Doctor of Philosophy

Inorganic Chemistry Laboratory

University of Oxford

Trinity 2025

Declaration

The work presented in this thesis was carried out between October 2021 and July 2025 in the Inorganic Chemistry Laboratory, University of Oxford under the supervision of Prof. Volker L. Deringer. This dissertation is the result of my own original work, and where it draws on the work of others, or of previous work of my own, this is acknowledged at appropriate points in the text and the acknowledgements section in each chapter. This dissertation has not been submitted in whole or in part for a degree at this or any other institution.

Daniel F. Thomas du Toit

Abstract

Machine-learning interatomic potentials (MLIPs) are a powerful tool for studying the properties of materials via atomistic simulation. The modelling of dynamical processes and disordered materials requires large time- and length-scale simulations for modelling relevant to the processes as observed in experiments, making MLIPs the best choice for accurate simulations.

This thesis explores how the application of hyperparameter optimisation to MLIPs improves our ability to systematically improve the efficiency and accuracy of ML studies of processes and structure in materials. I use Bayesian optimisation (BO) to fit MLIPs optimised for a range of materials, and develop XPOT, a cross-platform optimiser for MLIPs, to automate this hyperparameter optimisation and apply it to not only a range of disordered materials, but a range of model architectures too.

First, I introduce XPOT and benchmark it by comparing my models against existing models from a benchmark study in literature. This proof of concept study is used to understand the numerical performance of models fitted by XPOT, and prove the viability of the method. I then investigate the use of different validation datasets and weighting of performance criteria to understand the relationship between the accuracy of energy and force predictions in MLIPs.

Secondly, I move on to a state-of-the-art model architecture, the Atomic Cluster Expansion (ACE). Linear and non-linear ACE models are computationally efficient, but demonstrate impressive accuracy despite this, being used for a range of large-scale materials simulations. I use XPOT to optimise ACE models for disordered material systems, and perform physical validation of my models. By optimising Atomic Cluster Expansion (ACE) potentials, this work enabled large-scale, long-timescale molecular dynamics simulations which are key to understanding the processes in disordered materials. XPOT-optimised ACE potentials are fitted for amorphous silicon (a-Si) and the phase-change material antimony telluride (Sb_2Te_3), and discuss the importance of both numerical and physical validation of the optimised models for use in simulation. Next, I apply these techniques to the technologically important Ge–Sb–Te and elemental Te systems, revealing details of crystallisation processes in phase-change memory devices which were previously computationally inaccessible.

Building on these results, I use an optimised MLIP to improve the structural model of $Li_{14}SiP_6$, showing with molecular dynamics (MD) that the Li-ion site disorder behaves differently to the structure reported in literature. I use an optimised MLIP to study the Li-ion diffusion process, a key property for solid-state electrolytes.

Finally, I explore the application of knowledge distillation, leveraging foundation models to generate efficient and accurate potentials for complex systems like liquid water, and study the effects of hyperparameter optimisation on the learning of the student model.

In conclusion, this thesis establishes that systematic hyperparameter optimisation is a key tool for reliable and efficient MLIP studies. The tools and methodologies presented herein provide a systematic pathway to creating high-quality potentials, enhancing the understanding of complex functional materials.

Acknowledgements

First, I would like to thank my supervisor, Volker Deringer, for his guidance and support throughout not only my DPhil, but my entire research career to date. I hold many fond memories of starting our research together during COVID, and I was lucky to be able to start my research career under your tutelage. Your mentorship has been invaluable, and your unwavering support for my research, alongside your passion for the science we do, has been a source of great inspiration to me.

I cannot thank the members both past and present of the Deringer group enough for making my four years so enjoyable. First, to Yuxing: Your enthusiasm is infectious, and you've never failed to make my day better in the office; Tom, your patience when I first joined the group will never be forgotten, and I'm delighted that I got the opportunity to contribute to your work on ZIFs. Joe, your enthusiasm for science, and your unending supply of interesting facts are much missed from the lab. John and Zoe, you've been the most fantastic friends I could ask for, pulling me up when I was at my lowest ebb, and I'll miss walking in to the lab to you both each day. Zak, thank you for showing me around Berlin, and all the fun we've had along the way. Louise, a confidant, a friend, and so much more; thank you for making the three years we've been here together so enjoyable. Natascia, our ramblings have been a source of great joy. Chiheb, your guidance and one-liners were brilliant in equal measure. To Andy, Bianca, and Litong - thank you for being such fantastic colleagues and friends, and for the fun we've shared. Finally, to Kristian, Joe, Aneta, Theo, Nim, Hunter, Max, and all the other past and present members of the group, thank you for being such a fantastic group of people to work, chat, and laugh with.

To those who've stuck by me through it all: I couldn't have done this without you. Zach and Zoe, you'll never know just how much your friendship means. To sharing the best moments, and being there in the worst, I couldn't ask for more. Bogdan, Petrut, Marek, and Morten: for not only the best holidays, but many wonderful evenings. Your support through the ups and downs over the last 3 years has been incredible, and I hope to see you all at another racetrack soon. Phil, for the fun we had, despite the distance - I'm looking forward to being able to see each other more. I also want to thank my family for their unwavering belief in me throughout.

Finally, Alexa. You've made the final year of my DPhil the most enjoyable yet. I cannot imagine having done this without you, and I'm so excited for what comes next.

Publications

The following publications contribute directly to chapters in this thesis:

[1] **D. F. Thomas du Toit**, V. L. Deringer*
Cross-platform hyperparameter optimization for machine learning interatomic potentials
J. Chem. Phys. **2023**, *159*, 024803

[2] **D. F. Thomas du Toit**, Y. Zhou, V. L. Deringer*
Hyperparameter Optimization for Atomic Cluster Expansion Potentials
J. Chem. Theory Comput. **2024**, *20*, 22, 10103–10113

[3] J. L. A. Gardner†, **D. F. Thomas du Toit**†, C. Ben Mahmoud, Z. Faure Beaulieu, V. Juraskova, L. B. Paşca, L. A. M. Rosset, F. Duarte, F. Martelli, C. J. Pickard, V. L. Deringer*
Distillation of atomistic foundation models across architectures and chemical domains
preprint at arXiv:2506.10956 **2025**

Additionally, I include the following publications which are discussed as applications of the work herein:

[4] Y. Zhou, **D. F. Thomas du Toit**, S. R. Elliott, W. Zhang*, V. L. Deringer*
Full-cycle device-scale simulations of memory materials with a tailored atomic-cluster-expansion potential
preprint at arXiv:2502.08393 **2025**

[5] Y. Zhou, S. R. Elliott, **D. F. Thomas du Toit**, W. Zhang*, V. L. Deringer*
The pathway to chirality in elemental tellurium
preprint at arXiv:2409.03860 **2024**

[6] T. C. Nicholas, **D. F. Thomas du Toit**, L. A. M. Rosset, D. M. Proserpio, A. L. Goodwin*, V. L. Deringer*
The structure and topology of an amorphous metal-organic framework
preprint at arXiv:2503.24367 **2025**

† Authors who contributed equally to the work.

* Corresponding authors.

Abbreviations

ACE - Atomic Cluster Expansion

AIMD - *ab initio* molecular dynamics

ANN - artificial neural network

ASE - atomistic simulation environment

BO - Bayesian optimisation

DFT - density functional theory

EAM - embedded atom model

FM - foundation model

FSDP - first sharp diffraction peak

GAP Gaussian approximation potential

GNN - graph neural network

LAMMPS - Large-scale Atomic/Molecular Massively Parallel Simulator

LDA - low density amorphous

MAE - mean absolute error

MD - molecular dynamics

ML - machine learning

MLIP - machine learning interatomic potential

NMR - nuclear magnetic resonance

NN - neural network

OTS - ovonic threshold switching

PCM - phase change material

PES - potential energy surface

QM - quantum mechanics

RDF - radial distribution function

RMSE - root mean square error

SNAP - spectral neighbour analysis potential

SOAP - smooth overlap of atomic positions

SSE - solid-state electrolyte

VHDA - very high density amorphous

XPOT - cross-platform optimiser for machine leaning potentials

ZBL - Ziegler–Biersack–Littmark (ZBL) screened nuclear repulsion

Contents

Chapter 1

Introduction

In the 21st century, artificial intelligence (AI) and machine learning (ML) have evolved from niche academic pursuits into mainstream, transformative technologies used in our everyday lives (for better and for worse).[7–9] From generative models for daily use[10] to autonomous coding agents capable of writing ready-to-ship applications[11] and self-driving systems,[12] AI's meteoric rise has been driven by an unprecedented growth in data availability and computational power.[13,14] However, it is not only directly that our daily lives that have been changed by ML techniques: ML has had an incredible impact across the sciences, powering new insights into structural biology, materials science, and our understanding of nature.[15–21]

Such is the scale of these breakthroughs, the Nobel Prize for Physics was awarded to John Hopfield and Geoffrey Hinton for their contributions to artificial neural networks (ANNs), the theoretical basis for the proliferation of AI in recent years. ML and the sciences have an inextricable history, with the creation of the Hopfield network (the grandfather of the ANN) being driven by his understanding magnetic moments, and how magnetic moments align in solids.[22] Further work by Hinton developed upon the Hopfield network, making use of statistical physics in the form of the Boltzmann equation to assign states to the nodes (biases) of the network, and bridging the gap between the seminal work by Hopfield and the modern ANN.[23,24]

In the same year, the Nobel Prize for Chemistry was awarded to the scientific team behind AlphaFold, an ML model which revolutionised structural biology by predicting protein folding patterns with unprecedented accuracy and efficiency.[15] These constitute the first Nobel Prizes awarded for ML, and underline the importance of ML in the sciences today. Against this backdrop, atomistic simulation has been transformed by the development of machine learning techniques to enable data-driven modelling of materials.

1.1 Atomistic simulation

The core objective of atomistic simulation is to accurately model the interactions between atoms, which are governed by the potential energy surface (PES): a high-dimensional function that dictates the forces on atoms given their positions. This potential energy surface drives the dynamics of a system. For over half a century, computational materials scientists have endeavoured to develop ever-improving approximations of this surface at ever-reducing computational cost.[25–27] The accurate description of interactions between atoms is fundamental to our understanding and prediction of material properties and chemical processes, but the practicalities of doing so have been a persistent challenge, and the trade-off between computational efficiency and accuracy has shaped the development of our estimations of the PES thus far, and continues to do so today.

The first molecular dynamics (MD) simulation was performed in 1953 to study the motion of a vibrating string by solving the equations of motion as a function of time for a many-body system, now known as the Fermi–Pasta–Ulam–Tsingou problem.[28] Just four years later, Alder and Wainwright simulated the dynamics of a liquid of hard spheres with elastic collisions, and in 1964 Rahman used a Lennard-Jones potential to model argon.[29] These early simulations using empirical (or "classical") interatomic potentials to simulate condensed matter provided the foundation for the field of atomistic simulation.

Empirical models employ a fixed functional form, usually derived from physical intuition, and the parameters are tuned per-material to minimise the discrepancy between data and simulation. However, as the functional form is rigid, and can only be affected by changing the relatively few parameters (when compared to machine learning models), the accuracy of the model upon a certain material is limited by the suitability of the functional form to describe the interactions in that material. Thus, different models have been created for different materials. Widely used empirical potentials include the aforementioned Lennard-Jones potential,[30,31] the Stillinger–

Weber potential for silicon,[32] and the Embedded Atom Model (EAM) for metals.[33,34] The computational efficiency of these interatomic potentials enabled simulations of millions of atoms with what are now considered modest computational resources. However, their reliance on a fixed functional form inherently limits their accuracy and transferability, rendering them unsuitable for describing complex chemical environments (such as those in amorphous phases), bond breaking and formation, or energetically subtle phase transitions.

To overcome the limitations of empirical models, first-principles, or *ab-initio*, methods grounded in quantum mechanics have been used to drive molecular dynamics simulations. These methods approach modelling the PES from a different angle. Instead of defining a fixed functional form with which to approximate the PES, these methods approximate the many-body Schrödinger equation for the system, taking into account the interactions of the atoms (and their electrons) within a structure, foregoing the need to make prior assumptions about bonding or the body-order of interactions in matter. Of importance to atomistic modelling of bulk materials is Density Functional Theory (DFT), which is based upon the electron density rather than the many-electron wave function.[35] The profound impact of DFT on chemistry and physics was recognised with the 1998 Nobel Prize in Chemistry awarded to Walter Kohn and John Pople.[36] DFT offers a far more fundamental and accurate description of the PES without prior assumptions about the nature of chemical bonding. As a result, *ab-initio* calculations made up over 50% of ARCHER-2 (the UK's largest supercomputer) usage in January 2022,[37] owing to ease-of-use, reliability, and accuracy for modelling materials.

However, the accuracy of DFT comes at a substantial computational cost, scaling as at least $\mathcal{O}(N^3)$ for conventional plane-wave methods, although linear scaling methods can be used in systems in which the density matrix exhibits locality. N is defined as size of the system, which is proportional to the number of valence electrons in the system. The need to solve the electronic structure for each atomic

configuration restricts DFT-driven molecular dynamics simulations to systems of a few thousand atoms and timescales measured in picoseconds, despite computational advances. This "quantum bottleneck" has long prevented the direct study of many critical phenomena, such as the behaviour of amorphous materials, diffusion in disordered solid-state electrolytes, and other processes that demand large system sizes and long simulation times. For decades, the materials modelling community was faced with a difficult choice: the efficiency of classical potentials or the accuracy of first-principles methods.

1.2 Machine learning interatomic potentials

Hailed as the "fourth paradigm" in science,[38] machine learning provides a data-driven approach to modelling, and, via machine learning interatomic potentials (MLIPs), atomistic simulation. With the increases in computational resources and quantity of DFT-driven MD simulations and individual structures available, the amount of high-quality, quantum-accurate data has increased dramatically. Rather than relying on a predefined functional form or solving quantum equations on the fly, MLIPs "learn" the complex relationship between atomic structure and energy from data, which can be generated by any method, but is often achieved by labelling representative structures with DFT for disordered solids, although other methods are often used dependent on the system being studied (e.g., coupled cluster methods for small molecular systems). This paradigm shift, which mirrors the broader data-driven revolution in AI, allows MLIPs to achieve quantum-level accuracy at a computational cost that is orders of magnitude lower, approaching that of empirical potentials.[6]

The existence of MLIPs has enabled the study of disordered materials inadequately described by empirical potentials at scales inaccessible to DFT. From investigating the structures of long-debated amorphous phases,[39–42] to the modelling of critical processes in functional materials,[43–47] ML-driven MD has helped to further our understanding of the atomistic structure of materials.

Early MLIPs using NNs were developed in the 1990s,[48–50] and the modern era of

MLIPs was ushered in by the work of Behler and Parinello in 2007, who extended the use of ANNs to model high-dimensional potential energy surfaces.[51] Soon after, Bartók *et al.* introduced the Gaussian Approximation Potential (GAP),[52] kicking off a wave of studies using these models.[53–55] The advent of MLIPs has enabled simulations of billions of atoms approaching quantum accuracy,[56–58] and catalysed research into complex processes in materials previously out of reach.[6,43,59] In the almost two decades since, the field has experienced a steady growth of new model architectures, with developments being made to descriptors (i.e., the way we represent the local atomic environment) and regression (i.e., the way we fit the PES in descriptor space).[51,60–66]

These developments have led to a rapid rise in the capability of MLIPs, not only in the accuracy of the predicted PES, but the scope of said PES that can be modelled by an individual model. One of the intrinsic problems with any interatomic potential (i.e., a mathematical function that describes the PES), whether empirical or machine learned, is that they must be parameterised before being used. For complex MLIPs, training a model is a non-trivial task, with large amounts of reference data and careful hyperparameter selection required to generate a model which can be used to accurately model the PES of a given chemical space.

In recent years, significant progress has been made in attempting to create a class of MLIPs which can be used to model the PES of a given chemical space without the need for developing large, bespoke datasets to train models from scratch. "Foundation models" are MLIPs which are trained upon vast datasets comprising materials from across the periodic table, aimed at becoming an out-of-the-box tool similar to DFT for modelling a material.[67–76] These models allow for zero-shot simulation, where a model can be used to simulate a material without any training from the user, although "fine-tuning" of a model to improve accuracy on the system of interest is often preferable.[67,77] However, despite the rapid growth of MLIPs, there are still many open questions to how to most efficiently and accurately model the

PES of chemical space, and drive scientific discovery with MLIPs.

Developing MLIPs

As MLIPs become mainstream simulation tools, the research focus is shifting from demonstrating feasibility to ensuring accuracy and generalisability, maximising efficiency, and developing standardised best practices. Improvements in architectures,[78–80] benchmarking,[81–84] training protocols,[85] and datasets[71,86,87] continue to improve the accuracy of MLIPs. To extract the true potential of current and future MLIPs, it is important to explore the entire fitting process, and find the best approaches and settings to fit the PES for any specific problem.

Automation of the fitting of MLIPs has been explored in literature, beginning with automating dataset generation,[88–91] and progressing to the recent interest in "end-to-end" fitting tools.[92–94]

A critical, yet under-explored, aspect of this process is hyperparameter optimisation.[95–97] The accuracy, cost, and robustness of an MLIP are sensitive to a number of user-defined choices made before training begins—such as the cutoff radius, the complexity of the basis set, the number of layers in a neural network, or indeed, the training protocol itself. In literature, these hyperparameters are often selected manually based on heuristics and previous experience,[98,99] a process that may not only involve significant trial and error, but may still not yield a model whose hyperparameters are approaching optimal values. Despite well-established optimisation routines defined for "black-box" optimisation, there remains a lack of an out-of-the-box solution for optimisation of MLIP hyperparameters to act upon existing MLIP fitting software, representing a hurdle to the widespread efficient parameterisation of MLIPs.

1.3 Research Aims and Thesis Structure

The central motivation of this thesis is to address the challenge of creating accurate, robust, and efficient machine learning interatomic potentials in a systematic and

transferable way. This work is built on two primary aims:

(i) To develop a performant, flexible cross-platform software package for the automated hyperparameter optimisation of a wide range of MLIP architectures.

(ii) To apply this software package to challenging problems in materials science, specifically the study of amorphous phase-changing materials and disordered solid-state ion conductors, demonstrating that systematic optimisation enables new scientific insights with order of magnitude cost reductions compared to previous modelling methods.

To achieve these aims, I introduce my software package, `XPOT`, a Cross-platform Optimiser for Machine Learning Potentials. The subsequent chapters are structured as follows:

Chapter 2: This chapter introduces the theoretical foundations of MLIPs, detailing the key components of data generation, atomic environment descriptors, and regression algorithms. The specific architectures explored in this work—GAP, (q)SNAP, and ACE—are discussed in detail, alongside Bayesian optimisation used in this work.

Chapter 3: Here, the design and implementation of the `XPOT` package are presented. Initial benchmarks are used to demonstrate the improvements in model accuracy and efficiency that can be gained through systematic hyperparameter optimisation compared to existing models in literature.

Chapter 4: The framework is extended to the efficient ACE class of potentials. I use the optimisation of models for elemental silicon and the phase-change material Sb_2Te_3 to benchmark the performance of the `XPOT` framework, with rigorous numerical and physical validation compared to state-of-the-art models in literature. Finally, I explore collaborative work wherein I fit models for the Ge–Sb–Te system, and elemental Te to drive the study of crystallisation in phase-changing memory devices.

Chapter 5: I develop an optimised MLIP for the solid-state electrolyte $Li_{14}SiP_6$ (LSP). I explore how the large-scale, long-timescale simulations enabled by this

model allow us to iterate on previous structural models for LSP, and study the diffusion mechanisms in this disordered electrolyte.

Chapter 6: I explore the application of "knowledge distillation" for MLIPs. Using XPOT, I study water, and compare the effects of hyperparameter optimisation to existing state-of-the-art foundation models and MPNN architectures.

Chapter 7: Finally, this chapter provides a summary of the key conclusions of the thesis and offers an outlook on future directions for the field, particularly in the context of the continued rise of foundation models and the ongoing integration of AI into materials discovery.

Chapter 2

Methods

This Chapter is a technical introduction to the methods that are used in the chapters to follow. I will summarise dataset generation (Section 2.1.1), MLIP architectures (Section 2.1.2 and Section 2.1.3), and Bayesian optimisation (Section 2.2). However, newly derived methods specific to work carried out in individual chapters are not included here, and these are introduced in their respective chapters to provide a more cohesive reading experience.

2.1 Machine Learning Interatomic Potentials

Machine learning is the task of training a computer system to learn from reference data, and in turn, make predictions based upon it, without being specifically programmed to do so. In the case of Machine learning interatomic potentials (MLIPs), a function is trained to approximate (as accurately as possible) the potential energy surface (PES) of a material based on a given reference labelling method (e.g., DFT). As such, there are two main areas of study for improving MLIPs: 1) the nature of said functions (and therefore models) themselves, and 2) the quality of the reference data they learn from. In this section, I will lay out the MLIP architectures used in my work, and the methods used to generate the reference datasets used to train them.

The task of developing an MLIP can be split into three steps: training data curation, descriptor generation, and regression.[100,101] In this section, I will provide an overview of these three steps, via the methods which are used in this thesis. Figure 2.1 visualises the MLIP training process, summarising the task of fitting an MLIP to a reference dataset. In this spirit, I describe the methods from left to right, chronologically for fitting of an MLIP.

2.1.1 Data for machine learning of the PES

The first step in building any machine learning model is determining the data from which the model will learn its functional form. In the wider data science landscape the field of "data engineering" refers to the design of the systems which collect and "clean" data, such that they are ready for use in training machine learning models. In the case of MLIPs, we need to design a collection of structural models labelled with target features which we want the model to learn. This is often referred to as the "reference database". In building a reference database for MLIP training there are two main considerations; firstly, the number and variety of structural models (cells), and secondly, the method used to provide the labels of energies and forces (that define the PES) for these structures.

There is an old adage in computer science which is especially relevant for machine learning: "Garbage In, Garbage Out" or GIGO. The phrase was first popularised

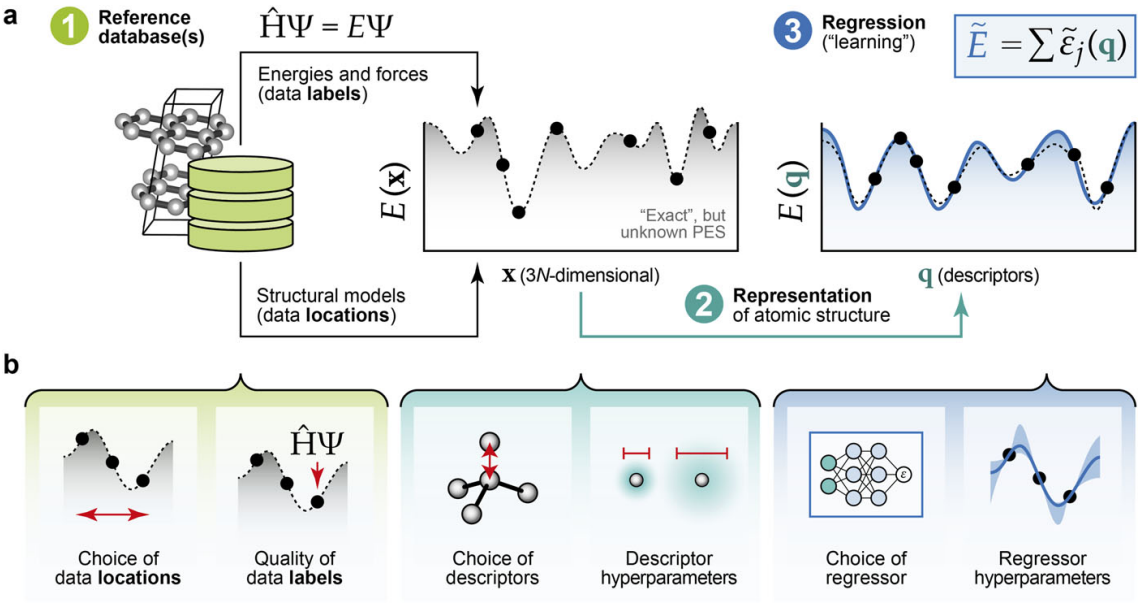

Figure 2.1: Machine Learning Interatomic Potentials. (**a**) An overview of the MLIP training process, with the three main steps highlighted. The dataset (step 1) contains structures labelled with ground-truth energies and forces. The representation (step 2) encodes each atomic environment into a vector **q**. The regression (step 3) fits the model in **q** space. (**b**) The key factors which determine the quality of the resulting MLIP for each stage. Figure reproduced with permission from Ref. 102, Copyright 2023 American Institute of Physics.

after being published in an article on the work of military mathematicians in The Times of Northwest Indiana in 1957.[103] As succinctly explained by Army Specialist W. D. McMellin:

> *If the problem has been sloppily programmed the answer will be just as incorrect. If the programmer made mistakes the machine will make mistakes. It cannot do one thing: It can't think for itself.*

While McMellin was referring to much more direct programming of computing devices of the 1950s, the sentiment also applies to machine learning. One can underline the parallels by considering human programming of these machines as analogous to the choosing of reference dataset for machine learning models. If one provides a machine learning model with data that are of insufficient diversity and quality, that is, that the data does not represent the real-life PES, then the outputs from the model will have a similarly loose grip on the PES, and thus, the processes one wishes to model.

For the purpose of this thesis, I define a reference data point as such: a structural model containing *at least* the relative positions in 3D space of the atoms in reference to each other, the energy of the structural model, and the force components calculated per-atom. An individual reference data point contain have more labels included than this (e.g., virials, atomic charges, or magnetic moments), but this definition encompasses the information used to train many MLIPs, although some may be trained on solely energy labels (e.g., EDDP models[104]), or other labels.

Without sufficient sampling of the potential energy surface (read: variety of chemical environments present in the structural models making up the reference dataset) the model will not have the opportunity to accurately represent large swathes of configurational space, in turn condemning it to instability or inaccuracy in these areas. In short, the model must be trained upon a sufficiently diverse set of structures to ensure the validity of its predictions across configurational space. But it is not only breadth of data that is required; to ensure the model's ability to

perform its prescribed purpose, the reference dataset should centre sampling of the PES around relevant chemical environments (e.g., relevant crystalline structures, or reaction intermediates).

Once these samples within configurational space are selected, the method for calculating the ground-truth PES labels must be chosen. A restriction on the size of the representative structural models is that they must be sufficiently small (in number of electrons, not volume) that a large number of *ab-initio* calculations can be done to provide labels. *Ab-initio* methods calculate the energies and other chemical properties based solely on the positions of the atoms, and therefore electrons, for a given structural model. The label quality is critical to an MLIP's ability to provide useful insights into the behaviour of materials. There is not one true method for directly characterising the PES of any material, indeed, there are a variety of *ab-initio* methods which can be used to characterise the PES. Generally, researchers use Density Functional Theory (DFT). The process begins with selecting a functional based on benchmarks for systems of a similar nature to theirs, but often different models for the same material may be based on different functionals.[42,55]

Ab-initio MD

One of the earliest data generation techniques for MLIPs was that of *ab-initio* MD. In this protocol, DFT is used to drive simulations of (small) structural models, and samples are taken from the resulting trajectories. As an understanding of the requirement for diverse chemical environment data was developed, DFT-driven MD was performed at higher temperatures, as these high-temperature simulations sample a wider area of configurational space. The out-of-the-box nature of AIMD (being easily implemented in a number of packages which are well documented both technically and from a research standpoint) drove high uptake of this method for many early MLIPs.

There are, however, now understood to be drawbacks to this approach. Firstly, AIMD is expensive: the cost of AIMD in many ways has motivated the development

of MLIPs. Secondly, the structures are necessarily (strongly) correlated, meaning that running 1000 steps of AIMD does not give 1000 diverse and useful structures for training of an MLIP. As AIMD is driven by the PES (as described by the DFT functional used), the configurations explored will be those that are energetically favourable and accessible from the starting structure under the conditions defined. As such, AIMD *in isolation* has fallen out of favour as a technique for generating training datasets for MLIPs.

Manual Selection of Structures

On the other side of the human effort scale lies manual selection of structures. This involves the researcher manually defining structures which they deem to be relevant to the system of interest. This is particularly useful when one wants to study specific microstructural features such as defects, grain-boundaries, or pores. Here, a researcher leverages their subject matter expertise to create a number of structures which are relevant for describing the chemistry of the system of interest, and manually ensure the inclusion of diverse relevant chemical environments. Many MLIPs have been fitted on datasets curated this way,[21,54,55] and it is often used as the starting point from which datasets can be extended using more automated techniques.[4,105,106]

In this way, datasets can be curated which are centred around domain-specific structural models directly related to the scientific task the model is designed for, while automated data generation techniques can be used to improve robustness and diversity of chemical environments, improving model stability beyond the target scope. Manual identification of structures of interest continues to be used today, and is as relevant as ever in light of the rise of the fine-tuning of foundation models (Chapter 6).

Iterative MD

A technique which combines the approaches of AIMD and manual structure selection is that of iterative MD. Iterative MD works by using an interim MLIP (that is, one

that has been fitted to a small, incomplete dataset, or one under development) and running MD simulations with it to generate new configurations which will be labelled with DFT and fed back into the dataset.

While such interim models are not well suited to production simulations to describe physical and chemical processes, this lack of completeness is advantageous for generating new structures for a dataset. The model (being trained on a limited dataset) is more likely to make noisy and inaccurate predictions in regions of the PES not represented in the training data, increasing the probability of exploring configurational space which would not otherwise be explored. By adding structures that occur during simulation, the model learns more the configurational space explored, and a new model is fitted on the new dataset. This iterative process occurs until the model is sufficiently converged.

One advantage of this approach is that MLIP-driven MD (or MLMD) is significantly cheaper than AIMD, and scales with the number of atoms, N_{at}, and not N_{at}^3, allowing for a significant increase in throughput. However, the process is often driven by manual analysis of trajectories (especially with very early models) where one must identify which structures are relevant or useful, and perform many iterations. Conversely, in the latter stages (when a potential is relatively well-converged for the conditions being described) new protocols often need to be designed in order to continue sampling novel atomic environments and avoid continuous layering of similar data points.

Active Learning

Active learning for MLIP training attempts to more efficiently sample phase space, selecting optimally diverse structural models to label for either generating from scratch or extending a dataset iteratively.[90,107–109] To do so, it is helpful to evaluate the uncertainty of any prediction made by a model for any structure. Once an uncertainty metric has been defined, it is possible to automate detection of when a model has become "too uncertain" of its predictions and the structure can be selected

for labelling and addition to the training dataset. Dependent on the architecture of the MLIP, there are several avenues which have been explored for generating these uncertainties.

Ensembles of models are often used for neural networks across machine learning fields, with the spread of predicted values determining the uncertainty of the prediction (i.e., a larger disparity in the predictions signals a greater level of uncertainty).[110,111] Such ensembles are usually generated by training several models on the same training database and same hyperparameters, but with different random initial states (i.e. that the parameters of a model are assigned random initial values). For ACE and Moment Tensor Potential (MTP) models, the D-optimality criterion is used to provide a representation of how far from the existing reference database a new structure is in feature space.[90,112] By doing so, it is possible to determine whether the model is "extrapolating" or "interpolating" and thus provide a level of uncertainty based on this distance from the convex hull of the known data. For Gaussian Process Regression (GPR) models, Bayesian variance can be used (Section 2.2).

Recent studies of materials often use active learning as a tool for the building upon hand-picked data in their iterative data generation process, extending the dataset via the guidance of the model's uncertainty.[4,98,113]

Synthetic data

Thus far, I have discussed methods for generating *structures* for creation of a training database, with the expectation that DFT will be used to generate the labels for the structures. For many years of my research this was true not only for the models I built, but the field at large. However, it is also possible to create "synthetic" data by using a cheaper method to label the data, approximating DFT.[59,114] Indeed, some of the earliest investigations of using NNs for modelling the PES were performed on a form of synthetic data in 1995, where Blank *et al.*[115] trained an ANN on the parameters of a Lennard-Jones potential from 42 dimer configurations.

In theory, one could use any model which can predict the PES — such as the cheap empirical potentials used by Blank *et al.*[115] — but in most cases it is preferable to use a model which most accurately approximates the PES within the confines of the computational resources available. By using an approximate model, the cost of labelling a single structure is significantly reduced compared to DFT. With access to these faster labelling methods, it is possible to create a reference dataset of millions of synthetically-labelled structures for fitting MLIPs, allowing robustness and accuracy due to reduced extrapolation based on the distance between datapoints.

There are however, risks in using synthetic data. The labels provided must be of a high quality, and valid for all structures that are included. The reference model must be validated thoroughly against DFT methods to ensure that in training a model it does not enter the "garbage in, garbage out" regime, and that the training dataset for the MLIP is sufficiently representative of the "true" PES.

The primary aim in generating synthetic data is that model architectures with less predictive accuracy and reduced inference cost can learn the behaviour of more expensive models under the sheer weight of data used in training.[59,116,117] Note, this assumes that there is a corresponding increase in data diversity, as if atomic environments are replicated, then the increase in data is not beneficial. Further discussion of using synthetic data for training is included in Chapter 6.

Beyond the use of fully synthetic datasets as a final target for training, synthetic data use has been proposed as a pre-training tool for NNs,[116] and as a tool for directly augmenting a reference dataset.[118] In pre-training, a large corpus of synthetic data is used to train a model to learn the approximate PES of a system of interest, before the model is "fine-tuned" on a smaller dataset of DFT-labelled structures, imparting the robustness of a large dataset, but requiring far fewer expensive DFT calculations.

The desired aims of synthetic data augmentation are much the same as using only synthetic data, but the scope is often reduced, working to provide only stability in regions not explored by the reference database. In the scheme of synthetic

data augmentation included in `pacemaker` (fitting software for linear and non-linear ACE models), existing structures from the reference dataset are taken, altered, and labelled with an empirical Ziegler-Biersack-Littmark (ZBL) potential, a universal short-range repulsive model that describes the pairwise interaction between atomic nuclei at small distances. These structures are then added to the reference dataset, and weighted at a reduced level, aiming to provide "guardrails" for the model, but not affect the model's fit to the physically relevant structures in the reference dataset. This process is aimed at ensuring repulsive behaviour at short interatomic distances not included in the reference dataset. In all, synthetic data provides an interesting avenue for improving the robustness of MLIPs, as long as the method of generation is carefully considered and validated.

2.1.2 Describing local atomic environments

In machine learning, often raw data is transformed to improve the efficiency of learning and improve model performance. Referred to as featurisation, for MLIPs the chemical environments of the structure are transformed into a vector which is then passed to the model to use as input for making predictions. All MLIPs considered in this thesis make predictions per-atom, and thus any descriptor needs to describe the environment in which an atom resides (that is, its atomic environment). An ideal descriptor should efficiently provide information about the local structure, be translationally, rotationally, and permutationally invariant (like the true PES), characterise every unique atomic neighbourhood into a unique point in latent space (i.e., complete), and be smooth (i.e., not include any discontinuities). Additionally, finding computationally efficient descriptors is required, as they are computed both during training, and at each inference step.

Beyond fitting potentials, descriptors can also be used to query the nature of local atomic environments in a dataset. By calculating descriptors we can understand how our reference dataset is represented to a regressor when fitting an MLIP, or carry out analysis using descriptor similarity on structures. The latter is used widely for

the studying of crystallisation in amorphous materials.[4,5,59]

In this thesis, three descriptors are used: bispectrum components for (q)SNAP, smooth overlap of atomic positions (SOAP) for GAP, and atomic cluster expansion (ACE) for ACE potentials.

Bispectrum components

Bispectrum component hyperparameters are optimised in Chapter 3, and I briefly outline the formulation of these descriptors here.

The Spectral Neighbour Analysis Potential (SNAP) methodology employs bispectrum components as descriptors of an atom's local environment.[119] These descriptors are generated by defining local atomic density around a central atom, and projecting it onto a 4D hyperspherical harmonic basis.

For a central atom i, a neighbour density function, $\rho_i(r)$, is defined. This function is constructed as a combination of Dirac delta functions; each function signifies the precise location of either the central atom i, or one of its neighbouring atoms j. To modulate the influence of these neighbours, a weight, w_j, is assigned to each (capable of differentiating between atomic species), and their contribution is smoothly attenuated by a switching function, $f_c(r_{ij})$. The switching function ensures that the weight of the neighbour's contribution to the descriptor approaches 0 upon approaching the cutoff distance, R_{cut}, giving the following equation:

$$\rho_i(r) = \delta(r) + \sum_{r_{ij}<R_{\text{cut}}} f_c(r_{ij}) w_j \delta(\mathbf{r} - \mathbf{r}_{ij}) \tag{2.1}$$

where $\mathbf{r}_{ij}$ is the vector from atom i to atom j. To implement invariance with respect to translation, rotation, inversion, and permutation (referred to as TRIP invariance), Thompson *et al.*[119] transform the neighbour density from 3D to the 4D basis. The radial coordinate, r, is mapped to a third spherical angle θ_0:

$$\theta_0 = \theta_0^{max} \frac{r}{R_{\text{cut}}} \tag{2.2}$$

This mapping represents all of 3D space onto the surface of a hypersphere. Functions residing on this hypersphere are suitably expanded using the basis of 4D hyperspherical harmonics, denoted as $U^j_{m,m'}(\theta_0, \theta, \phi)$. These harmonics constitute a complete set for describing functions on the hypersphere.

Consequently, the neighbour density function (Equation 2.1), following the radial remapping, can be expressed as a sum over these basis functions:

$$\rho(r) = \sum_{j=0,1/2,\ldots}^{\infty} \sum_{m=-j}^{j} \sum_{m'=-j}^{j} u^j_{m,m'} U^j_{m,m'}(\theta_0, \theta, \phi) \tag{2.3}$$

The coefficients of this expansion, $u^j_{m,m'}$, quantify the contribution of each specific hyperspherical harmonic to the total neighbour density. As the neighbour density is a weighted sum of the δ-functions, an expansion coefficient can be defined as:

$$u^j_{m,m'} = U^j_{m,m'}(0,0,0) + \sum_{r_{ij}<R_{\text{cut}}} f_c(r_{ij}) w_j U^j_{m,m'}(\theta_0, \theta, \phi) \tag{2.4}$$

$u^j_{m,m'}$ coefficients are complex numbers and not rotationally invariant. To achieve rotational invariance Thompson *et al.* constructed the bispectrum components, $B_{j_1,j_2,j}$ by taking the following scalar triple products from the set of expansion coefficients $u^j_{m,m'}$:

$$B_{j_1,j_2,j} = \sum_{\substack{m_1,m'_1=-j_1 \\ m_2,m'_2=-j_2 \\ m,m'=-j}}^{\substack{j_1 \\ j_2 \\ j}} (u^j_{m,m'})^* H^{jmm'}_{j_1 m_1 m'_1, j_2 m_2 m'_2} u^{j_1}_{m_1,m'_1} u^{j_2}_{m_2,m'_2} \tag{2.5}$$

where $(u^j_{m,m'})^*$ is the complex conjugate of the expansion coefficient $u^j_{m,m'}$, $H^{jmm'}_{j_1 m_1 m'_1, j_2 m_2 m'_2}$ are coupling coefficients, analogous to the Clebsch-Gordan coefficients for rotations on a 3D sphere. These coefficients ensure that the triple product combination is a scalar under rotational transformations, in turn ensuring the bispectrum components $B_{j_1,j_2,j}$ are both real-valued and rotationally invariant. Physically, these

components provide a measure of the correlation strength between density features at three distinct "locations" on the hypersphere, as indexed by j, j_1, j_2.

Smooth overlap of atomic positions

SOAP descriptors were originally introduced as a tool for fitting more accurate MLIPs[60], but have seen widespread use beyond this as a tool for quantifying the nature of local atomic environments within large-scale simulations of complex structures.[40,59,120,121] A full definition of SOAP can be found in Ref. 60, but I include here an overview of the SOAP descriptor formulation, and henceforth I assume the single element case for simplicity.

SOAP descriptors are defined by first encoding the local atomic environment of an atom i through constructing a neighbour density $\rho^i(\mathbf{r})$ as a sum of Gaussians of width σ centred on atom i and all the neighbouring atoms j within a defined cutoff r_{cut},

$$\rho^i(\mathbf{r}) = \sum_j \exp\left[\frac{-|\mathbf{r} - \mathbf{r}_{ij}|^2}{2\sigma^2}\right] f_{\text{cut}}(r_{ij}) \tag{2.6}$$

where σ controls the smoothness of the atomic density, and $f_{\text{cut}}(r_{ij})$ is a smoothing function such that

$$f(r_{ij}) \to 0 \text{ smoothly as } r_{ij} \to r_{\text{cut}} \tag{2.7}$$

Ensuring rotational invariance is achieved by expanding in a basis of orthogonal radial basis functions $R_n(r)$, and spherical harmonics $Y_l^m(\hat{\mathbf{r}})$,

$$\rho^i(\mathbf{r}) = \sum_{nlm} c^i_{nlm} R_n(r) Y_l^m(\hat{\mathbf{r}}) \tag{2.8}$$

where the coefficients c^i_{nlm} are defined as:

$$c^i_{nlm} = \int d\mathbf{r} R_n(r)^* Y_l^m(\hat{\mathbf{r}})^* \rho^i(\mathbf{r}). \tag{2.9}$$

In practical use n and l are convergence parameters (within reasonable values)

which determine the complexity of the descriptor. Extension to multi-element systems can be achieved through defining linear combinations of densities for efficiency reasons[122], or a complete extension can be achieved by naïve expansion returning one channel per element combination.

Finally, by summing over all neighbouring atoms within the cutoff of the central atom i the SOAP vector is defined as:

$$\rho^i_{nn'l} = \frac{1}{\sqrt{2l+1}} \sum_m (c^i_{nlm})^* c^i_{n'lm}. \tag{2.10}$$

The resulting SOAP vector is TRIP invariant, and smooth and continuous as atomic displacements change. For most applications, the SOAP vector is then normalised (i.e., divided by its own modulus). To use SOAP descriptors for Gaussian Approximation Potentials (GAP), we must define a similarity kernel which can determine the similarity between the local environments of two given atoms. Most applications use a low-order polynomial kernel

$$k(A, A') = (\xi \cdot \xi')^\zeta \tag{2.11}$$

where A and A' are the atomic environments being compared, ξ and ξ' are the corresponding SOAP vectors (Eq. 2.10) which have been normalised, and ζ is the power to which the kernel is raised. Using a polynomial kernel significantly reduces computational cost when compared to the Gaussian equivalent. The value of ζ in turn determines the amount to which the differences between the two SOAP vectors are emphasised.[60]

Atomic cluster expansion

ACE builds upon the spin cluster expansions,[123] extending them to handle continuous degrees of freedom inherent in atomic positions. It has been shown that the ACE descriptor unifies a number of other descriptors, including the bispectrum component and SOAP descriptors outlined in the previous sections.[61,124]

To begin, it is possible to write the energy of a group of atoms $i = 1, ..., N$ as a many body expansion:

$$E = V_0 + \sum_i V_1(\mathbf{r}_i) + \frac{1}{2!}\sum_{i,j} V_2(\mathbf{r}_i, \mathbf{r}_j) + \frac{1}{3!}\sum_{i,j,k} V_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) + ... \tag{2.12}$$

where again, for simplicity of notation, I have assumed a single element system. As V_0 is an offset, it can be set to zero for the purposes of this section. To decompose this total energy into an individual atomic contribution, one can write the energy of an atom i as:

$$E_i = V_1(\mathbf{r}_i) + \frac{1}{2!}\sum_{j} V_2(\mathbf{r}_i, \mathbf{r}_j) + \frac{1}{3!}\sum_{j,k} V_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) + ... \tag{2.13}$$

where E is a sum over i of the per-atom contributions E_i. However, in this form the cost of computing the energy is the combinatorial scaling of the summation as the body order increases.

To combat this, a "density trick" is introduced, as used in SOAP and bispectrum descriptors[60], ensuring permutational invariance in constructing the atomic density around each atom i of species z as:

$$\rho_i^z(\mathbf{r}) = \sum_j^{j\neq i} \delta_{zz_j}\delta(\mathbf{r} - \mathbf{r}_{ji}) \tag{2.14}$$

where z_j refers to the species of the neighbour atom j.

The one-particle basis functions, $\phi_v(\mathbf{r})$, are irreducible representations of the rotation group defined as:

$$\phi_{nlm}^{z,z_j}(\mathbf{r}) = R_{nl}^{z,z_j}(r_{ji}) \cdot Y_l^m(\theta_{ji}, \phi_{ji}) \tag{2.15}$$

where $R_{nl}^{z,z_j}(r)$ is the radial basis, and $Y_l^m(\theta, \phi)$ is the real spherical harmonic. With these basis functions in hand, the atomic base (A_{iv}) is defined as:

$$A_{z_i v} = \sum_j \phi_{nlm}^{z,z_j}(\mathbf{r}_{ji}) \tag{2.16}$$

where $v = znlm$ is a unique index for each basis function characterising the chemical species, type of radial function, and spherical harmonics.

The permutationally invariant "A-basis" is defined as a product of the atomic bases for each atom in the cluster:

$$\mathbf{A}_{z_i v} = \prod_{t=1}^{v} A_{z_j v_t}, \tag{2.17}$$

where the value of v is the maximum body-order, (i.e., $t = 2$ for a three-body basis function as t is the number of neighbours not including the central atom). It should be noted that these products are not invariant with respect to rotation, but this is achieved through the use of Clebsch-Gordan coefficients to convert between the A- and B-basis:

$$\mathbf{B}_{i\mathbf{v}} = \sum_{\mathbf{v'}} \mathcal{C}_{\mathbf{vv'}} \mathbf{A}_{i\mathbf{v'}}, \tag{2.18}$$

2.1.3 Regression for MLIPs

Finally, having generated a labelled dataset, and transformed the input features $\mathbf{X}$ (that is, the Cartesian coordinates) into a descriptor $\mathbf{x}$, we can now fit a regressor to these descriptors to model the PES, $f(\mathbf{x})$. I do not discuss here the requirements for validation of models, but this is discussed for each model individually, and in the context of loss functions in Chapter 3, a good overview is given by Ref. 102. Generally, we can understand that the accuracy of a model is dependent on its ability to learn from the training data the underlying PES, not just in the interpolative regime, but in the extrapolative regime as well.

In the case of MLIPs, and specifically regression over local descriptor space, the feature space is high-dimensional, and thus regression methods must be capable of

handling this inbuilt complexity. Among the many regression methods available, the most common for MLIPs can be described into three main categories:

- **Linear models**: These models are often the most efficient, such as SNAP,[119] linear ACE,[118,125] and the Moment Tensor Potential (MTP).[126]
- **Kernel methods**: Most notably Gaussian Approximation Potentials (GAP),[52,101] these methods are often more expensive than linear models, but can be smoother and more accurate.
- **Artificial neural networks (NNs)**: Examples of neural network architectures include Behler-Parinello NNs,[51] SchNet,[127] and DeepMD.[128] Recent breakthroughs have taken advantage of graph neural networks, implementing message-passing between atoms to model the PES. Popular implementations include MACE,[129] NequIP,[62] and GRACE.[66]

Each model architecture has its own strengths and weaknesses, and the choice of descriptor and regression method is often a balance between performance, efficiency, and practicality for a given application. In this thesis, I fit SNAP, qSNAP, and ACE models to capture the PES of disordered materials. Additionally, GAP models are fitted in the iterative curation of datasets, and so I include a brief overview of these model architectures here.

SNAP

Based upon the bispectrum components defined in Eq. 2.5, the SNAP energy of an atom can be as a function of K bispectrum components B_k^i where

$$\mathbf{B}^i = \{B_1^i, ..., B_K^i\}, \tag{2.19}$$

then the predicted atomic energy E_i can be written as:

$$E^i(\mathbf{B}^i) = \beta_0^{z_i} + \sum_{k=1}^{K} \beta_k^{z_i} B_k^i \tag{2.20}$$

where $\beta_0^{z_i}$ and $\beta_k^{z_i}$ are the intercept (isolated atom energy) and coefficients for the atomic energy of atoms of species z_i. As the total energy (for a single element system) is a sum over all atomic energies, the PES is defined as:

$$E = \sum_i^N E^i(\mathbf{B}^i) = N\beta_0 + \boldsymbol{\beta} \cdot \sum_i^N \mathbf{B}^i \tag{2.21}$$

Forces are then calculated as the negative gradient of the PES with respect to the atomic positions:

$$\mathbf{F}^j = -\nabla_j \sum_{i=1}^N E^i(\mathbf{B}^i) = -\boldsymbol{\beta} \cdot \sum_{i=1}^N \frac{\partial \mathbf{B}^i}{\partial \mathbf{r}_j} \tag{2.22}$$

where $\mathbf{F}^j$ is a weighted sum of the derivatives of the bispectrum components of all atoms i with respect to the position of atom j.

Quadratic SNAP

Quadratic SNAP extends the maximum body-order of the SNAP descriptor by including an embedding energy term to the linear form

$$E^i = \boldsymbol{\beta} \cdot \mathbf{B}^i + \frac{1}{2} \left(\mathbf{B}^i\right)^\intercal \cdot \boldsymbol{\alpha} \cdot \mathbf{B}^i, \tag{2.23}$$

where $\boldsymbol{\alpha}$ is a symmetric matrix.

Substantial accuracy improvements were found in transitioning from the linear form of SNAP to the quadratic form.[81,130] However, substantially more reference labels were needed to achieve convergence between training and validation errors,[130] and the computational cost of training the quadratic form (for a fixed number of bispectrum components) is significantly higher.

Zuo *et al.*[81] trained models for a benchmark study on the performance of MLIPs, and the results are shown in Fig. 2.2. Panel **a** shows the benefits of the qSNAP formulation, where the errors of the model are up to two times lower than the corresponding linear SNAP models. However, panel **b** shows that the flexibility of the model comes at a cost, whereby the flexibility of the model is limited by

the number of reference labels available. Increasing flexibility of the model quickly results in overfitting, as the model improves its fit to the training data, but is incapable of generalising at the same accuracy to the testing data.

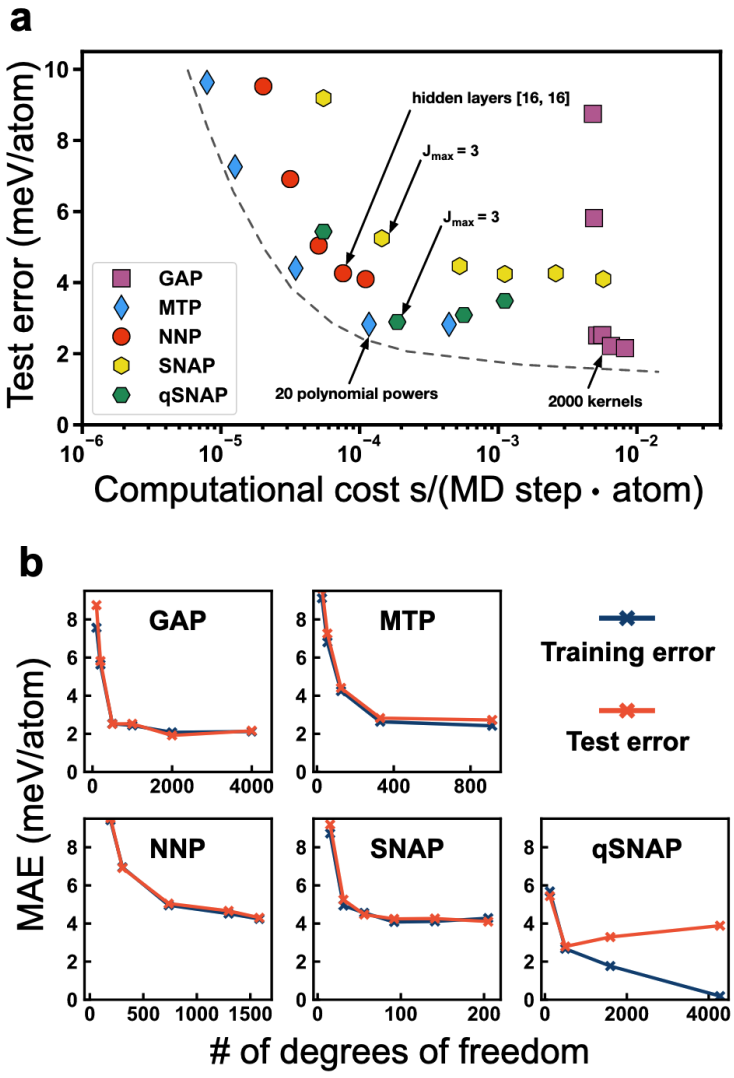

Figure 2.2: (**a**) Comparison of the testing RMSE vs. computational cost for multiple MLIP architectures. (**b**) Plots of training and testing errors against flexibility of the model. Reproduced with permission from Ref. 81, Copyright 2020, American Physical Society.

This behaviour is not unique to qSNAP, but is a general trend for increasing the complexity (read: number of parameters) of a model, whereby the flexibility of the model allows better fitting to the data, but thus requires more data to ensure that the model's predictions are valid across the phase space of interest.

Gaussian Approximation Potentials

Gaussian approximation potentials use Gaussian process regression (GPR), normally over SOAP vectors, to model the PES. GPR is a non-linear, non-parametric regression method which is used to predict the PES from the high-dimensional space of the SOAP descriptors. The locality of a single prediction is determined by the width σ of the Gaussian kernel, with individual points being the atomic environments described by the SOAP descriptors. As GPR is a Bayesian method, it not only provides a prediction of the atomic energy, but also an uncertainty of this prediction based on the distance of the atomic environment from the set of known atomic environments, (see Ref. 101).

The formulation of the energy of atom i can be written as:

$$E(\mathbf{x}_i) = \sum_{s=1}^{M} c_s K(\mathbf{x}_i, \mathbf{x}_s), \tag{2.24}$$

where $\mathbf{x}_i$ is the SOAP descriptor of the atomic environment of atom i, and s runs over a representative set of M atomic environments, providing $x_1, ..., x_M$ as the set of descriptors whose environments are the basis in which the model is expanded. Finally, c_s are the coefficients of the model learned during fitting.

Importantly, the representative set, $x_1, ..., x_M$, could in theory be the entire training set, but as training set often comprised of more than 10^5 atomic environments, this is computationally infeasible. As such, the representative set is typically a subset of the training set, and is chosen to be the most representative of the phase space of interest, and the models as such are in reality "sparse" (Fig. 2.3).[101,131]

To fit the GPR model the coefficients are learned by minimising the loss function, $\mathcal{L}$, over the training data, which is defined as:

$$\mathcal{L} = \sum_{i=1}^{N} \frac{[y_i - \tilde{y}(\mathbf{x}_i)]^2}{\sigma_i^2} + R, \tag{2.25}$$

where y_i is the true energy (read: database reference) of atom i in the training data.

The parameter σ_i is an effective weighting parameter set during fitting, allowing the model to weight predictive errors for some configurations (e.g., crystalline, defects, transition states) more heavily than others, which is often set manually, although an energy-based convex hull of the training data can be used to set the weights automatically. R is a regularisation term.

The regularisation term in the loss function, R, is a form of the Tikhonov regularisation, and is defined as:

$$R = \sum_{m,m'}^{M} c_m k(\mathbf{x}_m, \mathbf{x}_{m'}) c_{m'} \tag{2.26}$$

where $k(\mathbf{x}_m, \mathbf{x}_{m'})$ is the kernel function, and c_m and $c_{m'}$ are the coefficients of the model for the m and m' basis functions. R works to prevent overfitting of the model by penalising large coefficients, discouraging the model from relying heavily on a single feature.

One further important aspect of the sparsification of the GPR model is the ability to define the loss function across the entire training set despite the representative set being a subset. This in turn ensures that the model is optimised towards the

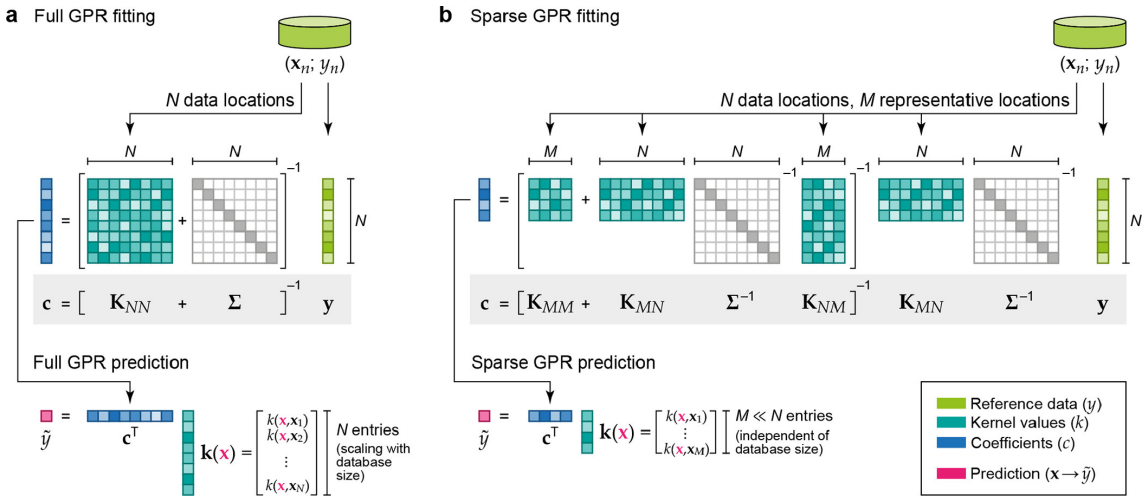

Figure 2.3: The sparsity of the GPR model is determined by the number of representative locations, M, specified by the user. Sparsification of the model decouples dataset size from the size of the model, and allows for the use of large datasets without increasing model cost. Reproduced with permission from Ref. 101. Copyright 2021, American Physical Society.

entire training set without increasing model cost beyond realistic values.

As described in Equation 2.11, the kernel function K is a low-order polynomial, and the value of ζ can be set to tune the sensitivity of the kernel to difference in atomic environment (vis. Table 3.1 in Chapter 3), with higher values of ζ resulting in a more sensitive kernel.

In the case of a single value of σ, whereby $\sigma_i = \sigma$ for all i, σ can be thought of as a global "regularisation strength". Considering the loss function above in matrix form, we can write:

$$\mathcal{L} = (\mathbf{y} - \mathbf{K}_{MN}\mathbf{c})^{\mathsf{T}} \boldsymbol{\Sigma}^{-1} (\mathbf{y} - \mathbf{K}_{MN}\mathbf{c}) + \mathbf{c}^{\mathsf{T}}\mathbf{K}_{MM}\mathbf{c} \tag{2.27}$$

where $\mathbf{y}$ is the vector of true energies, $\mathbf{c}$ is the vector of coefficients, $\boldsymbol{\Sigma}$ is a diagonal matrix of σ_i for $i = 1, ..., N$, and $[\mathbf{K}_{MN}]_{mn} \equiv k(\mathbf{x}_m, \mathbf{x}_n)$.

Minimising the loss function is then equivalent to solving:

$$\nabla_{\mathbf{c}^T} \mathcal{L} = 0 \tag{2.28}$$

and thus the optimal coefficients are given by:

$$\mathbf{c}_{\text{opt}} = \left(\mathbf{K}_{MM} + \mathbf{K}_{MN}\boldsymbol{\Sigma}^{-1}\mathbf{K}_{NM}\right)^{-1} \mathbf{K}_{MN}\boldsymbol{\Sigma}^{-1}\mathbf{y} \tag{2.29}$$

These fitted coefficients can then be used to predict the energy of a new atomic environment, $\mathbf{x}_i$, as:

$$\tilde{y}(\mathbf{x}_i) = \mathbf{c}_{\text{opt}}^{\mathsf{T}}\mathbf{k}(\mathbf{x}_i). \tag{2.30}$$

Thus, the energy of the new atomic environment is given by the coefficients of the model, $\mathbf{c}_{\text{opt}}$, and the kernel vector of the new atomic environment, $\mathbf{k}(\mathbf{x}_i)$, which is effectively comparing the similarity of the new atomic environment to each atomic environment contained within the representative set.

ACE models

ACE models are based upon the atomic cluster expansion (ACE) formalism outlined in Section 2.1.2. All formulations are based upon the idea of an atomic property, φ^i, which is a function of the atomic environment of atom i. An atomic property is arbitrary, but in the case of a linear model is the atomic energy, ε_i (Eq. 2.33). The relationship between the A-basis defined in Eq. 2.17 and the atomic property is given by:

$$\varphi_i = \sum_{\nu=1}^{\nu_{max}} \sum_{\mathbf{v}} \tilde{\mathbf{c}}_{\mathbf{v}} \mathbf{A}_{i\nu} \tag{2.31}$$

where $\tilde{\mathbf{c}}_\nu$ are the model coefficients, and ν_{max} is the maximum body-order of the basis functions. This equation is equivalent to the following B-basis expansion:

$$\varphi_i = \sum_{\mathbf{v}} \mathbf{c}_{\mathbf{v}} \mathbf{B}_{i\mathbf{v}} \tag{2.32}$$

in which the B-basis functions are as described in Eq. 2.17.

In the most simple case, that is the linear case, the energy of an atom i is given by:

$$\varepsilon_i = \varphi_i = \sum_{\mathbf{v}} \mathbf{c}_{\mathbf{v}} \mathbf{B}_{i\mathbf{v}}, \tag{2.33}$$

where ε_i is the per-atom energy of atom i. Again, the energy of the system is a sum over all per-atom energies.

However, not all ACE models are limited to the linear case, and a common extension is to introduce non-linearity via a Finnis–Sinclair-inspired embedding[132] such that:

$$\varepsilon_i = \varphi_i^{(1)} - \sqrt{\varphi_i^{(2)}} \tag{2.34}$$

where $\varphi_i^{(1)}$ and $\varphi_i^{(2)}$ are two separate atomic properties, which each act upon the

same *basis* but have separate coefficients for said basis functions. More generally, it is possible to extend the ACE model to include any number of atomic properties such that

$$\varepsilon_i = \mathcal{F}(\varphi_i^{(1)}, ..., \varphi_i^{(P)}) \tag{2.35}$$

where P is the total number of atomic properties, and $\mathcal{F}$ is a function of the atomic properties.[61] This could in theory be either an explicit non-linearity (such as that in the Finnis–Sinclair example above), or a more general approximation such as in a neural network, but in ACE models as fitted by `pacemaker` is explicit.[133] The nature of this expansion means that there are two respects in which an ACE model can be converged, that is, with respect to the form of $\mathcal{F}$, and with respect to the basis functions $\mathbf{B}_{i\mathbf{v}}$.[118]

Before moving on to how the coefficients are determined, the loss function, and the fitting procedure, it is important to note that there are several fitting frameworks based on ACE descriptors. In this thesis, I fit ACE potentials using `pacemaker`, but there are many other implementations available.[66,129,134,135] As such, I will explain here the loss function and fitting procedure as laid out in `pacemaker`.

In the case of `pacemaker`,[133] non-linearity is implemented explicitly (i.e., the user defines the exact non-linear form of the model — by default this is a Finnis–Sinclair-inspired embedding vis. Eq. 2.33), without a notable increase in computational cost involved in increasing the non-linearity of a model.[2,98,133] Additionally, by increasing the degree of non-linearity in the model (that is, increasing P such that all exponents are distinct), the flexibility of the model is increased. The increase in the number of atomic properties used to model the atomic energy does affect the training speed, but does not significantly change the inference cost, this is determined predominantly by the number of basis functions and cutoff defined for the model.

To fit the ACE model, regardless of linearity, one must find the values of the coefficients $\mathbf{c}_\mathbf{v}$ that minimise the loss function. Notably, it is specifically $\mathbf{c}_\mathbf{v}$ in the

B-basis that are learned, due to the fact that the B-basis is rotationally invariant, and have appropriate symmetries. I note that for efficient inference, the B-basis is converted to the A-basis for use in simulation, and further details can be found in Ref. 133.

The loss function in `pacemaker` is defined as:

$$\mathcal{L} = (1-\kappa)\Delta_E^2 + \kappa\Delta_F^2 + \Delta_{\text{coeff}} + \Delta_{\text{rad}} \tag{2.36}$$

where Δ_E is the energy loss, Δ_F is the force loss, with their contributions scaled by κ, Δ_{coeff} is a regularisation term to penalise large values of the coefficients, and Δ_{rad} is a regularisation term for radial smoothness. If we expand the energy and force loss terms into their true forms in `pacemaker`, we get:

$$\mathcal{L} = (1-\kappa)\sum_{n=1}^{N} w_n^E \left(\tilde{E}_n - E_n\right)^2 + \kappa \sum_{n=1}^{N}\sum_{i=1}^{N_{\text{at}}} w_{ni}^F \left(\tilde{F}_{ni} - F_{ni}\right)^2 + \Delta_{\text{coeff}} + \Delta_{\text{rad}} \tag{2.37}$$

where N is the number of training structures, N_{at} is the number of atoms in each given training structure, $\tilde{E}_n$ is the predicted energy of structure n, E_n is the true energy of structure n, $\tilde{F}_{ni}$ is the predicted force on atom i in structure n, and F_{ni} is the true force of atom i in structure n. The weights w_n^E and w_{ni}^F are the weightings (per-structure and per-atom respectively) of the individual energy and force errors. These can be manually set by the user to target higher accuracies on specific structures of interest.

To understand the coefficient regularisation term, Δ_{coeff}, we must expand the multi-basis $\mathbf{v}$ into its constituent parts, and the resultant form of the term is:

$$\Delta_{\text{coeff}} = L_1 \sum_{p\mu\mathbf{nIL}} \left|c^{(p)}_{\mu\mathbf{nIL}}\right| + L_2 \sum_{p\mu\mathbf{nIL}} \left|c^{(p)}_{\mu\mathbf{nIL}}\right|^2, \tag{2.38}$$

where L_1 and L_2 are the regularisation parameters within the elastic net regularisation framework.[136] Although the L_1 and L_2 values do not seem to have a

significant effect on the accuracy for the potential within the interpolative regime, a previous study found convergence between interpolative and extrapolative accuracy when these parameters are increased.[118]

The second regularisation term, Δ_{rad}, works to stabilise the potential in regions lacking training data by restricting the shape of the radial functions. It can be written as:

$$\Delta_{\text{rad}} = \frac{w_0}{r_{\text{cut}}^2} \int r^2 \sum_{nl} |R_{nl}| \, dr + \frac{w_1}{r_{\text{cut}}^2} \int r^2 \sum_{nl} \left| \frac{dR_{nl}}{dr} \right| dr + \frac{w_2}{r_{\text{cut}}^2} \int r^2 \sum_{nl} \left| \frac{d^2 R_{nl}}{dr^2} \right| dr, \tag{2.39}$$

where r_{cut} is the cutoff radius, R_{nl} is the radial function, and w_0, w_1, and w_2 are the radial regularisation parameters. From this formulation, it becomes clear how each regularisation parameter constrains the shape of the radial function dependent on their relative values. Functionally, small values (resulting in $< 5\%$ of the total loss) are used to stabilise oscillations without affecting the model's ability to learn the overall shape of the radial function from the training data.[118]

Energy and force weighting terms are set uniformly across all structures by default, but can either be set manually or via an energy-weighting scheme which determines their weights based on the energy difference between the lowest energy structure and the structure of interest. I do not discuss this further here, as it is not utilised in this work, however, Ref. 118 provides a detailed discussion on the energy weighting schemes available in `pacemaker`.

2.2 Bayesian optimisation

Bayesian optimisation (BO) is an optimisation method which is capable of optimising black box objective functions, and is robust to the presence of noise in the objective function. A "black box" function refers to a function where only the inputs and outputs are known, and the functional form, internal workings, or any other information about the function is unknown. In our case, the inputs are the hyper-

parameters of the MLIPs, the outputs are the target properties (e.g., energies and forces), and the objective function is the difference (loss) between the ground-truth PES and the MLIP-predicted PES as measured at a variety of points in configuration space (through a set of validation structures). A surrogate model for the objective function (here the accuracy to the PES with respect to hyperparameter space) is constructed, including quantification of the uncertainty of this surrogate, providing not only information on the expected value of the loss for a given point in optimisation space, but also the uncertainty of that prediction. This is particularly useful for situations in which the objective function is expensive to evaluate, where each new point requires significant computational expense.

The Bayesian optimisation process aims to find the global minimum of the objective function, which can be written as:

$$\min_{x \in \mathcal{X}} f(x), \tag{2.40}$$

where $f(x)$ is the objective (often defined as the loss) function, and $\mathcal{X}$ is the search space. This can also be defined to maximise a certain quantity in other cases. The typical properties of problems for which Bayesian optimisation is efficient and successful are:

- The dimensionality of the search space is not too large, i.e., $d \leq 20$ where $x \in \mathbb{R}^d$. This is due to the $\mathcal{O}(d^3)$ scaling of the Gaussian Process, as well as the "curse of dimensionality", whereby more and more datapoints are required to sufficiently sample the hyperparameter space (although recent work has shown that it is possible through adjustments of the assumptions in BO to improve high dimensional optimisation, see Ref. 137).

- The search space $\mathcal{X}$ is a simple set. To expand, that it is possible and easy to assess membership of the set, and that it can be defined through a set of simple constraints.

- $f(x)$ is observed without derivatives, and may be noisy in observation. Additionally, $f(x)$ lacks any known specific functional form which could be used to guide the optimisation process instead of a black box approach. In these cases, Bayesian optimisation may still be used, but the optimisation process will be inefficient compared to techniques which leverage the special knowledge of the form of $f(x)$.

Finally, the desired end-goal is to find the global minimum of $f(x)$ within the search space $\mathcal{X}$. For finding local minima, gradient descent might be used. Of course, it is not guaranteed that the global minimum will be found, but through the use of Bayesian optimisation, the probability of finding the global minimum is increased.

I now define the workflow for Bayesian optimisation. I will assume the case in which a Gaussian process (GP) is used as the probabilistic surrogate model, and the optimisation is performed sequentially. Additionally, I assume starting from the case where we have no data for the objective function upon initialising the optimiser.

First, a Gaussian process is placed on the objective function, $f(x)$. However, with no data, the first n_{init} predictions should be initialised via an initial exploration strategy. This could be a grid search, random search, or pseudo-random sequence, for example. Once this initialisation process is completed, the Gaussian process is updated, and there is now a well-defined looped process which occurs for each value of $f(x_i)$ which is computed. An acquisition function is maximised with a result x_n such that the next point to evaluate is chosen by a measure of where it is expected that the value of $f(x)$ will be the lowest. The output of the objective function $f(x_n)$ is observed, and the GP is updated, and thus the location of the maximum of the acquisition function is updated, providing a new point to evaluate. This process is repeated until a stopping criterion is met.

The GP regression on $f(x)$ produces a posterior probability distribution for $f(x)$ at each evaluated point x in the search space. At any point x, the posterior is

Gaussian distributed with mean and variance $\mu_n(x)$ and $\sigma_n^2(x)$ respectively, where n represents the points evaluated and thus the number of data points in the GP. The mean vector is given by evaluating a mean function μ_0 at each point x_i, and the covariance matrix is given by evaluating a covariance kernel Σ_0 at each pair of points x_i and x_j.

Thus, the prior distribution which assumes a multivariate normal distribution for $(f(x_1), ..., f(x_k))$ is given by:

$$f(x_{1:k}) \sim \mathcal{N}(\mu_0(x_{1:k}), \Sigma_0((x_{1:k}), (x_{1:k}))), \tag{2.41}$$

where $\mathcal{N}$ is the multivariate normal distribution, and $x_{1:k}$ represents the application of the functions to the set of input points $x_1, ..., x_k$. From this, we can define the posterior distribution. Without noise, and in the case now where I label x_{k+1} as x, the posterior distribution is given by:

$$f(x)|f(x_{1:k}) \sim \mathcal{N}(\mu_k(x), \sigma_k^2(x)). \tag{2.42}$$

The posterior mean, $\mu_k(x)$, is taken as a weighted average of the prior mean, $\mu_0(x)$, and an estimate from the observed data, $f(x_{1:k})$. The posterior variance, $\sigma_k^2(x)$, is taken as the variance of the prior distribution, $\sigma_0^2(x)$, minus the variance which has been removed by observing the values, $f(x_{1:k})$.

Armed with this definition, I consider the choice of covariance kernel. In Section 2.1.2 I discussed the polynomial kernel used there, however, this kernel is not used in the BO implementation used in my work. Generally, kernels possess the property that the distance between a new point and the existing data points is used to determine the covariance between the new point and the existing data points, and that the shorter said distance, the higher the correlation of values. This basic understanding helps to understand the nature of the BO process I now outline. I will focus here on the covariance kernel formulation used in this thesis, the Matérn kernel. The Matérn kernel is a flexible kernel which can be used to model a wide

range of functions, and is a popular choice in Bayesian optimisation.[137] The Matérn kernel is given by:

$$\Sigma_0(x, x') = \alpha_0 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\|x - x'\|\right)^{\nu} K_\nu \left(\sqrt{2\nu}\|x - x'\|\right), \tag{2.43}$$

where ν is the smoothness parameter, K_ν is the modified Bessel function of the second kind, and Γ is the gamma function (used to normalise the variance of the kernel). The parameter α_0 is a scaling factor which is used to control the overall scale of the covariance kernel, and is a hyperparameter of the kernel along with ν that can be optimised by the user.

2.2.1 Acquisition functions

To select the next point to evaluate based on the posterior probability distribution, an acquisition function is used. The acquisition function is used to determine the best point to evaluate based on both the expected value and the uncertainty of the posterior distribution. There are several acquisition functions which can be used, and I focus herein on the expected improvement (EI), the probability of improvement (PI), and the lower confidence bound (LCB).

Probability of improvement

As discussed above, the objective function is modelled by a GP surrogate which provides a posterior distribution represented as a normal distribution with mean $\mu_n(x)$ and variance $\sigma_n^2(x)$. The best observed value of the objective function so far is given by $f(x^*)$.

The probability of improvement measures the likelihood that the value of the objective function at a new point x will be lower than the best observed value $f(x^*)$. However, to encourage exploration, a small positive constant ξ is added to the best observed value, such that the probability of improvement is given by:

$$\mathrm{PI}(x) = \mathbb{P}(f(x) < f(x^*) - \xi), \tag{2.44}$$

where the value of ξ denotes the minimum improvement required to be considered for improvement. The variable $f(x)$ is standardised such that $Z \sim \mathcal{N}(0,1)$, where $Z = \frac{f(x)-\mu_n(x)}{\sigma_n(x)}$. By substituting this into the inequality in Eq. 2.44, one can write:

$$\frac{f(x) - \mu_n(x)}{\sigma_n(x)} < \frac{f(x^*) - \xi - \mu_n(x)}{\sigma_n(x)}, \tag{2.45}$$

and the predicted improvement is by the cumulative distribution function (CDF) of the standard normal distribution, Φ, such that:

$$\mathrm{PI}(x) = \Phi\left(\frac{f(x^*) + \xi - \mu_n(x)}{\sigma_n(x)}\right). \tag{2.46}$$

The simplicity of the PI does not come without cost, as it only considers the probability of improvement and not the magnitude, resulting in no distinction being made between large or small possible improvements.

Expected improvement

By contrast, the expected improvement (EI) considers both the probability of improvement and magnitude of improvement. It calculates the expected value of the improvement. First, consider that improvement is defined by:

$$I(x) = \max(0, f(x^*) - \xi - f(x)), \tag{2.47}$$

As the exact functional form of $f(x)$ is unknown, the expected improvement must be calculated to estimate the improvement. To do so, the posterior distribution of $f(x)$ is used, and the expected improvement is given by:

$$\mathrm{EI}(x) = \mathbb{E}[I(x)] = \mathbb{E}[\max(0, f(x^*) - \xi - f(x))]. \tag{2.48}$$

To evaluate the expectation, one must integrate over the PDF of the posterior distribution of $f(x)$, $p(y)$, such that:

$$\mathrm{EI}(x) = \int_{-\infty}^{\infty} \max(0, f(x^*) - \xi - y) p(y) dy, \tag{2.49}$$

but, as the integral is non-zero only when $y < f(x^*) - \xi$, the integral can be simplified to:

$$\mathrm{EI}(x) = \int_{-\infty}^{f(x^*)-\xi} (f(x^*) - \xi - y) \cdot p(y) dy, \tag{2.50}$$

once again substituting for Z and via solving the integral into closed form (omitted here for brevity, but a complete derivation can be found in Ref. 138). The resulting expression for the EI is given by:

$$\mathrm{EI}(x) = (f(x^*) - \mu(x) - \xi)\Phi(Z_{imp}) + \sigma(x)\phi(Z_{imp}), \tag{2.51}$$

where Φ is the cumulative distribution function of the standard normal distribution, and ϕ is the probability density function of the standard normal distribution, and Z_{imp} is the standardised improvement, given by:

$$Z_{imp} = \frac{f(x^*) - \mu(x) - \xi}{\sigma(x)}. \tag{2.52}$$

In the context of this work, the importance of EI is that it is capable of quantifying the magnitude of expected improvement, and thus favour values of x where the magnitude of improvement is large, which is preferable for finding the global minimum of the loss function (here, the numerical error on energy and force labels upon a test dataset).

Lower confidence bound

The lower confidence bound (LCB), unlike the EI or PI, does not consider $f(x^*)$, but instead uses the lower bound of the credible interval of the posterior distribution of $f(x)$:

$$\mathrm{LCB}(x) = \mu(x) - \kappa\sigma(x), \tag{2.53}$$

where κ is a constant which is used to control the relative weighting of the uncertainty of the posterior distribution. In practical use, high values of κ result in exploration being favoured, where low values of κ result in exploitation of low predicted values of $\mu(x)$.

In `scikit-optimize`, the package used to implement Bayesian optimisation in this work, the default acquisition function is a hedge function which is a weighted combination of EI, PI, and LCB dependent on the values of each for a given set of observed data.[139]

2.3 Summary

In this chapter, I introduced the methods underpinninng the work presented in this thesis. I have outlined data generation approaches, MLIP architectures which are trained on datasets generated using said methods, and described the basis of the Bayesian Optimisation that will be used to optimise hyperparameters for these MLIPs architectures. Together, these methods form the foundation of the software I introduce in the next chapter to enable the process of cross-platform hyperparameter optimisation for MLIPs, aiming to improve the accuracy and robustness of MLIPs fitted in the field.

Chapter 3

Cross-Platform Hyperparameter Optimisation for MLIPs

Acknowledgements

The work described in this chapter has been adapted from work published in *Journal of Chemical Physics* in 2023 (Ref. 1), from which some figures have been re-used, where this occurs it is clearly marked. I am grateful to Dr. Aidan P. Thompson (Sandia National Laboratories) for his insights on our work on optimising (q)SNAP potentials. I thank J. L. A. Gardner (University of Oxford) and Dr. J. D. Morrow for their insightful comments on validating MLIPs systematically. The initial version of the XPOT package (in the state described in this chapter), as well as models and datasets from this work are available at `https://github.com/dft-dutoit/XPOT/releases/tag/v1.0.0`.

3.1 Introduction

Machine learning approaches to interatomic potential fitting are now widely used in computational chemistry.[140–144] MLIPs provide surrogate models for quantum-mechanical (QM) potential-energy surfaces at a fraction of the cost, unlocking longer-timescale and larger-lengthscale simulations than would be accessible with direct QM methods, while maintaining comparable accuracy. As a result, there has been an influx of researchers aiming to use MLIPs to simulate materials and molecules, with applications ranging from the structure of disordered solids[43,145] to modelling nuanced effects such as anharmonic phonons[146] or non-collinear magnetism in iron,[147] as well as materials not yet synthesized.[148]

Many MLIP architectures have been developed, from the early Behler–Parrinello

neural-network[51] and Gaussian Approximation Potential (GAP)[52,101,149] models to more recent atomic cluster expansion (ACE) potentials,[61,118,133] as well as graph-neural-network architectures.[62,129,150] Each of these methods has different characteristics: ACE models are fast at inference, but lack the state-of-the-art accuracy of graph-based message-passing neural networks, whose inference (prediction) is still relatively expensive for large systems.[3,66,84,133]

In fitting models, there are two key "practical" components: the data, and the model. As discussed in Chapter 2, there are a number of ways to generate data, and there exists a large corpus of work on efficient and automated dataset generation.[89,90,107,151]

Many machine-learned interatomic potentials (MLIPs) are fit using hand-selected hyperparameters, based on a researcher's knowledge of and experience with a particular model architecture. This leads to research groups often specialising in a handful of model architectures, or in some cases only one, such that tailoring the choice of model architecture to the project is a relatively uncommon process. Despite this, each year, new MLIP architectures are developed; attempting to implement the newest insights to improve the accuracy and efficiency of atomistic machine learning techniques.

In 2020, Zuo *et al.*[81] published a comparison of popular MLIP architectures available at the time, assessing their accuracy and inference costs. This consisted of benchmarking on six different elemental systems: Ni, Cu, Li, Mo, Si, and Ge across five different MLIP architectures: GAP,[52] SNAP,[119] qSNAP,[130] Behler–Parinello NNPs,[140] and MTPs.[126] Hyperparameters were optimised, but the nature of the optimisation and loss function was not specified. The work described the Pareto frontier between accuracy and cost for each system and potential type, exploring the relative performance of the potential types across a number of (relatively small) elemental datasets. More recently, Leimeroth *et al.* published a user-focused overview of popular MLIP architectures, discussing the strengths and weaknesses of each ar-

chitecture, and how to choose the appropriate architecture dependent on the specific use-case.[84]

In this general context, I determined the need for a tool that would allow easier and more efficient model fitting across a range of MLIP architectures. While hyperparameter optimisation is not new to machine learning, the effective, consistent, and reproducible optimisation of MLIPs had not been established. Work from Poelking *et al.* had used grid-based Bayesian techniques for MLIP fitting, but did not interface directly to individual architectures, and required significant re-writing of mathematic formulations in the event of the user wanting to optimise an architecture not supported out of the box in their package, BenchML.[96] Other approaches included writing custom scripts to interface individual methods to existing optimisation packages, such as in the case of DAKOTA,[130] where a genetic algorithm was used to optimise the relative weightings of configuration types in a training dataset for carbon. However, these approaches neglect to provide a single, consistent interface for optimising MLIPs fitted by existing implementations, and are not easily extendable to new architectures.

I set out to design a plug-and-play, architecture-agnostic, hyperparameter optimiser for MLIPs. I introduce here the Cross-Platform Optimiser for Potentials (XPOT) which aims to unify the hyperparameter optimisation process for fitting packages across MLIP architectures in computational chemistry. I approach the problem by adhering to the following key principles:

(i) **Treat models as black-box, to be architecture agnostic:** Training methods, descriptor forms, and regressors can vary significantly between MLIP architectures, and to ensure fair and useful optimisation across a number of architectures, there must be no assumptions made about the form of these components.

(ii) **Interface to fitting packages, not directly to the model architecture:** In order to best provide an up-to-date and consistent experience, XPOT needs

to interface to fitting packages, not implement its own fitting for each model architecture, so that models are directly useful for MD without needing to have a separate interface to popular molecular dynamics engines.

(iii) **Ensure reliable, reproducible results:** The optimisation process should be reproducible, and further, resuming optimisation runs should be possible to allow iterative improvement.

(iv) **Allow user extendability:** One key to providing value to the community is to ensure that it is possible for users to extend packages to their specific needs, ideally with minimal effort, and without them needing to re-write any existing code, such as altering the form of the loss function, or implementing a new model architecture.

(v) **Handle both continuous and discrete parameters:** Hyperparameters can come in many forms (e.g., continuous or discrete), and ensuring flexibility across the range of hyperparameters which can exist for MLIP architectures is important, so that studying all hyperparameters is possible via optimisation.

(vi) **Do not require significant setup for High Performance Computing (HPC) systems:** HPC systems often have different modules, compilers, and schedulers, so XPOT should only require a python environment and avoid interfacing directly with scheduler systems, unlike fully-automated MLIP development tools.[94,152]

3.2 A Cross-Platform Optimiser for Machine Learning Interatomic Potentials

The cost of training a model is a setup cost, i.e., once an MLIP is trained, including any iteratative process, and validated on a dataset for a system, it can then be used for any number of simulations upon systems which it has been trained to

describe. Often, models are trained via trial and error, grid search, or heuristic optimisation, resulting in a large number of models being trained and validated, without any guarantee that the next model might be better than the last. To avoid the pitfalls of human-driven trial and error, I introduce an optimisation tool, XPOT, which can be used to deterministically select hyperparameters for a given model architecture and dataset. Hyperparameter optimisation, however, still necessitates fitting multiple models, but one can justify this cost in the context of common use cases for MLIPs: large-scale simulations. If a model is to be used to drive large production simulations, even a 10% reduction in inference cost can offset the cost of optimisation during training.

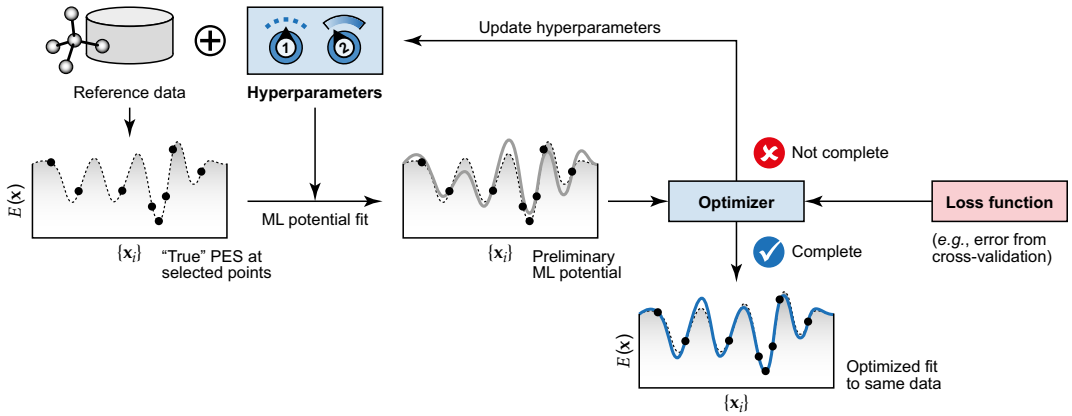

Figure 3.1: Workflow diagram of XPOT. The system takes a training dataset and model architecture as input, then performs hyperparameter optimisation to produce an optimised MLIP. The optimisation process involves iterative model training, evaluation, and parameter adjustment based on the defined loss function. Reproduced from Ref. 1, Original figure published under the terms of the Creative Commons CC BY license.

Fig. 3.1 provides an overview of the XPOT workflow. XPOT acts upon a training dataset, model architecture, and hyperparameter space determined by the user. Hyperparameter space here refers to the N-dimensional space of hyperparameters within the ranges defined by the user. The user can specify stopping criteria for the optimisation, and the optimisation will then continue until at least one of the stopping criteria is met. In Fig. 3.1, the workflow is shown as acting on a static training dataset, which is necessary to isolate the hyperparameter effects from the

effects of altering the training dataset. However, in most cases the fitting of an MLIP is an iterative process, whereby several iterations of training dataset are created. In this context, XPOT is not envisaged as a tool that is run at every iteration of a training dataset, but rather on a representative initial dataset and/or a late-stage converged dataset. This assumes that "good" hyperparameter settings transfer from one iteration of a database to the next, an assumption which is supported by the results in Sections 4.6 and 4.7. However, further investigation of the transferability of hyperparameters between datasets is a promising topic for further experimentation.

To create an optimisation tool for atomistic MLIPs, I not only need to select an optimisation routine, but also specify a loss function (Fig. 3.1). These choices will determine the efficacy, robustness, and speed of the optimisation, and I now introduce these problems in more detail.

3.2.1 Validating Potentials in Optimisation

To optimise hyperparameters for an MLIP (or indeed any other model), one must define a representative loss value, $\mathcal{L}$, that quantifies the quality of a model for the current choice of hyperparameters, $\mathcal{H}$. Hence, the central task for the optimisation is then:

$$\underset{\mathcal{H}}{\operatorname{argmin}} \quad \mathcal{L}(\mathbf{D}_{\text{val}}, \mathcal{H}), \tag{3.1}$$

where $\mathbf{D}_{\text{val}}$ is a set of validation data that are not seen in training.

When optimizing for a single objective (say, the prediction of atomisation energies for molecules in the QM9 dataset[153,154]), the definition of $\mathcal{L}$ is straightforward: it will simply be given by the root-mean-square error (RMSE) or a similar error metric for the individual energy predictions on the test set. The issue is more complex for interatomic potentials, where both energy and force errors are relevant and not typically directly proportional to each other.

To take a simple example, one can consider a loss function which only considers the RMSE of the force components. If the validation set consists of only fully relaxed

crystalline structures where no atoms experience any forces, the loss function will only measure the model's ability to predict force components of 0. A perfect loss value can thus be achieved by a potential which always returns 0, or by a potential which perfectly predicts the underlying PES. As such, this loss function does not provide any information about whether one potential is better or worse than another, as it does not consider the potential's ability to predict how forces change as atoms move, or the ability of the potential to predict the energy of a system. This is an extreme example, but it highlights the importance of both a well-chosen loss function and validation dataset. Firstly, I discuss the formulation of a loss function for XPOT, before discussing considerations for validation dataset choices for hyperparameter optimisation routines.

Loss function for MLIPs

When choosing a loss function to drive an optimisation routine, it is important to consider the properties of the loss function used in fitting each individual potential too. Of note, the loss function for XPOT acts upon the validation dataset, after fitting, and is calculated across hyperparameter space, where the loss function used in fitting acts upon the training dataset, and is determined by the fitting package (e.g., pacemaker, fitsnap, etc.), and may have a different form to the loss function used to optimise the hyperparameters. If the loss function of the fitting method and the loss function of a hyperparameter optimiser are not aligned in their task, the ability of a model with the given hyperparameters to learn is not being tested, but instead the alignment of the loss functions becomes a significant factor in the optimisation process.

Thus, to provide the most robust loss function to provide a sensible default, I studied the fitting process of the most popular MLIP packages. In particular, almost all loss functions consider both the energy and force components of the potential[62,63,118,127,133] and for some fitting packages, loss functions may also incorporate the stress components of the structures (e.g., `MACE`[62] and `gap-fit`[52,155]), and

regularisation terms (e.g., `pacemaker`[118]).

When developing XPOT, a key requirement was that the default loss function will provide a robust optimisation routine which can work across a number of varying qualities of input. Therefore, I approach the default loss function by prioritising alignment with existing model fitting frameworks, and only assuming the labelling of structural energies and per-atom force components.[62,118,129] The resulting generic form of the loss function for validating models in XPOT is:

$$\mathcal{L} = \alpha\mathcal{L}_E + (1-\alpha)\mathcal{L}_F \tag{3.2}$$

where $\mathcal{L}_E$ is the energy loss and $\mathcal{L}_F$ is the force loss, and α is a weighting factor between 0 and 1.

Error metrics for $\mathcal{L}_E$ and $\mathcal{L}_F$

Now that the overall form of the loss function has been defined, the individual energy and force error terms must be defined. Both root-mean-square error (RMSE) and mean absolute error (MAE) are commonly used values for reporting the accuracy of MLIPs. However, as MAE is non-differentiable at 0, it can cause problems for gradient-based optimisation routines and thus fitting algorithms. At the time XPOT was designed in 2021, the Huber loss was not a commonly used error metric by MLIP fitting packages, although it is now used in several,[66,129] and combines the differentiability of the RMSE with the reduced outlier sensitivity of MAE, achieved by transforming from MSE to MAE above a certain value.[156] For XPOT, I select RMSE as the default loss metric.

RMSE is more sensitive to outliers than MAE, and is a more conservative metric, ensuring that if the hyperparameter space has been chosen poorly (resulting in large areas with poor performance or overfitting), the increase in the loss function will be more pronounced, in turn being more selective in exploration of hyperparameter space. Of course, this does not come without downsides, and one such downside is

that if there are outliers in the validation set which are not learned across the full hyperparameter space, then the loss will be dominated by the contribution of these outlier errors to the loss function (Fig. 3.2). In the latest development version of XPOT, Huber loss is also implemented, but its performance in optimising hyperparameters has not been sufficiently tested at this time, and the implementation is not used in any of the work in this thesis. I include functionality to use MAE as the loss metric in XPOT, in case the user prefers to use it for the loss surface.

Now that an error metric has been decided upon, I approach the use of these metrics with regard to energies and forces.

Quantifying $\mathcal{L}_F$

Force components are calculated per-atom for both DFT and MLIPs. As such, individual predictions can be directly compared to the ground truth. I define the force loss in XPOT as:

$$\mathcal{L}_F = \sqrt{\frac{1}{N_{\text{at}}} \sum_{j \in \mathbf{D}_{\text{val}}} \left| \hat{\mathbf{F}}_j - \mathbf{F}_j \right|^2}$$

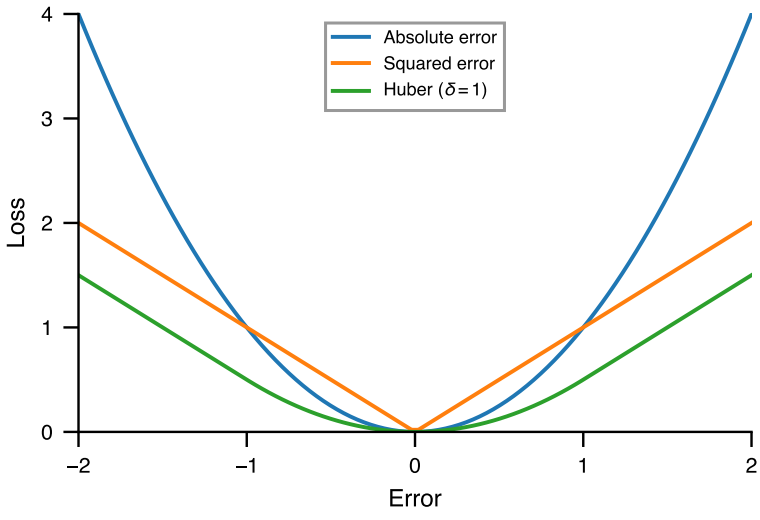

Figure 3.2: Error metric visualisation. A comparison between the squared error, absolute error, and Huber-defined error metrics.

where the index j refers to an individual atom and N_{at} is the total number of atoms in the validation set $\mathbf{D}_{\text{val}}$. $\hat{\mathbf{F}}_j$ indicates the ML model prediction of the force vector, rather than an individual component of the vector, such that there are N_{at} predictions in the sum.

Quantifying $\mathcal{L}_E$

Unlike with forces, where reference and MLIP labels are directly comparable, MLIPs predict per-atom energies, while reference energies derived from quantum mechanical calculations are calculated for the entire structure, and the quantities are not directly comparable. Comparison must be made between the total energy of the MLIP (by summing the per-atom energies) and the reference energy. However, this in turn means that the error of the model's predictions are also summed, and so the error of the model's predictions are not independent of the size of the system being validated.

In work by Morrow *et al.*,[102] the authors investigated the validation of MLIPs from literature, and discuss the importance of carefully considered numerical and physical validation. Via experimentation, they show that in order to correctly account for size-dependency in errors, the energy errors must be scaled by $\sqrt{N}$ where N is the number of atoms in the cell (Fig. 3.3).

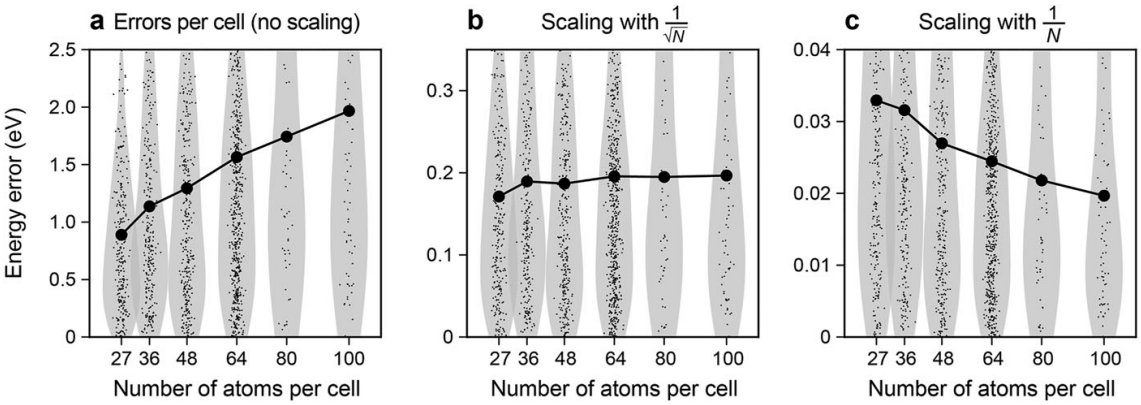

Figure 3.3: Scaling of energy error with system size, using the C-GAP-17 potential and testing dataset.[54] Plots are of the energy error of a potential as a function of system size, (a) no scaling, (b) scaled by $1/\sqrt{N}$, and (c) scaled by $1/N$. Raw errors are plotted using small symbols, with corresponding distributions shown with grey violin plots. The mean errors are plotted in black above these. Figure is reproduced with permission from Ref. 102, Original figure published under the terms of the Creative Commons CC BY license.

To justify this, I here derive the reasoning behind this finding. Let us consider a system of N atoms, with per-atom energies ϵ_i and errors δ_i. Assume that the error of the model's predictions per-atom are independent and random variables, with a mean close to 0 (i.e. the model errors are unbiased), and have variance σ^2. This assumption is true for well-trained MLIPs which have been fit to a representative dataset and have isolated atom energies that are zeroed, i.e., where errors are centred around the reference values.

Remembering that to predict the total energy we sum the per-atom energies: $E = \sum_i^N \epsilon_i$, I can consider the total error of the energy prediction: $E_{\text{err}} = \sum_{i=1}^N \delta_i$. As the errors are random and unbiased:

$$\text{Var}(E_{\text{err}}) = \text{Var}\left(\sum_{i=1}^{N} \delta_i\right) = N\text{Var}(\delta_i) = N\sigma^2$$

is true. The standard deviation, which characterises the magnitude of the error, is proportional to $\sqrt{\text{Var}(E_{\text{err}})}$, and so the standard deviation of the total energy is $\sigma\sqrt{N}$. Thus, to obtain a metric that is independent of the size of the system being validated, I must scale the energy error by $1/\sqrt{N}$.

Taking this into account, the XPOT energy loss is defined as:

$$\mathcal{L}_E = \sqrt{\frac{1}{N_{\text{cells}}} \sum_{i \in \mathbf{D}_{\text{val}}} \left(\frac{\hat{\mathbf{E}}_i - \mathbf{E}_i}{\sqrt{n_{\text{at},i}}}\right)^2} \tag{3.3}$$

where N_{cells} corresponds to the number of structures (cells) in the dataset, $n_{\text{at},i}$ is the number of atoms in cell i, and $\hat{\mathbf{E}}_i$ is the energy prediction for the i-th cell in the dataset.

Consequently, the total loss function is written as:

$$\mathcal{L} = \alpha\sqrt{\frac{1}{N_{\text{cells}}} \sum_{i \in \mathbf{D}_{\text{val}}} \left(\frac{\hat{\mathbf{E}}_i - \mathbf{E}_i}{\sqrt{n_{\text{at},i}}}\right)^2} + (1-\alpha)\sqrt{\frac{1}{N_{\text{at}}} \sum_{j \in \mathbf{D}_{\text{val}}} \left|\hat{\mathbf{F}}_j - \mathbf{F}_j\right|^2} \tag{3.4}$$

and both α and the loss is dimensionless as I divide both the energy and force loss terms by their respective units.

3.2.2 Validation and Testing Data

Now that the loss function has been defined, the data upon which it acts must be considered in order to provide a loss value which can be used to quantify the performance of a given model. A common approach in ML research is to split the data into training and validation sets, using k-fold cross-validation (Fig. 3.4a).[157] I included both k-fold cross-validation and external dataset validation in the XPOT package (Fig. 3.4b).

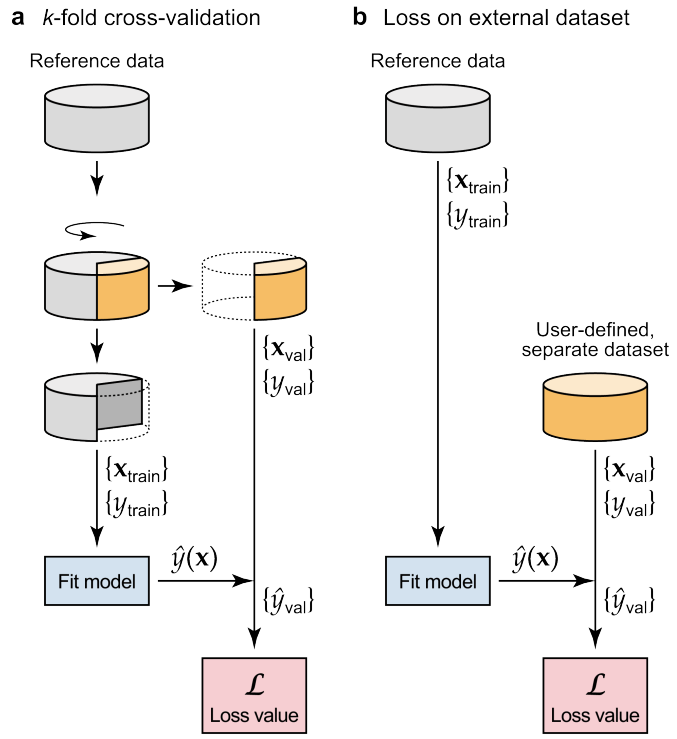

Figure 3.4: The two validation dataset implementations included with `XPOT` v1. (a) k-fold cross-validation. (b) A targeted validation dataset defined by the user. Figure reproduced from Ref. 1, Original figure published under the terms of the Creative Commons CC BY license.

There are two motivations for using an external validation dataset instead of k-fold cross-validation. On one hand, early on in the development process of an MLIP, the training dataset may be too small and insufficiently diverse to allow for useful cross-validation. On the other hand, in situations where the training dataset

is large and diverse, the cost of k-fold cross-validation can be prohibitive. In either case, an external validation dataset can be used to provide an accurate measure of the performance of the model. For the task of validation, an external dataset could either be computed by the user, or sourced from existing structure libraries, such as SACADA[158] for carbon allotropes, or the Materials Project[159] for a wide range of solid-state materials.

The choice of validation structures is important: if their selection, and thus the loss function, is not representative of the properties required for the task at hand, then the minimum found by optimisation will not necessarily be useful when applying the model in practice (e.g., in large-scale MD). I investigate the relative performance of validation datasets and 5-fold cross-validation in Section 3.3.2.

3.2.3 XPOT Implementation

An overview of the hyperparameter optimisation process in XPOT is shown in Fig. 3.5, visualising XPOT's role as an interface between well-established tools for MLIPs and optimisation.

Without XPOT, the workflow for optimising an MLIP typically requires bespoke scripts for each model architecture, or explicit creation of the model architecture within an optimisation package. XPOT provides an optimisation implementation that is agnostic to model architecture, and avoids the need for complex re-implementation of a model architecture by interfacing directly to the fitting package, avoiding assumptions about the model form, and treating it as a black box.

The user first specifies the fitting software, hyperparameters to optimise, and training and validation datasets via a JSON file. Next, the user selects the optimisation parameters: contributions to the loss function (α), number of iterations, stopping criteria, and any further specifications. These are specified as a Python dictionary, passed to the optimiser class. Then, the optimisation is run via python script, with each fit being run serially, such that the outcome of a fit determines the next optimisation. Outputs include tables of per-atom energy and force errors, as

well as hyperparameters and loss values.

On to the choice of optimisation strategy: as mentioned above, XPOT is designed as architecture agnostic, and thus does not assume any information about the form of the model it is optimising. The optimiser will only be provided two pieces of information: the inputted hyperparameters, and the outputted loss value.

As such, I selected an optimisation strategy well-suited for the black-box nature of this problem: Bayesian optimisation. Bayesian optimisation (BO) is a global optimisation strategy that is well-suited to the high-dimensionality of hyperparameter space and the black box nature of the problem at hand, as discussed in Section 2.2. [160] Furthermore, as the costs of fitting MLIPs and evaluating their errors are significant, the costs associated with Bayesian optimisation are not prohibitive in this case, with optimisation runs included in Section 3.3.2 spending less than 1% of the

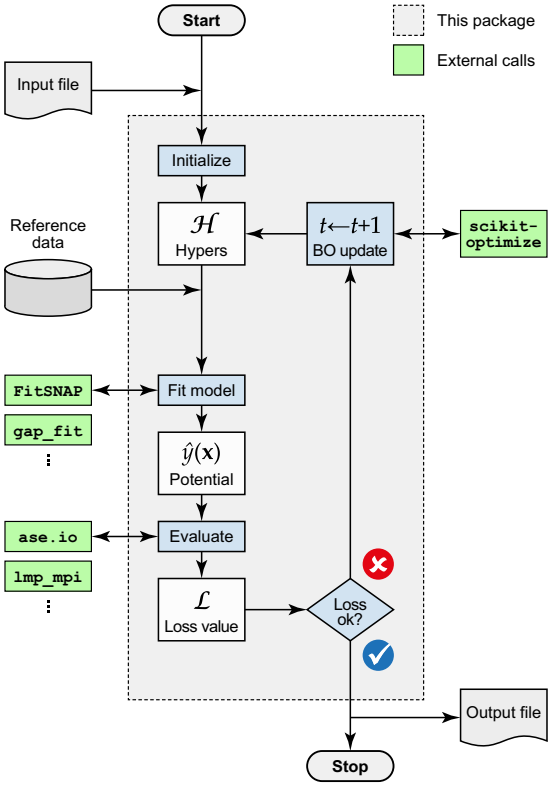

Figure 3.5: Operational Flowchart for XPOT. The software connects a number of tools into a single workflow, allowing for the optimisation of hyperparameters for a range of MLIPs in a consistent manner. Figure reproduced from Ref. 1, Original figure published under the terms of the Creative Commons CC BY license.

computational resources on the optimisation tasks.

The Bayesian optimiser is implemented by interfacing with the popular open-source Python library, `scikit-optimise`.[139] The choice of an external optimisation routine was taken in order to ensure long-term stability without requiring significant re-writing of XPOT, and to easily implement a range of initialisation, acquisition, and modelling strategies. In the present work, the initialisation strategy is pseudo-random sampling, specifically via Hammersley sequence.[161]

In related work, Stuke *et al.*[162] demonstrated the favourable performance of random searches and BO as the dimensionality of $\mathbb{H}$ (hyperparameter space) increases, especially against ordered "grid" style searches. Additionally, work on optimizing hyperparameters for atomistic neural-network models showed further improvement when using Centralised BO (CBO) over random-search methods[163].

The `scikit-optimise` implementation of BO used herein offers two key features which were important in the development of XPOT: firstly, the ability to handle discrete hyperparameters through the `skopt.space` module, and secondly, using a Gaussian Process to model the loss surface (Section 2.2), providing an accurate and robust estimation of not only the expected form of the loss surface, but also the uncertainty of the loss surface prediction at any given point in hyperparameter space. These two features allow for a truly robust and extendable optimiser, capable of powering user-driven tuning of the exploration and exploitation of hyperparameter space.

3.3 Benchmarking XPOT

The initial XPOT release included interfaces to GAP and (q)SNAP. These potentials sit at the extremes of the performance spectrum of potentials used in Zuo *et al.*[81], and were used to demonstrate the performance of XPOT. In Table 3.1, I summarise the hyperparameters for GAP and (q)SNAP relevant for the present work.

Those hyperparameters which can be considered convergence parameters (i.e., a larger value typically results in a better potential, albeit at the cost of increased

computational expense) are marked with asterisks in the "Conv." column in the table. These hyperparameters are not those XPOT was primarily designed to optimise (as they are not hyperparameters which can be used to increase performance without increasing cost) but are included for completeness. The hyperparameters which are expected to be most useful to optimise are those which do not have a known optimum — those that cannot be converged out. While Table 3.1 refers to the GAP and SNAP frameworks used in the present work,[52,119] XPOT has been extended beyond these fitting methods, as detailed in Chapter 4.

3.3.1 Performance

In this section, I demonstrate the functionality of XPOT by optimising a series of MLIPs for published datasets and by comparing the numerical performance of these re-fitted potentials to literature. I first optimise a series of qSNAP models for elemental systems, to show that XPOT can reach the same quality level as in a previous benchmark study,[81] and I then provide examples of re-fitting existing potentials with different fitting frameworks ("cross-platform" optimisation).

For the first test, I re-fitted qSNAP models to the datasets from Zuo *et al.*[81] and compared to the accuracies reported in that work. Fig. 3.6 shows two examples, and a comprehensive overview is provided in Table 3.2. After a preliminary low-cost sweep of values, α values of 0.7–0.9 were chosen to correctly balance energies and forces for fitting (Eq. 3.2). The qSNAP fits were optimised over 150 iterations. Further optimisation and more extensive initial exploration improved accuracy further, but improvements diminished beyond 300 iterations.

Fig. 3.6 illustrates the effect of using a combined energy and force loss during optimisation, most notably visible in the results for Cu. Whilst the loss value (lower panel) decreases, the constituent energy and force errors sometimes increase for one while the other decreases, emphasizing the importance of a well-chosen α value to representatively balance the contributions of the two in accordance with the user's priorities. This is an interesting result, as the force is a derivative of the energy, and

Table 3.1: Hyperparameters and their suggested default ranges in XPOT. I list hyperparameter settings that are relevant to the GAP and (quadratic) SNAP fitting frameworks discussed in the present work, respectively. Settings that are convergence parameters (i.e., will generally improve the model when increased) are marked with an asterisk ($*$) in the "Conv." column, and are therefore not optimised unless the user specifically requests it. Reproduced from Ref. 1, Original table published under the terms of the Creative Commons CC BY license.

		Description	Conv.	Default	Range
GAP	`cutoff`	Cutoff radius (Å)		4.0	2.5–8.0
	`atom_sigma`	Gaussian width for SOAP neighbour density (Å)		0.5	0.2–0.8
	`n_max`	Maximum n for radial basis functions (SOAP)	$*$		
	`l_max`	Maximum l for angular basis functions (SOAP)	$*$		
	`zeta`	Power to which the kernel is raised (SOAP)		4	2–6
	`n_sparse`	Number of representative points, M	$*$		
	`sigma`[a]	Regularisation for energies (eV at.$^{-1}$)		0.01	0.001–0.1
		Regularisation for forces (eV Å^{-1})		0.1	0.01–0.5
SNAP	`rcutfac`	Cutoff radius (Å)		4.0	2.5–8.0
	`twojmax`	(2×) Value of $J_{\max}$ used to formulate SNA descriptors	$*$		
	`rfac0`	Distance to angle conversion parameters		0.994	0.6 – 1.6
	`rmin0`			0	0–0.1
	`wj`	Weighting of each element in the fit		1.0	0.2–2.0
	`eweight`	Energy weighting for input data		100	10–10^5
	`fweight`	Force weighting for input data		1	0.1–100

[a]Input is given in the format `{energy:force:stress:hessian}`.

Table 3.2: Energy and force component RMSE values for a series of testing datasets for elemental systems, provided in Ref. 81. Reproduced from Ref. [1]. Original table published under the terms of the Creative Commons CC BY license.

	E RMSE (meV at.$^{-1}$)			**F** RMSE (eV Å^{-1})		
	Ref. 81		XPOT	Ref. 81		XPOT
	GAP	qSNAP	qSNAP	GAP	qSNAP	qSNAP
Ni	0.62	1.04	0.80	0.04	0.07	0.04
Cu	0.56	1.16	0.52	0.02	0.05	0.03
Li	0.63	0.85	0.59	0.01	0.04	0.01
Mo	3.55	3.96	3.51	0.16	0.33	0.17
Si	4.18	6.28	3.31	0.12	0.29	0.11
Ge	4.47	10.55	3.74	0.08	0.20	0.11

one might expect that minimising one would also minimise the other. However, this proves not to be the case, and highlights the importance of α in the loss function. This is especially true for models with independent prediction heads (i.e., where the forces are not derived from the energy predictions, but are predicted separately). The plots also illustrate that XPOT occasionally finds good hyperparameters in the initialisation section (grey). This is highly dependent on the chosen search space and sampling method, and some element of luck determining whether one of the sampling points lands close to a minimum on the loss surface. In these cases, as seen in Fig. 3.6, there is less improvement to be gained by further optimisation (Si, left) compared to other systems (Cu, centre).

More generally, I observe that the XPOT-optimised qSNAP models compare favourably to qSNAP and GAP results by Zuo *et al.*,[81] especially for Mo and Si (Table 3.2). The qSNAP potentials fitted in Ref. 81 were fitted in collaboration

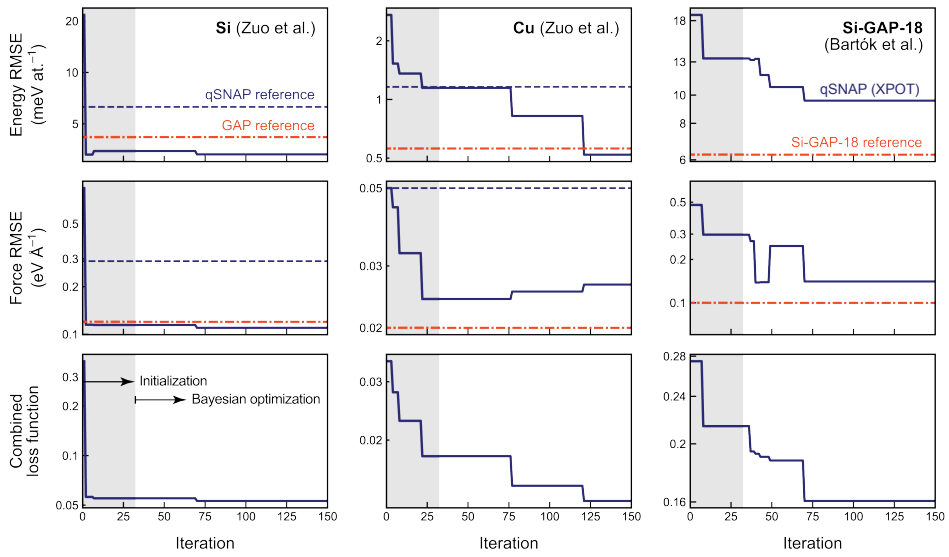

Figure 3.6: Illustrative examples of optimisation runs with XPOT, shown for three series of representative qSNAP models fitted to existing datasets. The figure shows the convergence of numerical error measures for sequentially optimised qSNAP models, evaluated for the Si and Cu datasets from Zuo *et al.*[81], as well as the structurally much more complex Si-GAP-18 reference dataset from Bartók *et al.*[55]. Blue (orange) dashed lines represent the errors for qSNAP (GAP) models from the cited papers, respectively. The grey shaded areas indicate the initialisation stage (here carried out using a Hammersley sequence[161]). Figure reproduced from Ref. 1. Original figure published under the terms of the Creative Commons CC BY license.

with the developers of the FitSNAP code, ensuring that the authors had sufficient familiarity with fitting qSNAP models. The main qualitative conclusion to be drawn from Table 3.2 is therefore that XPOT reaches good performance across a variety of chemical systems, and that in fitting my first qSNAP models, XPOT enables potentials fitted in this work to surpass the accuracy of the developers' own models.

For the second test, I moved to a more widely applicable, "general-purpose" MLIP with a highly complex reference dataset (more than one hundred thousand atomic environments), vis. Si-GAP-18 (Ref. 55). XPOT was used to fit optimised qSNAP models to the training dataset. In this case, I match exactly the train–test split and use the published testing set for validation (rather than cross-validation), for direct comparability with the original potential[55]. The results are shown on the right-hand side of Fig. 3.6: in this case, our optimised qSNAP model reaches an energy RMSE of about 10 meV at.$^{-1}$ (compared to 6 meV at.$^{-1}$ for the GAP of Ref. 55), and a force RMSE of about 0.15 eV Å^{-1} (compared to 0.10 eV Å^{-1}). I view this as a representative example of re-fitting, with a different MLIP framework, to the dataset of an already highly optimised model (here, Si-GAP-18) — one use-case could be to use a GPU implementation that is not available for the published model, or more generally to gain evaluation speed at the cost of some loss of accuracy.

Table 3.3: Performance of potentials fitted to the C-GAP-17 dataset.[54] Energy per atom and force errors are included, as well as MD benchmark results. Errors are calculated on the C-GAP-17 testing set.[54] All simulations were carried out on the same CPU architecture, and the speed is given relative to the C-GAP-17 model (4.4 steps/s). Reproduced from Ref.[1]

		E RMSE (meV/at.)	**F** RMSE (eV/Å)	MD speed (relative)
linear SNAP	($J = 3$)	192.2	4.34	65
	($J = 4$)	117.6	2.43	24
	($J = 5$)	76.8	1.68	9
	($J = 6$)	63.2	1.40	4
qSNAP	($J = 3$)	50.8	1.26	38
	($J = 4$)	41.4	1.00	14
	($J = 5$)	34.9	0.89	5
	($J = 6$)	28.8	0.81	2
C-GAP-17	(Ref. 54)	42.0	1.26	1

To expand the range of tests further, I fitted a series of linear and quadratic SNAP MLIPs to the C-GAP-17 dataset for elemental carbon.[54] C-GAP-17 is an example of a potential that has been (manually, at the time) optimised for the description of liquid and amorphous phases; it has notoriously high numerical errors compared to more recent potential fits,[164,165] yet is already able to describe important physical aspects related to carbon.[39,106,166,167] I use this example here to measure the performance of models refitted at different levels, and in Section 3.3.2 below to explore the consequences of hyperparameter choices on structural predictions.

Table 3.3 compares the accuracy of linear and quadratic SNAP models across J values. The cutoff radius was fixed to 3.7 Å, as in C-GAP-17,[54] and the training and testing sets used were identical to those in Ref. 54. These potentials were fitted using $\alpha = 0.75$, similar to the qSNAP tests on data from Ref. 81, but higher than for those on the Si-GAP-18 dataset. The `rfac0` and `rmin0` hyperparameters were optimised for the SNA descriptors, as well as the energy and force weights, within the default ranges given in Table 3.1. All these were optimised in a single sweep to investigate the ability of BO (as implemented in XPOT) to survey large hyperparameter spaces.

XPOT-optimised linear SNAP models do not match the accuracy of the C-GAP-17 potential up to $J = 6$. In contrast, qSNAP models optimised with XPOT show promising numerical energy and force errors for $J = 4$ and higher. MD simulations run with the resulting potentials showed that as J increases so too does the computational cost of evaluation, as expected. The relative inference costs with respect to C-GAP-17 are reported in Table 3.3. The computational cost of moving from linear to quadratic fitting was significantly less than that of increasing J, and should be the first step taken to improve the accuracy of SNAP models.

While the cost of evaluation does not change by more than a factor of two upon moving from linear to quadratic SNAP models, the cost of fitting is increased further (approximately 3.7 times for $J = 5$ and 7.4 times for $J = 6$) and cannot be

significantly improved by increasing computational resources. This is because of a limitation with the solver used by `FitSNAP`,[135] which is single-core. Thus, although the calculation of the descriptors is parallelized, the fitting of the potential becomes the bottleneck.

3.3.2 Results

Physical effects of hyperparameters

I investigate the physical effects of hyperparameters in the context of SNAP models through study of optimisation and behaviour of the resultant MLIPs. The first approach is regarding the meaning of specific hyperparameters: these could be viewed as purely free optimisation parameters, or as physically constrained and informed ones.[101]

Amorphous carbon (a-C) presents a relevant challenge for fitting interatomic potentials, with ML or otherwise,[167] due to the diverse nature of its atomic structure. In the previous section, I have shown SNAP model fits with optimised hyperparameters, leading to lower numerical errors than those of C-GAP-17. The XPOT-optimised qSNAP models are computationally cheaper than C-GAP-17, and can leverage GPU resources, but the robustness of the potential suffers compared to C-GAP-17, which smoothly extrapolates beyond the structure space of the training dataset. This lends the qSNAP models to be useful for some applications, but not to be a general-purpose potential for use at high temperatures in MD without further iteration of the training dataset.

I used relatively flexible, and numerically accurate, qSNAP potentials with $J = 6$ in this case. When fitting a potential from scratch, using a larger dataset with a lower J value may provide a more cost-effective potential for extensive large-scale simulations, such as the qSNAP model by Willman *et al.*[56] ($J = 4$) for high-pressure carbon systems.

I find that the value of `rfac0` significantly alters the robustness of the optimised models. `rfac0` is a hyperparameter for the formulation of the bispectrum components. As explained in Ref. 119, the radial distance r is mapped onto θ_0 via

$$\theta_0 = \theta_0^{\max} \frac{r}{R_{\text{cut}}} \tag{3.5}$$

However, based on typical SNAP implementations, the equation can also be written as:

$$\theta_0 = r_0^{\text{fac}} \pi \frac{r - r_0^{\min}}{R_{\text{cut}} - r_0^{\min}} \tag{3.6}$$

where r_0^{fac} corresponds to the `rfac0` hyperparameter, $r_0^{\min}$ to `rmin0`, and R_{cut} is the cutoff radius.

A low value of `rfac0` directly reduces the amount of the 3-sphere that can be used for mapping the possible neighbour positions because points south of θ_0 are excluded. Thompson *et al.*[119] state that "it is advantageous to use most of the 3-sphere, while still excluding the region near the south pole where the configurational space becomes highly compressed". Thus, both too high or low values of `rfac0` can diminish accuracy and robustness of the resulting potential. For a-C, I observed values lower than 0.7 resulting in potentials becoming unstable in MD, despite not necessarily reducing numerical performance. I hence specified `rfac0` to be restricted to values of 0.7 and above. While it may be possible for potentials with low values of `rfac0` to be stable, in the case of models trained on the C-GAP-17 dataset, I was unable to perform 9,000 K NVT simulations without erroneous behaviour – even when using a 0.2 fs timestep, a factor of 5 shorter than that used for typical C-GAP-17 MD simulations.[39,54]

The optimised qSNAP $J = 6$ potential is stable at 9,000 K with a 0.2 fs timestep, and when replicating the short, small-scale melt-quench simulations reported by Deringer *et al.*[54] it predicted 53.5% sp^3 environments (Fig. 3.7b), compared to 57.5% by GAP-17, and 65.7% from AIMD simulations after the melt-quench simulations.

However, below 3000 K a 1 fs timestep is stable, as tested for simulations of up to 1 ns. The predicted radial distribution functions for 216 atom-cells showed good agreement with DFT and C-GAP-17 predictions for quenched a-C between $1.5\,\mathrm{g\,cm^{-3}}$ and $3.5\,\mathrm{g\,cm^{-3}}$. It should be noted that the length of the simulations here is limited by the AIMD reference simulations available for comparison, and that longer simulations may alter the predicted sp^3 content, and the wider structure of the final structure.

Next, I investigated optimizing the cutoff radius. I ran a second optimisation sweep, with the same ranges selected for optimisation, adding a variable cutoff. All hyperparameters optimised for a fixed cutoff were re-optimised in the variable-cutoff sweep. Here, the "best" potential was fitted with a cutoff of 3.49 Å, compared to the 3.7 Å used previously, and the energy and force weightings were altered slightly (albeit the energies of the dimer structures are lowest-weighted in both cases).

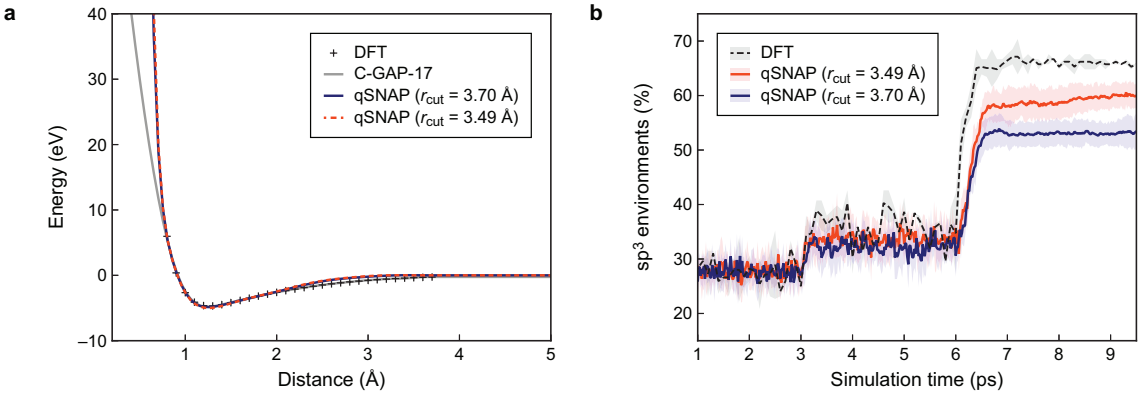

Figure 3.7: qSNAP model carbon analysis. (a) Potential-energy surface of an isolated C–C dimer, evaluated for XPOT-optimised qSNAP models compared to DFT and C-GAP-17.[54] (b) Percentage of atomic environments in an sp^3 (fourfold-connected) configuration during an MD simulation following the same protocol as Deringer *et al.*[54]: randomisation at 9,000 K (3 ps), followed by holding the liquid at 5,000 K for 3 ps, a rapid quench to 300 K (0.5 ps), and holding the quenched structure at 300 K (3 ps). The SNAP model characterised in blue was fitted with a fixed 3.7 Å R_{cut} (`rcutfac`); the model characterised in red is based on an optimisation run that included optimisation of the cut-off (final value: 3.49 Å). The shaded areas indicate standard deviations over 3 DFT (Ref. 54) and 10 (qSNAP, this work) separate runs, respectively. Figure adapted from Fig.[1]. Original figure published under the terms of the Creative Commons CC BY license.

Fig. 3.7 shows that the potential with optimised cutoff has almost identical behavior for the isolated-dimer curve compared to the fixed-cutoff potential, as too does the hybrid model. Interestingly, the optimised cutoff radius qSNAP better predicts sp^3 character in carbon MD. When repeating the tests from the previous section, the new potential predicts 60.0% sp^3 carbon environments, providing a better fit to AIMD results than even the C-GAP-17 potential. In Fig. 3.7b, I show these results, noting that for both qSNAP models the sp^3 content increased more slowly during quenching than for the AIMD simulations. To isolate the effects of altering the cutoff radius vs. the other hyperparameter changes I fitted a potential with $R_{\text{cut}} = 3.49\,\text{Å}$ with the other hyperparameters fixed to the optimised values in the fixed-cutoff potential. This potential predicted a structure with 56.9% sp^3 content, which sits between the fixed- and optimised-cutoff potentials. These findings suggest that optimisation of cutoff can be beneficial, especially alongside optimizing other hyperparameters, to optimise not only the numerical behaviour, but the physical behaviour of an MLIP.

Comparing the hyperparameters of optimised potentials across the datasets from Zuo *et al.*[81] (numerical results given in Table 3.2), I found that the optimal `rfac0` changes significantly from element to element, ranging between 0.5 (Cu) and 0.845 (Mo). Notably, not only does the optimal value of `rfac0` depend on the system, I observed that in the case of the Si GAP-18 dataset, the optimal `rfac0` and `rcutfac` values were different to that of the Zuo *et al.*[81] dataset for silicon. Specifically, `rfac0` was optimised to 0.87 (Si-GAP-18[55]) and 0.58 (Zuo *et al.*[81]), respectively. The optimised cutoff for these datasets also differed, vis. 4.68 Å vs. 4.23 Å. These findings emphasize that different types of reference data (e.g., liquid versus crystalline structures; general versus specialised datasets) require different hyperparameters for an optimised fit, and therefore they underline the usefulness of hyperparameter optimisation in the development of MLIP models for a variety of applications.

Balancing energies and forces in the loss function

As defined in Eq. 3.2, a combined loss function that includes energy and force error terms is implemented in XPOT, controlled by a factor α that takes values between 0 and 1. I investigated the effect of tuning this parameter, by optimizing a series of potentials from one where the loss only depends on force errors ($\alpha = 0$ in Eq. 3.2) to one where it only depends on energy errors ($\alpha = 1$). For this experiment, at each value of α I started 10 independent optimisation runs of qSNAP models with $J = 5$ for the Si-GAP-18 dataset[55] (as in Section 3.3.1).

Fig. 3.8a illustrates the requirement for combined optimisation of the energy and force errors, which is achieved through the selection of α. In optimizing for only forces or energies, I produced potentials which are skewed significantly to either force or energy, and are not accurate at predicting the respective other property.

Across the range $0 \leq \alpha \leq 1$, the energy RMSE decreases, and the force RMSE increases. However, the change is not linear, and for this particular system, the optimal value of α lies between 0.3 and 0.5. Not only do the errors change with varying α, but the standard deviation of the errors across the 10 optimisation runs also broadly decreases as the weighting of the corresponding part of the loss function is increased. For $J = 4$, I found this effect to be even further pronounced. Thus, a well-chosen α also increases reliability of optimisation, improving consistency *and* performance across optimisation sweeps.

Fig. 3.8b shows the optimised cutoff values for potentials after optimisations at each α. When studying the trend in optimised cutoff values, I see that as α increases, so too does the optimal R_{cut}. Interestingly, the best force errors are found when the cutoff is lower, and better energy errors are found where higher R_{cut} values are preferred. In testing, I found that potentials at intermediate α values show the most robust performance in melt-quench simulations, while models with very small α occasionally demonstrated non-physical behaviour.

Using either $\alpha = 0$ or 1 results in significantly less accurate potentials, and

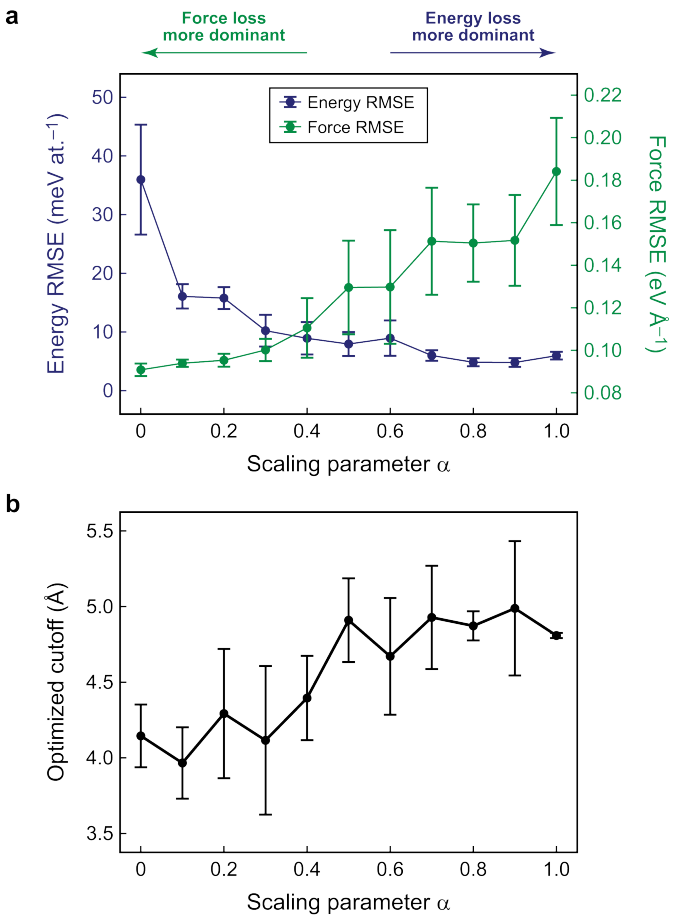

Figure 3.8: Effect of varying α. (**a**) energy/force RMSE vs. α and (**b**) optimised R_{cut} vs. α for qSNAP models on the Si-GAP-18 dataset. Figure reproduced from Ref. 1. Original figure published under the terms of the Creative Commons CC BY license.

neither should ever be selected in general use. To create well-balanced potentials, α should therefore be chosen to be mindful of over-weighting either energies or forces. In order to calculate α for a given optimisation run, one should calculate a desired accuracy for the energy and force error, and determine a value of α that would result in an even contribution of both to the loss function based on these values.

Role of the validation set

Having implemented cross-validation and external validation datasets, I explore whether cross-validation offers large enough of an improvement over external validation to justify the k times additional computational cost during hyperparameter optimisation, where k is the number of folds in the cross-validation.

In validating potentials, and thus in hyperparameter optimisation too, the validation dataset used plays a crucial role. If the validation set is overly structurally similar to the training data, it will not comprehensively assess the ability of the potential to describe "unseen" structures (extrapolate). In the case of C-GAP-17, the training and validation data are structurally similar because they have been obtained by a random split of the overall dataset.[54] I therefore tested whether a less correlated, or more generic, validation set might allow for efficient hyperparameter optimisation, emphasizing robustness over accuracy.

I optimised a-C potentials using two further validation sets aimed at varying levels of disorder: (i) a set of snapshots from very-high-temperature, 15,000 K MD simulations performed with C-GAP-17, and (ii) a set of fully randomized structures (generated via a hard-sphere random packing model). Here I created our own validation datasets aimed specifically at highly disordered structures, but existing external datasets such as SACADA[158] could also be used to guide hyperparameter optimisation for carbon (as long as the level of theory is consistent). The new datasets were labelled with DFT and used independently as validation sets for optimisation with XPOT. I then evaluated the errors of the optimised potential for the respective other datasets to study the effects of the validation set on the form and accuracy of the potential across different datasets. The results for qSNAP ($J = 5$) potentials are shown in Table 3.4.

Table 3.4: Energy and force RMSE values (meV at.$^{-1}$ / eV Å^{-1}) of MLIPs trained on the C-GAP-17 dataset when tested on three separate test sets. The validation set used for evaluating the loss function (cf. Fig. 3.4) is marked in bold.

	Energy / force RMSE on defined test sets		
	C-GAP-17 test set (meV/at.) / (eV/Å)	High-T MD (meV/at.) / (eV/Å)	Random packing (meV/at.) / (eV/Å)
qSNAP (5-fold cross-validation)	**35.5 / 0.85**	38.1 / 1.43	38.9 / 2.14
qSNAP ($\mathcal{L}$: C-GAP-17 test set)	**31.9 / 0.90**	36.6 / 1.47	36.8 / 2.27
q qSNAP ($\mathcal{L}$: High-T MD)	39.5 / 0.87	**38.0 / 1.32**	41.4 / 2.12
qSNAP ($\mathcal{L}$: Random packing)	194.8 / 0.90	47.1 / 1.36	**39.3 / 1.68**
C-GAP-17 (Ref. 54)	**42.0 / 1.26**	65.8 / 1.51	33.8 / 1.60

I demonstrate that the validation set chosen has a direct impact on the hyperparameters selected during optimisation, and, as a result, on the form and accuracy of the potential itself. The qSNAP model optimised against the random-packing dataset has very high energy errors on crystalline structures (which is to be expected as the atomic environments are so different). I attribute this to the fact that during optimisation, the model is guided towards improving the fit to the random-packing structures, without any contribution towards the loss of the errors on crystalline data. By studying the hyperparameters, I observe that this results in very high weighting on the more disordered structures in the training dataset (to minimise loss on the random-packing validation set), fitting a potential that is very poor at describing crystalline environments.

The high-T MD validation set provides generally robust and balanced results across the three testing sets; conversely, a purely random packing of atomic spheres is not informative for "real-world" applications. Comparing the optimised SNAPs to the original C-GAP-17 model, an interesting pattern emerges: C-GAP-17 overall has slightly higher errors, and notably the highest energy error on the high-T MD dataset, and yet it maintains reasonable accuracy when tested on the random-packing set – outperforming all the qSNAP models tested for this dataset, including the one specifically optimised with this set as validation. This greater variance in performance further reinforces the observation that our parametrisations of qSNAP are less robust than C-GAP-17, and are more sensitive to the choice of validation set, and that C-GAP-17 is robust outside of regular structures, consistent with the physically meaningful performance seen in previous applications (e.g., Caro *et al.*[39]).

3.3.3 Outlook

I described XPOT, a package to facilitate hyperparameter optimisation for MLIPs by providing interfaces to established MLIP fitting software. I have demonstrated practical applications of XPOT for cross-platform optimisation of MLIPs fitted to datasets of quantum-mechanical reference data. In a series of numerical experiments,

I explored the role of the loss function in this optimisation and how it affects the characteristics of the resulting potentials, with implications for the present software package and for atomistic ML more generally.

I have shown that potential fits with XPOT perform favourably compared to literature data.[81] A scaling factor, α, is used to balance the weighting of energy and force validation during optimisation, and the optimal α value is found to be not only dependent on chemistry but also structural make-up of the dataset. This requirement for manual user input remains one of the challenges in fully automating the fitting process. However, in being interfaced to a Bayesian optimisation code, XPOT can explore large search spaces: up to 11 hyperparameters were optimised simultaneously in fitting qSNAP models for carbon.

One of the most important use cases for XPOT is in optimizing those hyperparameters that do *not* significantly affect the evaluation cost of an MLIP: in this way, one can improve the quality of prediction with no additional computational cost at runtime. Other parameters do affect the cost – and in such cases, especially for general-purpose MLIPs and large-scale simulations, the extra time spent on hyperparameter optimisation is likely favourable over the cost one would incur with more expensive settings at inference time.

The results of this chapter suggest that optimisation of hyperparameters can help to close the gap between the accuracy of models with significantly different inference efficiency, and that this is a promising avenue for future work. In the following chapter, I apply XPOT to the ACE framework, an efficient MLIP architecture which promises improved accuracy and efficiency if correctly parameterised. Additionally, I further probe the alignment of the numerical errors of MLIPs to their physical errors, especially with respect to the role of the loss function (i.e., how well do numerical errors predict the physical behaviour of an MLIP?).

Chapter 4

Atomic Cluster Expansion Optimisation with XPOT

4.1 Acknowledgements

The work described in this chapter has been published in the *Journal of Theoretical and Computational Chemistry* (JCTC) in 2024, from which some figures and text (where labelled) have been re-used. Yuxing Zhou provided the GST-GAP-22 dataset, and advised on the composition of the dataset for training of the Sb_2Te_3 models. Additionally, he advised on optimal setups for crystalline growth simulations and performed DFT on the structures collected in Fig. 4.7 I thank Dr. Minaam Qamar, Dr. Anton Bochkarev, Dr. Yury Lysogorskiy, and Prof. Ralf Drautz (all at ICAMS, Ruhr University Bochum) for their insightful communications regarding ACE potentials. Further thanks go to Dr. Thomas C. Nicholas (now at Ghent University) for testing the code on complex, multi-element structures. The updated version of the XPOT package (in the state described in this chapter), as well as models and datasets from this work are available at `https://github.com/dft-dutoit/XPOT/`. The work on the application of XPOT-ACE models to GST and Te is published in the preprints in Ref. 4 and Ref. 5 respectively.

4.2 Introduction

In introducing XPOT in the previous chapter, I demonstrated how hyperparameter optimisation can help in optimising MLIP hyperparameters to improve their accuracy and efficiency. This approach is particularly useful where the model architecture used can be both sufficiently flexible to learn the nuances of the potential energy

surface, and the robust enough to avoid unphysical behaviour without requiring a prohibitively large number of training points.

In order to access simulations at the time and length-scales required to study processes such as Li-ion diffusion,[168] phase transitions,[40,43,169] or crystal growth,[5] it is necessary to fit models which are not only accurate, but also efficient for large-scale MD simulations.

Each ML architecture has its own unique strengths and weaknesses; for example, graph-based neural networks currently define the state-of-the-art for accuracy in materials modelling, but inference is still relatively expensive for large systems, with varying levels of inference parallelisation availability.[84] As such, I focus herein on non-linear ACE models fitted using the `pacemaker` software introduced by Bochkarev *et al.*,[118] which have demonstrated the ability to produce accurate, efficient, and robust models for a range of chemical systems[98,133,145,147].

In this chapter, I extend the XPOT framework to ACE models via the `pacemaker` package, and showcase the ability to retain knowledge of the chemistry of the system when moving between fitting frameworks. To do so, I fit ACE models to two datasets used for the fitting of state-of-the-art GAP models for disordered systems, without adding any further training data. I study the performance of optimised ACE models on a-Si and Sb_2Te_3 numerically and physically, and compare the results to not only the GAP models that were used to generate the training data, but also to an ACE model fitted by developers of the `julia-ACE`[134] software to demonstrate XPOT's ability to improve upon models fitted by experts with carefully selected hyperparameters. After validating that XPOT-optimised ACE potentials are able to reproduce the results of existing state-of-the-art models, I extend the application of XPOT to the optimisation of non-linear ACE models for the Ge–Sb–Te and Te systems in collaboration with Y. Zhou, and as published in Ref. 4 and Ref. 5 respectively.

4.3 ACE implementation into XPOT

The extension of XPOT for ACE models (using the `pacemaker` package) does not require any changes to the optimiser, or overall class structure in XPOT. The ACE implementation simply interfaces XPOT to the `pacemaker` package. This is because of the modular class system which XPOT uses, whereby the package is extended to the fitting software, but the core functionality remains the same. An overview of the methodology for fitting ACE models with XPOT is shown in Fig. 4.1, where the extension of the software is embodied by the arrows between the centre and left-hand panels.

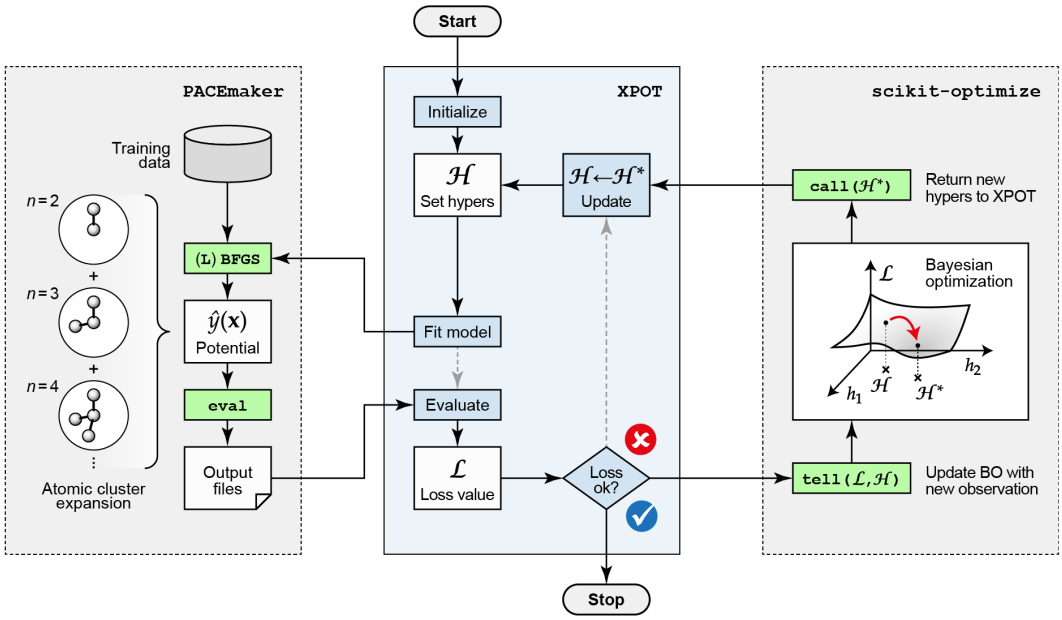

Figure 4.1: Overview of the methodology for automated optimisation of ACE potentials, showing a simplified flowchart of the computational tasks involved. The core functionality of `XPOT` is highlighted in the central box in blue, and external calls to `pacemaker` (*left*) and `scikit-optimise` (*right*) are indicated. Figure reproduced with permission from Ref. 2, Copyright 2024 American Chemical Society

By default, I implemented using the predictions outputted by `pacemaker` as input for the loss function, sidestepping the need for inference of the potential after each iteration. However, for custom loss functions that require results which are not contained within energy and force predictions (which are available directly from

the fitting software), users are able to provide their own error collection function (without needing to modify the rest of the pipeline) to ensure that the loss value can be calculated according to their specification. Beyond this, the only changes are made to the preprocessing of the input file, as well as the command to run the fitting itself. The rest of the class is inherited from the parent `MLP` class as with generalised methods. This class centralises all processes which are not dependent on the specific model architecture, including processing hyperparameter ranges from input files, setup and running of the optimisation loop over said hyperparameter regions, definition of the loss function, and logging of results and loss values.

4.4 Datasets

I now describe the datasets used for fitting and validating XPOT-driven hyperparameter optimisation for ACE models for Si and Sb_2Te_3. For details, I refer to the cited original publications; for an overview of validation techniques for ML potentials more generally, I refer to Morrow *et al.*[117].

Silicon

In the case of silicon, I use three different datasets (labelled with the same DFT parameters) across fitting and testing of the ACE models, in a similar manner as for carbon in the Chapter 3. First, I take the training dataset for the Si-GAP-18 general-purpose potential by Bartók *et al.*[55] as an example of a well-developed and largely handcrafted dataset. I fit ACE models to this dataset using XPOT, and use the corresponding test set for validation (both accessed in Ref. 55). Second, I use DFT-labeled snapshots from a GAP-MD simulation described in Ref. 170 (referred to as "MQ-MD" in the following). This dataset contains diamond-type supercells with vacancy defects, liquid, and amorphous structures, as well as transitions between these phases. The structures were generated via Si-GAP-18-driven MD and subsequently labelled with single-point DFT computations.[170] Validating on this dataset aims to explore the model's performance on physically-relevant structures

which are generated separately using a different protocol to the training dataset. The final dataset is made up of random structure search (RSS) configurations generated in Ref. 117. This "RSS" dataset contains a number of higher energy structures which are not derived from physical processes (e.g., melting, quenching, or pressurisation MD simulations) and provide a stern test for the robustness of models trained on hand-picked physically-motivated structures.

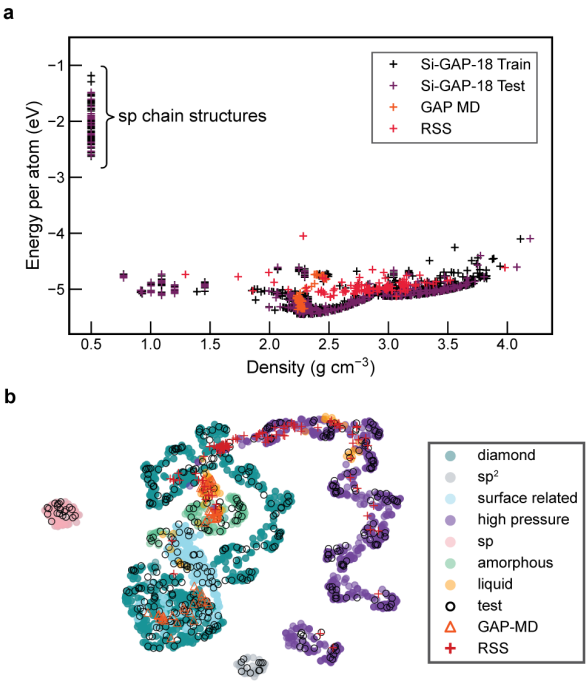

Figure 4.2: Characterisation of training and testing datasets. (**a**) Energy vs. density plot of structures in each dataset. (**b**) UMAP similarity map of ACE vectors generated by the Ref-ACE model from Ref. 133. Figure reproduced with permission from Ref. 2, Copyright 2024 American Chemical Society

In Fig. 4.2a, I characterize the various training and testing sets used by plotting the energy against the mass density for each structure contained in any of the four datasets. The bulk structures have densities which are primarily relevant to amorphous, crystalline, and liquid silicon, with sp-like chain structures constituting the only dataset entries at very low density and high energy. These structures are only included in the training and testing sets from Si-GAP-18, with a large discrepancy between these structures and the other classes of structure included in any of these

datasets. The RSS dataset is not only comprised of higher energy structures, but contains a relatively flat energy hull compared to the testing, training, and MQ-MD datasets which access lower-energy configurations between $2.0 - 2.5\,\mathrm{g\,cm^{-3}}$. Despite the difference in the composition of the RSS, MQ-MD, and Si-GAP-18 test datasets, their structures can be viewed as broadly related when compared in this plot, that is, their energies and densities are relatively cohesive. However, as I will show, this does not necessarily mean that an MLIP will perform consistently across all of these structure types.

In Fig. 4.2b, I show a similarity map obtained via UMAP dimensionality reduction[171] of ACE vectors of the atomic environments averaged for each structure. The ACE vectors were generated using the linear ACE potential fitted from Ref. 133, resulting in an 6827-dimensional feature space, which is mapped onto a 2D plot. Unlike in Figure 4.2a, dimensionality reduction of the ACE vectors separates the individual structure configuration classes of the training set, and I map the structures from the three validation sets onto this representation of feature space. The resultant map indicates distinct regions, mirroring the varied structure types included in the dataset. By colour-coding the points in the map according to the configuration types defined in the Si-GAP-18 training dataset,[55] I visualise the isolated nature of the low-coordinate ("sp" and "sp^2") and high-pressure structures (β-Sn-type and simple hexagonal), and the relation between the various phases included. The RSS dataset includes entries that (although clearly higher in energy; cf. Fig. 4.2a) resemble diamond-like, high-pressure, and liquid-like structures, suggesting a large range of atomic environments across the dataset. In studying the surface-related and diamond structures, I note a significant overlap. Upon further investigation, I surmise that this is because the surface-related structures (slab models) in the Si-GAP-18 training set are based on the diamond-type bulk structure, and so the similarity of these structures to bulk diamond-type Si is expected, and demonstrates that the vectors are describing well the local atomic environments seen in these structures.

Antimony Telluride

The dataset used to study Sb_2Te_3 is extracted from the GST-GAP-22 dataset.[43] In this case, I remove all structures which include Ge, yielding a dataset for the binary Sb–Te system. While the target of the model is solely the Sb_2Te_3 phase-change process between crystalline and amorphous phases, I include the structures across the phase space to provide context to the model about phase segregation. I split this dataset into train and test data (80:20) through random sampling of each configuration type as defined in the dataset. The result is a testing and training set with the same proportions of each configuration type as each other. Structure configuration labels examples include "`crystalline`", "`aimd`", and "`liquid`". By sampling randomly from each configuration subset, I ensure faithful reproduction of the existing dataset composition and that the training *and* testing databases include a variety of structures and improve the target for the loss function by avoiding oversampling of a specific subset of the database.

4.5 Optimising ACE models for disordered systems

The quality of models is determined by a number of factors including the choice of training data, hyperparameters, and model architecture. In the previous chapter, I used XPOT to fit efficient SNAP-based MLIPs across a number of elemental benchmark datasets. Here, I present how using XPOT to fit non-linear ACE models can generate models with accuracy and efficiency improvements over the state-of-the-art, and be used in the large-scale study of structurally disordered systems.

4.5.1 Si: optimisation & numerical validation

As discussed in the previous chapter, XPOT optimises hyperparameters by minimising the value of the loss function $\mathcal{L}$. The form of the loss surface is determined by the choice of validation set and the formulation of the loss function. In this case, the optimisation is guided by changes in the errors on the Si-GAP-18 test set.[55]

Table 4.1: Energy and force RMSE values of silicon potentials, evaluated on three different test sets. For the XPOT-ACE models, the number of atomic properties, P, as defined in Eq. 2.35, ranges from linear (1) to quaternary (4). An "F" denotes a potential where the number of functions was optimised by XPOT. XPOT-ACE-6827 is an optimised fitting of same number of radial basis functions as the linear ACE potential fitted by Lysogorskiy et al.[133] (denoted as "Ref-ACE" here).

			Energy RMSE (meV at.$^{-1}$)			Force RMSE (meV Å^{-1})			
	P	# Func.	Si-GAP-18[55]	GAP-MD[170]	RSS[59]	Si-GAP-18	GAP-MD	RSS	MD speed
Opt-1	1	3,000	3.5	5.1	27.1	69	105	158	47
Opt-2	2	3,000	2.5	5.0	23.1	63	97	150	46
Opt-3	3	3,000	318	5.2	$> 10^6$	300	98	$> 10^8$	45
Opt-4	4	3,000	4.8	5.5	62.6	63	99	274	43
Opt-3F	3	2,000	4.6	4.1	20.5	65	97	139	65
Opt-4F	4	1,625	2.5	5.4	72.5	64	100	187	80
Opt-Ref	1	6,827	3.0	4.4	34.5	63	104	179	16
Ref-ACE[133]	1	6,827	3.2	4.3	42.1	77	124	175	16
Si-GAP-18[55]	—	—	1.6	8.5	34.9	83	139	177	1

To determine the quality of the XPOT-optimised models, I use the Si-GAP-18 test set, MQ-MD, and RSS datasets as benchmarks, ensuring that there is a broad range of structure types included for analysis, and that the Si-GAP-18 dataset is a representative target for the optimiser. As the MQ-MD and RSS datasets were *not* used as targets in XPOT optimisation, I leverage them as distinct benchmarks for accuracy and robustness across models. The MQ-MD dataset from George *et al.*[170] is similar in character to the parts of the Si-GAP-18 test set, allowing further accuracy tests on larger structures which were not split from the training set. In contrast, the RSS dataset contains randomly generated structures which are typically higher in energy (Fig. 4.2a). No explicit RSS structures are present in either the Si-GAP-18 test or train sets. Using these two datasets, I can validate the performance of the Si-GAP-18 test set as a general-purpose optimisation target, that is, whether it well describes the larger structure-space accessible via MD of silicon.

I fit optimised ACE potentials to the general-purpose Si-GAP-18[55] training dataset (labelled as Opt-P, where P is the number of atomic properties). Previously, Lysogorskiy *et al.*[133] fitted a linear ACE model to the same data (that is, where $P = 1$) and non-linear ACE models have since been fitted to describe a variety of systems.[98,145,147]

For each value of P, I optimised up to 6 hyperparameters at once (Table 4.2), and performed 32 iterations (model fits), this is significantly fewer than the 150 iterations used for SNAP-based MLIPs on the Zuo *et al.* dataset, however, the larger training dataset size and increased fitting cost for these higher fidelity ACE models necessitates a smaller number of iterations, and I aimed to create a representative usecase for real-world application. Optimisation of the hyperparameters for these models took two weeks on a single NVIDIA A100 GPU and 16 cores of an AMD EPYC 7443 CPU.

The first four sets of model hyperparameters were selected via Hammersley sampling. These are referred to as "initialisation" iterations (shaded areas in Fig. 4.3).

After these four fits, the loss surface approximated via BO is applied for the remaining twenty-eight iterations. For all models, I optimised φ_i exponents, radial cutoff, the radial basis functions (including `radbaseparameter`), and `dcut` (the smoothing distance at the outer limit of the cutoff), with ranges as shown in Table 4.2. I used universal structure weighting unlike Ref-ACE, which includes weightings based on the type of structure in the training set.[133] I took this approach to prove that it was possible to forego manual "tuning" of weights across dataset entries when optimising hyperparameters, and the process of potential fitting can be further automated to allow focus on validation and production simulations. Future work should be undertaken to study the effect of weighting techniques to further improve fitting for smaller datasets, however, the focus here was demonstrating that improved models could be fit upon existing datasets without significant manual intervention.

After the optimisation run completed, I "upfitted" the best model from each optimisation process. While fine-tuning refers to the taking of a pre-trained model and further training it on a new, smaller and task-specific dataset, upfitting refers to the process of retaining the same dataset and only altering the force and energy weighting ratio. The accuracy of ACE models may be improved in this way: I first use a higher value of κ (see Eq. 2.36) to prioritise force prediction accuracy while fitting, before refining energy accuracy with low κ values in the upfitting process. This process is analogous to the `-swa` protocol used in MACE fitting.[172] For both

Table 4.2: Optimised hyperparameters and their ranges during the XPOT iteration process for Si.

Hyperparameter	Optimisation Range
`rcut`	5–8
`dcut`	0.001–0.1
`radbase`	'SBessel', 'ChebExpCos', 'ChebPow'
`radparameters`	1–10
`fs_parameters`	0.1–10
Where number of functions optimised ("-F" models):	
`functions_per_element`	500–2000

Si and Sb_2Te_3, I first fitted using $\kappa = 0.8$, before upfitting the same potential with $\kappa = 0.02$. After the final potentials were upfitted, I performed numerical validation on two external datasets (see Datasets section).

Comparing first fitting using the same B-basis functions and embedding as the Ref-ACE model, I fit the Opt-Ref model for Si. The optimised model achieves comparable energy errors to the Ref-ACE model from literature,[133] while improving in force prediction accuracy across all validation and testing databases (see Table 4.1). This model was the first ACE model which I had fitted, and at the cost of 32 iterations (that is, fitting 32 models) it was possible to improve the numerical accuracy compared to the state-of-the-art model. This demonstrates XPOT's ability to improve upon existing parameterisation of models, allowing researchers to confidently approach fitting ACE models on their datasets.

After fitting the Opt-Ref model, and determining that XPOT's optimisation was beneficial for fitting ACE models, I proceeded to fit a series of ACE models with XPOT over a range of P values. These models use fewer basis functions than the "Ref-ACE" model fitted by Lysogorskiy *et al.*[133] to improve inference speed. In Table 4.1, I compare the numerical errors of optimised models, as well as providing comparisons to the literature GAP and ACE models. The "best" model I fitted, Opt-2, achieves comparable accuracy to the Ref-ACE model, with a 3 times reduction in the inference time. This is achieved by taking advantage of optimised hyperparameters and the increased flexibility of non-linear ACE models, retaining accuracy despite reducing the number of basis functions, which can be treated as a convergence parameter.[173]

All of these potentials were optimised with a fixed number of functions (3,000) for P between 1 and 4. I observed an increased likelihood of overfitting for $P \geq 3$ as the number of parameters increases. Particularly alarming are the energy errors of the Opt-3 model. An energy prediction error of 318 meV at.$^{-1}$ on the dataset used to guide the loss optimisation is of particular concern. These elevated errors of

the Opt-3 model occur during the upfitting process. The original model (before the upfitting process) has an energy prediction RMSE on the Si-GAP-18 test dataset of 7.9 meV at.$^{-1}$. While this is definitely more reasonable, it is still 4 times the training error, suggesting the model is already overfitted. However, upon undergoing the upfitting process, the error increases significantly, and the RSS prediction errors also become unacceptable. Analysing the errors of the upfitted potential exposes that the increased error arises from two individual structures with energy prediction errors of over 1 eV at.$^{-1}$. These structures do not have any special characteristics by eye, nor are they separated by PCA or UMAP analysis of their ACE vectors (as in Fig. 4.2b), suggesting that the model is failing to generalise from the training data to the validation structures. The errors on the RSS dataset here highlight the need for testing of models on a diverse structure set, as the unsuitability of the Opt-3 model for MD would not have been exposed by testing on the MQ-MD database, and instead a more wide-ranging analysis including RSS structures reveals that the potential has significant stability issues.

After identifying these issues, I optimised `number_of_functions_per_element` for $P = 3$ and $P = 4$, to allow XPOT to converge upon the ideal number of functions for these more flexible models. XPOT does not consider inference cost in the default loss function (used here), and so I only optimise against overfitting, searching for the best possible potential within the hyperparameter space limitations. These models are labelled with the "F" suffix.

The Opt-3F and Opt-4F models improve upon both the accuracy and robustness of their 3,000 function counterparts. The Opt-3F model is more robust, and experiences less variation between errors on the Si-GAP-18 test dataset (which it was optimised towards) and the MQ-MD dataset. However, the Opt-4F model is less accurate on the RSS and MQ-MD datasets, despite improved accuracy on the test dataset. The Opt-4F model improved inference time by 1.9 times compared to the Opt-4 model, and retains the numerical accuracy of Opt-4. However, as the op-

timisation still occurs for the Si-GAP-18 test set, accurate predictions on RSS data is still not guaranteed, as shown by the high RSS energy errors for Opt-4F. The Opt-4F model numerical accuracy results exemplify the importance of the composition of the validation set (that is the dataset that is used to guide the loss surface and thus the hyperparameter choices made) in model optimisation, as discussed in Chapter 3.

In addition to boasting improved numerical accuracy and efficiency, the -F models were stable in melt–quench simulations for 4,096-atom cells (Table 4.5). This is particularly of note as it indicates that elevated RSS force errors and molecular dynamics instability are related, although one does not directly ensure the other.

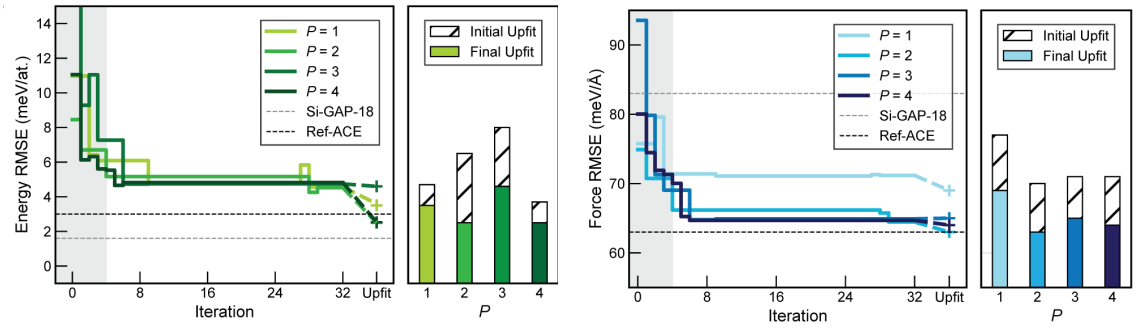

Figure 4.3: Evolution of energy and force errors for silicon ML potentials through iterative optimisation using XPOT. I show (**a**) the per-atom energy RMSE and (**b**) the force component RMSE across iterations. Errors are evaluated on the Si-GAP-18 test set from Ref. 55. On the left, the grey region indicates the initial sampling via Hammersley sequence. The bar charts show the errors on the test set for "upfitted" potentials, as described in the text: both for the best potentials from the initialisation protocol (hatched), and the best potentials after Bayesian optimisation (solid). The data for $P = 3$ and $P = 4$ refer to the '-3F' and '-4F' potentials from Table 4.1, respectively. Figure reproduced with permission from Ref. 2, Copyright 2024 American Chemical Society

The optimisation learning curves are visualised in Fig. 4.3, including the upfitting step. I do not show the Opt-3 and Opt-4 models, rather showing the improved Opt-3F and Opt-4F models. At the time that the fitting was carried out with `pacemaker`, early stopping of fitting based on plateauing of the loss function was not implemented and so manual determination of the number of iterations was necessary. Notable improvements in energy predictions are achieved through upfitting, with relatively

minor improvements to the force prediction RMSE achieved. The overall effect is to further converge the models and improve accuracy on the Si-GAP-18 test set.

Additionally, I upfitted the initial potentials (those produced within the first four iterations; Table 4.3). This did not produce a repeatable trend, with some models improving in accuracy, and others becoming less accurate. This suggests that the initial hyperparameters (those sampled by Hammersley sequencing) are not sufficiently providing a generalisable description of silicon, as is to be expected if hyperparameter choice is important to resultant behaviour. When testing these potentials in MD up to 1,800 K, stability over 100 ps is only possible with Initial-1. The rest of the "Initial" models are incapable of completing MD simulations at these elevated temperatures. The numerical accuracy of these potentials is explored in Table 4.3.

Taking into account all of these results, I determined that the Opt-2 model was the most accurate and robust - thus the most suitable for the task of a general-purpose MLIP for modelling silicon, henceforth referred to as "XPOT-ACE".

The optimised exponents for the summation of atomic properties for each non-linear model are presented below with all values rounded to 2 decimal places. Linear models are not included, as their functional form is as defined in Eq. 2.33.

4.5.2 Si: Physical Validation

As discussed previously, numerical validation provides a view into a model's understanding of the potential energy surface, but does not necessarily allow one to ascertain the usefulness of a model in simulating chemical processes. Alongside numerical validation, physical validation is required to benchmark a model's behaviour, and ensure that the results align with experimental and theoretical findings.

I plot the errors for each structural snapshot from the MQ-MD test set in Fig. 4.4. Compared to the results in Table 4.3, this experiment provides a more nuanced view of which specific configurations the models predictions are better or worse.[170] I show that XPOT-ACE improves the force predictions compared to Ref-ACE, whilst hav-

ing slightly higher errors (under 1 meV at.$^{-1}$). My optimised model, XPOT-ACE, is comparatively most improved for the higher-energy liquid structures, while improvements for crystalline and amorphous structures are reduced. This can be understood to be a consequence of the uniform weighting of the dataset that was used in my optimisation routine. In both Si-GAP-18 and Ref-ACE model fitting, specialised custom regularisations or weightings of energies and forces were used to maximise performance of the model on key configurations. In both cases, more weight was placed upon the accuracy of crystalline configurations. While the higher weighting of the crystalline phases for existing potentials[55,133] improved their accuracy for crystalline configurations, the XPOT-ACE model was still able to maintain improved force predictions upon these configurations.

Across the simulation snapshots, the same structures are resulting in "spiking" (high error) energies across all three models. This suggests some underlying characteristic of the Si-GAP-18 training dataset results in these less accurate predictions across models, especially on freezing of the liquid state (which is not well represented in the training data). All three models show increased errors in predicting the energy of the structures during the quench stage of the simulation. These fluctuations in errors are much reduced in force predictions, but there is still a visible "bump" in these predictions for the same structures.

After testing the investigating further the meaning of the numerical errors on the MQ-MD dataset, I tested the outputs of melt-quench simulations against existing models and DFT. I quenched five 500-atom randomized structures at fixed rates between 10^{14} to 10^{11} K s^{-1}, from 2000–500 K and relaxed them. I carried out quenches using both the XPOT-ACE and Ref-ACE models and compared the energies of the structures produced with several models. On top of labelling these structures with XPOT-ACE, Ref-ACE, and Si-GAP-18, I computed DFT energies for the structures, to quantify the predictive accuracy of all three potentials (akin to studies of quenched SiO_2 in Table 2 of Ref. 174). For each rate, five random struc-

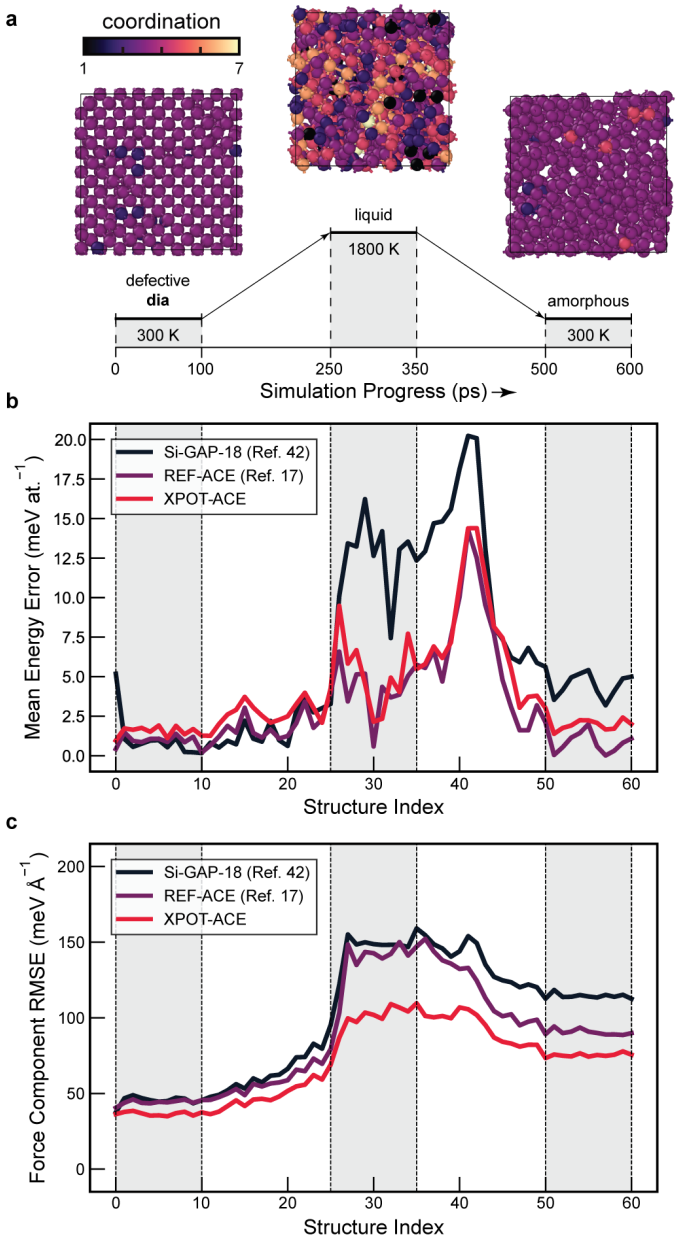

Figure 4.4: Accuracy of silicon ML potentials evaluated on DFT-labelled snapshots from a Si-GAP-18-driven melt–quench simulation reported in Ref. 170, and visualised in the style of that prior work. (**a**) An overview of the constant-pressure simulation protocol, adapted from Ref. 170. The images show the three classes of structure seen at each stage of the simulation, colour-coded according to coordination number. DFT snapshots were computed every 10 ps throughout the simulation. (**b**) Energy errors compared to DFT snapshots along the simulation trajectory. (**c**) Force errors for the same structures. The optimised model (XPOT-ACE) outperforms all other potentials studied here in terms of force errors, but is less accurate for energy errors cf. Ref-ACE from Ref. 133. The schematic in panel (a) and the overall style are adapted from Ref. 170, which is published under a CC BY licence `https://creativecommons.org/licenses/by/4.0/`. Figure reproduced with permission from Ref. 2, Copyright 2024 American Chemical Society.

tures were quenched, with both mean and standard deviation reported in Table 4.5. I also tested faster quench rates, but observed that rates of 10^{14} and 10^{15} K s^{-1} led to structures within 2 meV at.$^{-1}$ of each other for all models. Therefore, I do not include results for quench rates greater than 10^{14} K s^{-1} in Table 4.5, as they do not provide extra context to the relative behaviour of the models.

XPOT-ACE and Ref-ACE perform very similarly for 10^{12} and 10^{13} K s^{-1}. However, when quenching at 10^{11} K s^{-1} the XPOT-ACE-driven simulations resulted in lower energy structures, suggesting a more relaxed (and therefore more stable) a-Si structure. All models behave as expected, predicting energies broadly in line with each other and DFT. I note that in both cases for quench rates below 10^{13} K s^{-1} the model used to relax the structure predicts lower energies than either of the other MLIPs. This is as expected: the quenching and relaxation of the structure have been driven by the approximation of the potential energy surface provided by this particular model, and is thus biased towards lower energies on the PES. However, it is especially interesting because the DFT values are lower for all quenches of below 10^{12} K s^{-1}, suggesting that the models are overpredicting the energy offset from diamond-type c-Si.

Alongside this numerical validation against DFT, I performed a 4,096-atom MD quenching simulation with XPOT-ACE and Ref-ACE to compare to the results from Ref. 166 and experimental findings.[175,176] The quenching occurs via a constant-pressure simulation protocol with a variable quench rate before relaxing the final structure, in accordance with the procedure outlined in Ref. 166.

The structure factors of the final a-Si structures are shown in Fig. 4.5. Both Si-GAP-18 and XPOT-ACE almost perfectly reproduce the FSDP of Xie *et al.*,[176] while the Ref-ACE model does not achieve the same agreement with experimental results. All three models then reproduce the $S(Q)$ of Laaziri *et al.*,[175] although the Si-GAP-18 model seems to slightly overpredict the ordering (i.e., has very slightly sharper peaks and troughs) compared to the experimental data and ACE models.

Finally, satisfied with the numerical and physical predictions of the optimised model, I performed structural validation tests for the compression of silicon, as described in Ref. 40. This test was comprised of compressing a 100,000-atom low density amorphous (LDA) silicon model up to 20 GPa. The pressurisation rate was 0.1 GPa ps^{-1}, and the temperature is held at 500 K for the duration of the 200 ps simulation, in alignment with Ref. 40. I observed similar behaviour to predictions from simulations using Si-GAP-18,[40] whereby low-density amorphous (LDA) silicon collapses into a very-high-density amorphous (VHDA) phase under compression at $\approx$ 12 GPa, and subsequently simple-hexagonal (sh) crystallites nucleate and grow throughout the rest of the simulation. Agreement between the optimised model (XPOT-ACE) and Si-GAP-18, but also to experimental findings[177] demonstrates that my XPOT-optimised ACE model has retained not only the numerical accuracy, but the physical insight of Si-GAP-18 based on the training data available.

In Fig. 4.6 I visualise the results of the simulations. The formation of the VHDA phase occurs approximately 5 ps later in the XPOT-ACE-driven simulation when

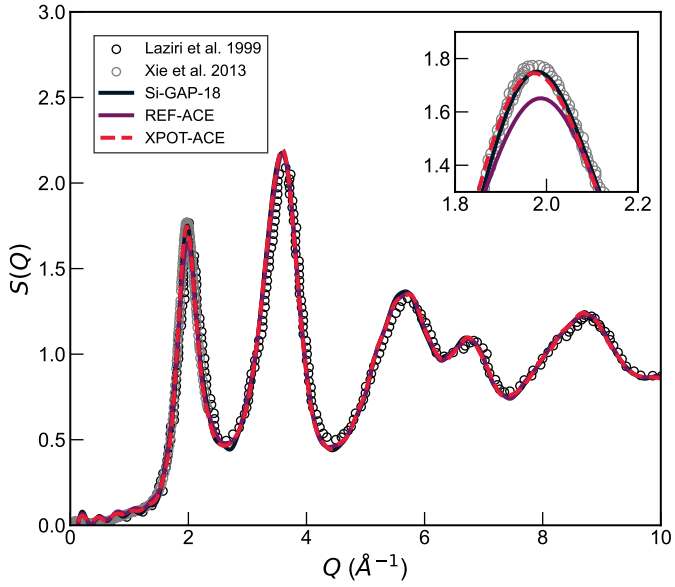

Figure 4.5: $S(Q)$ for a 4,096-atom structural model (in the case of coloured plots), compared to experimental results from Ref. 175 and Ref. 176. Inset: Comparison of the first sharp diffraction peak.

compared to Si-GAP-18, and is shorter lived (vis. Fig. 4.6f). I also observe this effect in Fig. 4.6b where there is a relative lack of highly-coordinated silicon atoms. Despite these subtle differences, both potentials predict very similar densities for each phase, and show behaviour that is consistent with experiments for both VHDA formation[179] and crystallisation under pressure.[177]

4.5.3 Antimony Telluride

To test my approach for a more complex material system, I fitted a potential for Sb_2Te_3, which is an important chalcogenide material used in various phase-change materials (PCM)-based devices for ultra-fast data storage[180] and high-performance neuromorphic computing tasks.[108,181] PCMs have long served as key application cases for ML potentials, including early work on the binary material GeTe[182] and the ternary $Ge_2Sb_2Te_5$.[183] The former potential has been used for studies of crys-

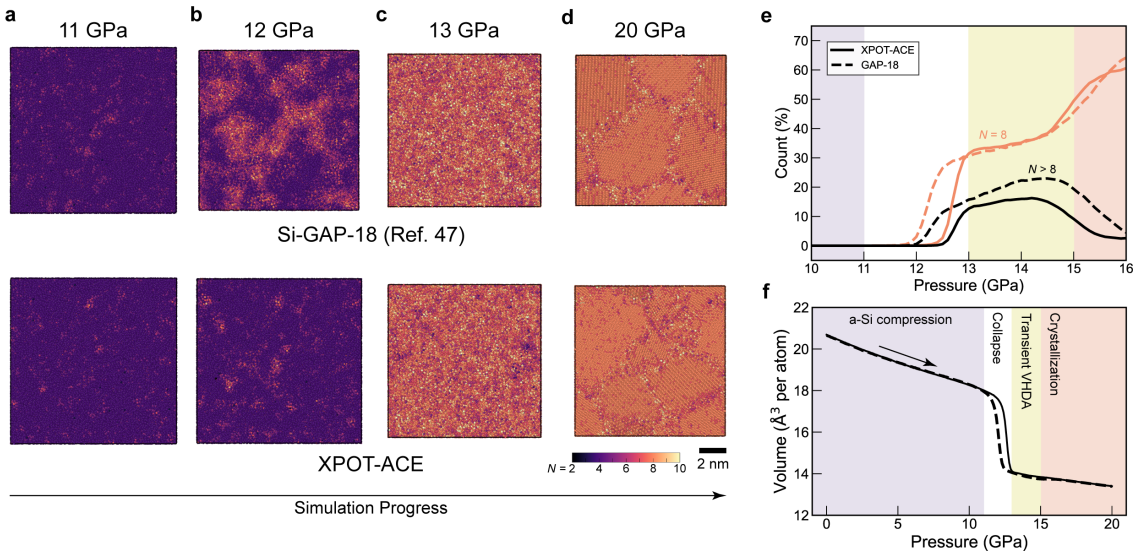

Figure 4.6: (**a–d**) Simulations of amorphous silicon under isothermal compression with both Opt-2 ("XPOT-ACE" for brevity) and Si-GAP-18,[55] under identical conditions to the simulation reported in Ref. 40 from which data for the GAP simulation are taken. Both potentials predict a collapse into VHDA occurring between 12–13 GPa from which simple hexagonal crystallites then form. (**e**) Coordination numbers as a function of pressure in the trajectories (determined by counting neighbours up to 2.85 Å). The initial increase in $N > 8$ atoms corresponds to the formation of the VHDA phase before crystallisation occurs. (**f**) Volume against pressure during the simulations. Simulations were carried out using LAMMPS.[178] The results are visualised in a similar style to Ref. 40 for consistency in comparison. Figure reproduced with permission from Ref. 2, Copyright 2024 American Chemical Society.

tallisation[184] and thermal properties of GeTe;[185] the latter has been applied to study structure and bonding in $Ge_2Sb_2Te_5$[186] and tested for the binary Sb_2Te_3.[187]

Our previous work has introduced an ML potential based on the GAP framework for Ge–Sb–Te (GST) alloys located along the compositional tie-line between GeTe and Sb_2Te_3.[43] This GAP model, which I call "GST-GAP-22", can accurately describe disordered structures of GST alloys and complex phase transition processes under practical programming conditions (e.g., non-isothermal heating) on the length scale of real-world devices.[43] I took a subset of the GST-GAP-22 dataset, which only contains elemental crystal structures of Sb and Te as well as binary bulk structures (including crystalline, amorphous, and intermediate crystallisation configurations) found in the Sb–Te system.

A validation set was created using the protocol described in the Methods section, providing a representative sample of structures, and allowing us to consistently quantify the performance and robustness of my potentials. This is important for defining the loss function in XPOT (cf. Eq. 3.4), and so the validation set is fixed for all potentials fitted during optimisation.

Additionally, to confirm that the reduction in scope of the training set (from the full Ge–Sb–Te system to only the Sb–Te system) was not unfairly advantaging my own optimised potentials, I fitted a GAP with the same hyperparameters as for the GST-GAP-22 potential,[43] but using only the Sb–Te subset of the data for training. I use this potential ("SbTe-GAP" in the following) as a benchmark as it provides very similar force errors to GST-GAP-22, but resulted in improved accuracy for energy prediction.

In this case, I cannot directly compare to GST-GAP-22 numerically, as the validation data are taken from the GST-GAP-22 training dataset. Therefore, to quantify the accuracy of the potentials, I created a new benchmark dataset similar to that built for silicon in Ref. 170 (cf. Fig. 4.4). Specifically, I performed a GAP-MD simulation in which Sb_2Te_3 was melted starting from a defective rocksalt-like crystal

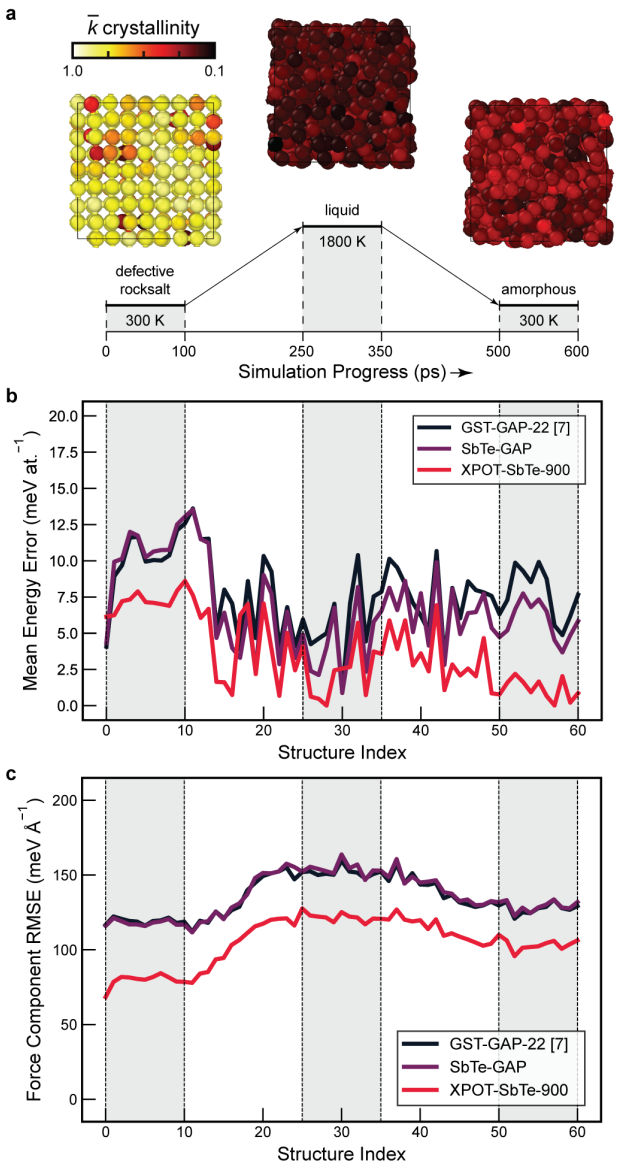

Figure 4.7: Prediction accuracy of Sb_2Te_3 ML potentials across DFT-labeled snapshots from a GST-GAP-22 melt-quench simulation. (**a**) An overview of the MD-based benchmark protocol created in a way similar to Ref. 170, now for Sb_2Te_3. Structures show the three classes of structures seen at each stage of the simulation, colour-coded according to a per-atom crystallinity measure.[60,188] Snapshots were labeled with DFT every 10 ps throughout the simulation. (**b**) Energy errors compared to DFT snapshots across the simulation trajectory. (**c**) Force errors across the same structures. XPOT-ACE outperforms both GAP potentials on both energy and force errors. Figure reproduced with permission from Ref. 2, Copyright 2024 American Chemical Society.

with an anion (Te) vacancy, before being quenched at 10^{13} K s^{-1} to form the amorphous phase. This trajectory was then labelled with DFT to produce a benchmark set representing crystalline, liquid, and amorphous Sb_2Te_3. Snapshots of these three

phases are shown in Fig. 4.7a. The atoms are color-coded by crystallinity as defined by the Smooth Overlap of Atomic Positions (SOAP)-based similarity,[60] with respect to rocksalt-like Sb_2Te_3, as discussed in Ref. 188.

In Fig. 4.7, I show that my XPOT-optimised ACE potential has improved not only upon the GST-GAP-22 predictions, but also the SbTe-GAP fitted on the Sb–Te subset of the full GST-GAP-22 database. The numerical accuracy is improved, and predictions are over 400 times faster than for the GAP potentials. Fig. 4.7b shows that the energy errors on the defective crystalline structures are consistently higher than for the liquid or amorphous phases across all potentials, with the same structural snapshots showing higher errors in this region. This occurs due to the relatively small number of crystalline structures with vacancy defects in the training database, and while my potentials all predict the energy to within 15 meV $at.^{-1}$, I see that these structures present more of a challenge to these ML potentials than the

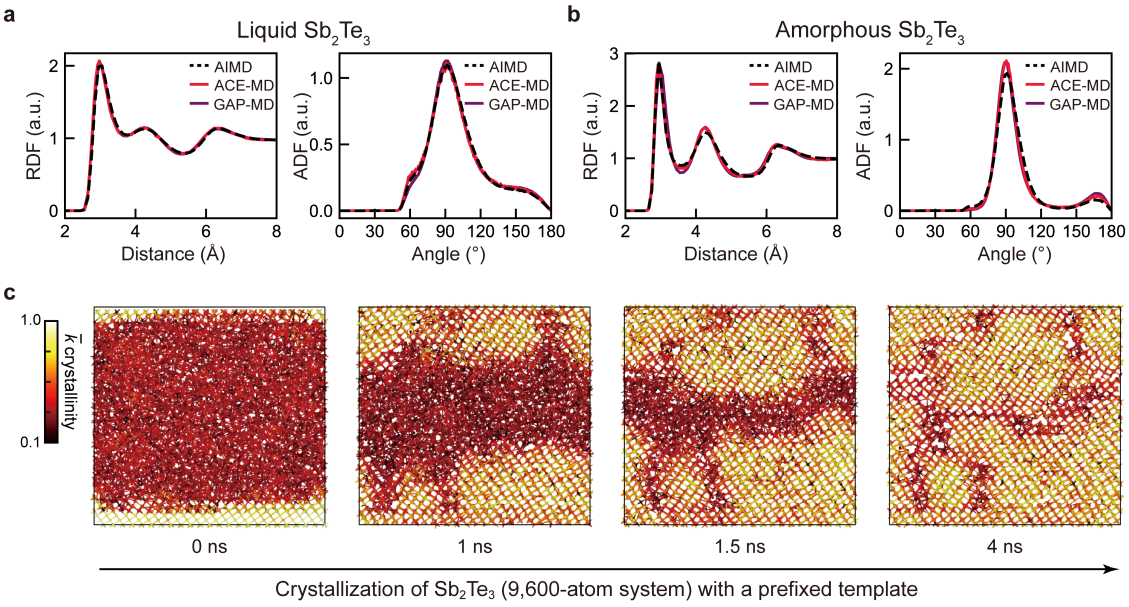

Figure 4.8: Structure and dynamics of Sb_2Te_3. (**a–b**) RDF and ADF plots for liquid and amorphous structures simulated by the SbTe-XPOT-ACE (red) and GST-GAP-22 (purple) models as well as AIMD (black). The data for the latter two are taken from Ref. 43. (**c**) Snapshots of an MD crystal-growth simulation. SOAP similarity is used to highlight the growth of crystalline Sb_2Te_3 (yellow) across snapshots. Several crystal grains and grain boundaries are visible. Simulations were carried out using LAMMPS.[178] Figure reproduced with permission from Ref. 2, Copyright 2024 American Chemical Society.

liquid or amorphous structures do. The SbTe-XPOT-ACE potential is significantly more accurate than both GAPs in predicting energies and forces for amorphous structures, and the force predictions are uniformly over 10% closer to DFT than those of by either GAP potential.

I carried out further physically-guided validation, testing the similarity of structural predictions between AIMD, GST-GAP-22, and my XPOT-ACE potential, aiming to verify that numerical errors correspond to well-described physical properties and processes in simulations. I first characterize the structure of amorphous Sb_2Te_3. I ran a melt-quench simulation on a crystalline Sb_2Te_3 model containing 360 atoms. In Fig. 4.8a–b, I show that my liquid and amorphous structures generated using SbTe-XPOT-ACE closely match the RDF and ADF of structures generated using AIMD and GST-GAP-22, including the shoulder in the ADF of the liquid phase. Both GST-GAP-22 and the XPOT-ACE potentials marginally over-order the amorphous phase of Sb_2Te_3, evidenced by a slightly larger peak at 90° in the ADF and a larger second peak of the RDF, as compared to the AIMD reference (Fig. 4.8b).

To move beyond structural validation, I produced a 9,600-atom structural model of amorphous Sb_2Te_3 (in a box of $4.3 \times 9.0 \times 8.5$ nm^3) with a pre-fixed crystalline template to simulate the crystal-growth process. Fig. 4.8c shows a crystallisation simulation for this templated structural model. As in Fig. 4.7, I used SOAP-based similarity[60] with respect to rocksalt-like Sb_2Te_3 to quantify the per-atom crystallinity during the crystallisation process.[188] Our structural model was annealed at 600 K for 4 ns, and the growth proceeded quickly at the rough crystalline–amorphous interface. Upon nanosecond crystallisation, I found many defects (e.g., point defects and layer stacking faults) in the recrystallised model (cf. the dark red atoms in Figure 4.8c), indicating competing growth of different crystalline regions with different crystal orientations. I note that such local disorder is challenging to fully characterize due to the short timescales on which they happen (e.g., in the programming operations of real-world devices), and high-temperature annealing

can help to eliminate the local defects, e.g., via vacancy ordering,[189] resulting in an energetically more favourable crystalline phase with fewer defects.[190]

4.6 Ge–Sb–Te

The critical problem in the development of accurate MLIPs for PCMs is reducing the inference cost to allow for nanosecond device-scale MD simulations. Existing models such as GST-GAP-22[43] are too expensive to run at device-scale over tens of nanoseconds (corresponding to realistic write operations in real-world devices), requiring over 3 years on a single node (128 cores) to run a 1 ns device-scale (over 500,000 atoms) simulation. With the rise of PCM-based devices, the need to model not only the amorphisation, but also the crystallisation process, is becoming increasingly important. However, amorphisation takes less than 100 ps, but crystallisation can take upwards of 10 ns depending on conditions.[43]

Following the success of the XPOT-ACE model for Sb_2Te_3, I applied the same approach to the GST-GAP-22 model, which was trained on the full Ge–Sb–Te system. The dataset used was the GST-GAP-22 dataset, re-labelled with PBE, augmented with further AIMD configurations of amorphous GST (Ref. 43). This dataset is named "Iter-0" henceforth. Based on my success in creating general purpose optimised ACE models for Si and Sb_2Te_3, I identified an inference cost target, and used XPOT to optimise a model with 3,000 basis functions per element in the first case. In order to define a representative target for optimisation, we curated a validation set comprised of AIMD-generated structures of conventional disordered structures and configurations during phase-transition processes, sourced from Ref. 43 and Ref. 188 respectively. The dimer structures were removed from the dataset, as doing so had provided improved accuracy in the above studies. This validation set remains fixed for all models fitted. The composition of the validation dataset was chosen as the most important behaviour to capture is the crystallisation process, and the amorphous phase also contains a large variety of local atomic environments.

4.6.1 Model Development

Hyperparameter optimisation was performed using XPOT, with 32 iterations. 8 sets of hyperparameters were chosen via Hammersley sampling, and the remaining 24 iterations were chosen via BO. 4 hyperparameters were optimised, and the loss against the validation set with $\alpha = 0.4$ was used. I do not optimise the radial basis function, as both the Si and Sb_2Te_3 models found spherical Bessel functions to be the most effective. The cutoff was limited to 8 Å for performance reasons, as higher cutoffs would continue to increase the cost of the model in inference. The final values of the optimised hyperparameters are shown in Table 4.6. Once the optimised hyperparameters were determined, the Iter-0 model was created by upfitting the model with the optimised hyperparameters using $\kappa = 0.8$, followed by $\kappa = 0.01$ to converge energy errors.

The initial Iter-0 model achieved reasonable accuracy on the validation dataset (15 meV at.$^{-1}$ / 125 meV Å^{-1}), but in device-scale MD simulations of the RESET process (whereby the structure is heated to melt the crystalline phase, and then quenched to form the amorphous phase), the optimised Iter-0 model was not robust enough to complete the RESET process consistently. At high temperatures, occasionally atoms would reach areas of structural space dissimilar to those included in the training dataset, and unrealistically large changes in ε_{ML} would occur, causing unphysically large forces which result eventually in lost atoms from the simulation box. Lost atoms are caused by instability in the potential resulting in unphysically high velocities of atoms which cannot be resolved between steps in LAMMPS. In the case of my simulations, this occurs due to unreasonably high gradients in the precited energy surface providing strong forces which produce high velocities which propogate amongst atoms in the system.

In order to solve this, we generated an iterative process for adding both high-energy structures to the dataset in order to improve the robustness of the model, and domain-specific structures to improve the accuracy of the model in phase-transition

processes. To generate domain specific structures, the ACE model is used to drive MD simulations annealing crystalline structures and simulating crystal growth. The high-energy structures are very small hard-sphere random structures (6–40 atoms) of varying chemical composition. The structures have reduced sizes to ensure efficient DFT labelling, reducing the cost of the iterative process for non-domain-specific critical structures. These structures were aimed at reducing some unphysical "clustering" behaviour of the model which occurred in simulations over 1 ns when the temperature was above 500 K. After the optimised ACE model became robust enough to drive random structure search (after Iter 3), two more iterations were completed adding more ACE-relaxed random structures to the dataset.

We then defined a "survival" metric as the percentage of high-T MD runs that were completed for ablation models, as well as using the numerical errors on the validation set to assess the loss of accuracy in key areas (Fig. 4.12a). Ablation refers here to the process of altering (via removing or reducing) the training data or "convergence" hyperparameters to produce less complex models, allowing one to evaluate the importance of various aspects of the training procedure. Thus, an ablation model is a model which has been simplified from the final model in one way (e.g., reducing the number of basis functions). Once the model's accuracy and robustness were sufficient, the iterative process was stopped.

Overall, 1,944 new training structures were added to the dataset, corresponding to a 70% increase in the number of atomic environments in the training dataset. This iterative process is outlined in Fig. 4.9.

At the end of Iter-3, I performed another XPOT optimisation, in an attempt to determine whether further optimisation of the model on this new training dataset would provide a significantly different set of hyperparameters and an improved model. However, I found that in fact there was no improvement to the model, both the models optimised against Iter-0 and Iter-3 achieved accuracy within 5 meV Å^{-1} of each other in force prediction accuracy, and there was no notable difference in their

performance in MD. As such, the original optimisation values are used to continue the iterative process, and are used in the final model.

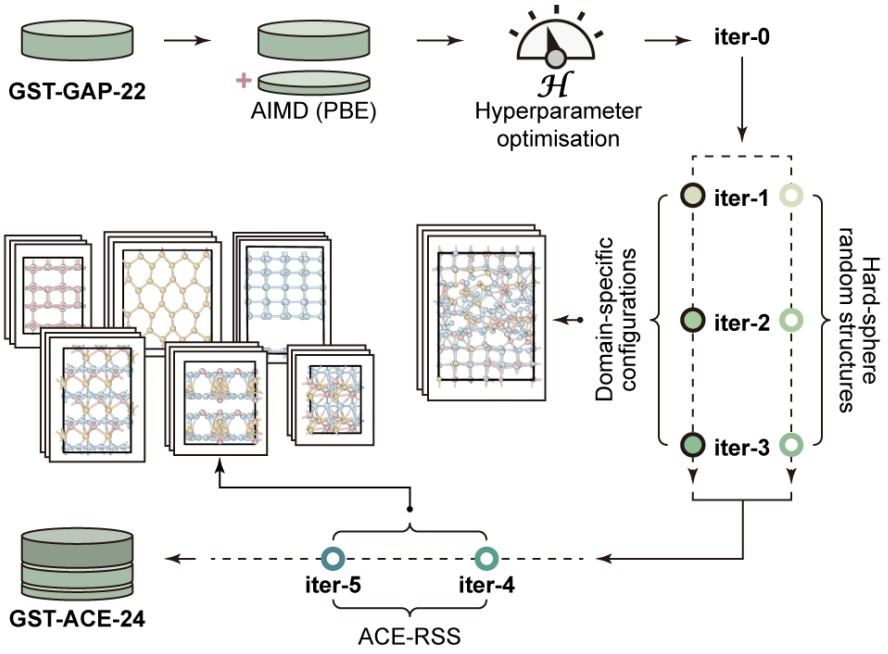

Figure 4.9: Overview of dataset iteration for GST-ACE-24. Optimisation occurs on the initial Iter-0 dataset with XPOT, and the dataset is extended throughout. The second XPOT optimisation is excluded as it was not used to drive any part of the iterative process. Adapted with permission from Ref. 4. Figure adapted with permission from Ref. 4.

The final model is named "GST-ACE-24" henceforth. We validate the final model against the local structure of GST as predicted by AIMD and GST-GAP-22. Preliminary analysis includes RDF and ADF comparison across the compositional tie-line of GeTe–Sb_2Te_3 in the GST system, where GST-ACE-24 and GST-GAP-22 both recover the same predictions as structures generated through AIMD. The relative frequency of homopolar bonds and Ge environments which are tetrahedral are key structural oddities within the Ge–Sb–Te class of materials. We test our models against the AIMD structures for the same tie-line, and find that my GST-ACE-24 model exhibits a significantly better match to the reference data than the GST-GAP-22 model.

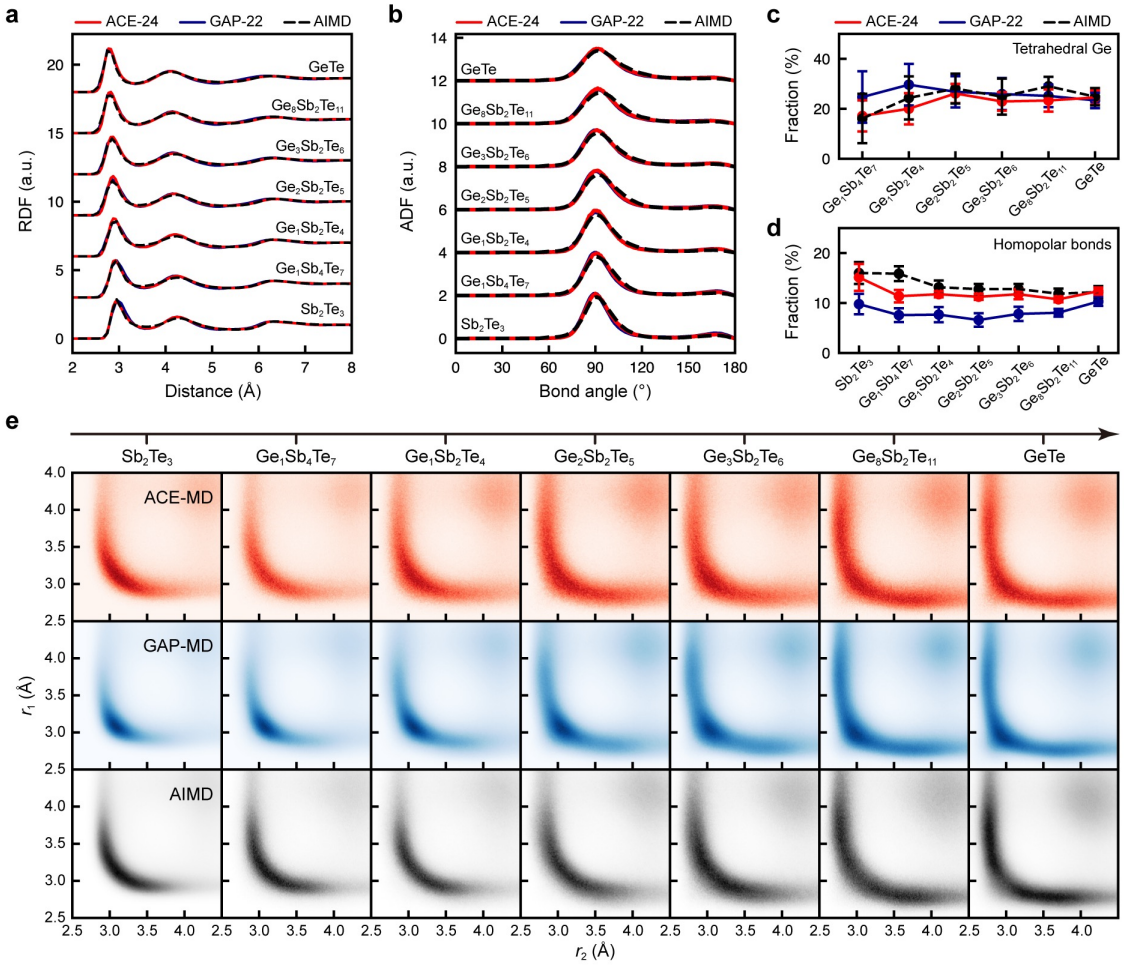

Figure 4.10: Local structure of amorphous GST compositions from ML-driven MD and AIMD simulations. (**a**) Radial distribution functions (RDFs) for seven different compositions along the GeTe–Sb_2Te_3 tie-line. (**b**) Angular distribution function (ADF) for the seven compounds. (**c**) The calculated fraction of tetrahedral Ge atoms, defined by a bond-order parameter, as discussed in previous work.[191] (**d**) Fraction of homopolar and "wrong" bonds, i.e., bonds between two cation-like atoms or two anion-like atoms (viz. Ge–Ge, Ge–Sb, Sb–Sb, and Te–Te). (**e**) Angular-limited three-body correlation (ALTBC) functions for the amorphous structures of seven GST alloys. The ALTBC function expresses the probability of having a bond of length r_1 almost aligned with another bond of length r_2 with angular deviations $< 30°$, providing a measure for the degrees of local distortion in the amorphous structures.[192] Results for AIMD, GAP-MD, and ACE-MD structures are shown in black, blue, and red, respectively. All data shown in this figure present mean values over 6,000 snapshots of AIMD, GAP-MD, and ACE-MD simulations (which were collected from the last 40 ps trajectories of 3 independent melt-quench runs). Error bars in (c–d) indicate standard deviations. All structural analyses were performed using the same type of benchmarks as in Ref. 43: all AIMD and GAP-MD results were produced based on the atomic trajectories as reported in Ref. 43. Figure reproduced with permission from Ref. 4.

Extending the analysis to the presence of Peierls distortions, Fig. 4.10 shows that GST-ACE-24 also improves on GST-GAP-22's match to reference data AIMD simulations for the angular-limited three-body correlation functions. Across all comparisons we completed, the optimised GST-ACE-24 model proved capable of recovering the same interactions and physical motivation as the reference data, and at this stage, the model was considered to be a good candidate for device-scale MD simulations.

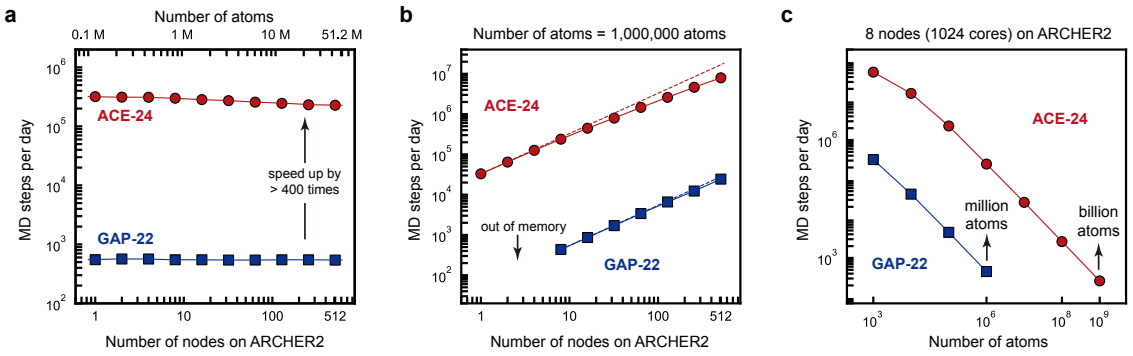

Figure 4.11: Scaling results for the GST-ACE-24 model compared to GST-GAP-22. (**a**) weak scaling at 100,000 atoms per node. (**b**) strong scaling up to 512 nodes. (**c**) MD steps (of 2 ps) per day for 8 nodes, a representative number of nodes based on the scaling behaviour observed in the strong scaling test. Figure adapted with permission from Ref. 4.

Once this validation was completed, we performed scaling tests compared to the previous GST-GAP-22 model. XPOT optimisation to leverage the efficient implementation of non-linear ACE models enabled a 400 times speed-up at inference time. This change takes us from over 1000 node days to run a 1 ns device-scale simulation to under 3 node days. Thus, it is now possible to directly assess the crystallisation process at the device scale. Indeed, it is possible to investigate the crystallisation process across a range of conditions.

4.6.2 Ablation Testing

To validate that the chosen level of complexity was appropriate for the model, we designed an ablation study on the final model. Ablation studies are designed to

quantify the importance of various individual aspects of a model and training dataset on the final predictive behaviour of a model.

Three parameters were to be varied: 1) the number of basis functions per element, 2) the number of embeddings (i.e., the non-linearity of the model), and 3) the number of RSS structures included in the dataset. I fit all ablation models using the same protocol: take the optimised hyperparameters of the final model, alter *only* the hyperparameter being ablated, and then perform the full upfitting process as completed for the final model, and defined above. The results of the survival tests are shown in Fig. 4.12b, and the numerical accuracy of the ablation models are shown in Table 4.7. By performing these ablation tests, we aim to quantify the importance of each aspect of the model and the training dataset on the final behaviour of the model. In this way we ensure that we test the key elements of behaviour for the model without introducing significant costs for ablation studies.

Beginning with survival rates, I observe that the iterative process has had a significant impact on the robustness of the model. The ACE-RSS structures have significantly improved the ability for the model to make meaningful predictions at high temperatures. The question then becomes what happens when we start to remove these structures? Fig. 4.12b shows the ablation of the RSS structures from the dataset. We observe that the model is able to maintain a high numerical accuracy, but the survival rate (a measure of rubustness) of the model is significantly reduced. This is a clear indication that the ACE-RSS structures are essential for the model to make accurate predictions at high temperatures. This is to be expected: the model is flexible, with a large number of parameters, and to remove non-equilibrium structures is to remove a corpus of information that is critical to the high temperature behaviour of the model. Other ablation models are not included in the survival analysis as they all achieve a 100% survival rate. This, again, aligns with my expectations: the flexibility of the model is being reduced (i.e., a reduction in parameters) via either the number of embeddings or the number of basis functions.

Turning to the numerical accuracy of the models with respect to ablation, I observe that the ablation of the random structures improves the energy and force errors of the model on the validations set. I explain this behaviour by considering the make-up of the training and validation datasets. Removing random structures from the training set in turn increases the ability of the model to fit to the lower energy, physically relevant structures which are akin to those in the validation set. Thus, observing a reduction in validation set errors can be understood against this change in the constitution of the training set.

When considering the complexity of the model, accuracy reduced as the flexibility of the model was reduced. Notably, the energy errors quickly jump upon reduction of basis functions, and the force error continues to increase as the number of basis functions is further reduced. I observe that the linear expensive model ($P = 1$) achieves comparable accuracy to the non-linear model with 300 basis functions, which offers significant computational efficiency improvements. This once again demonstrates the benefits of introducing non-linearity into the model for efficient device-scale MD simulations.

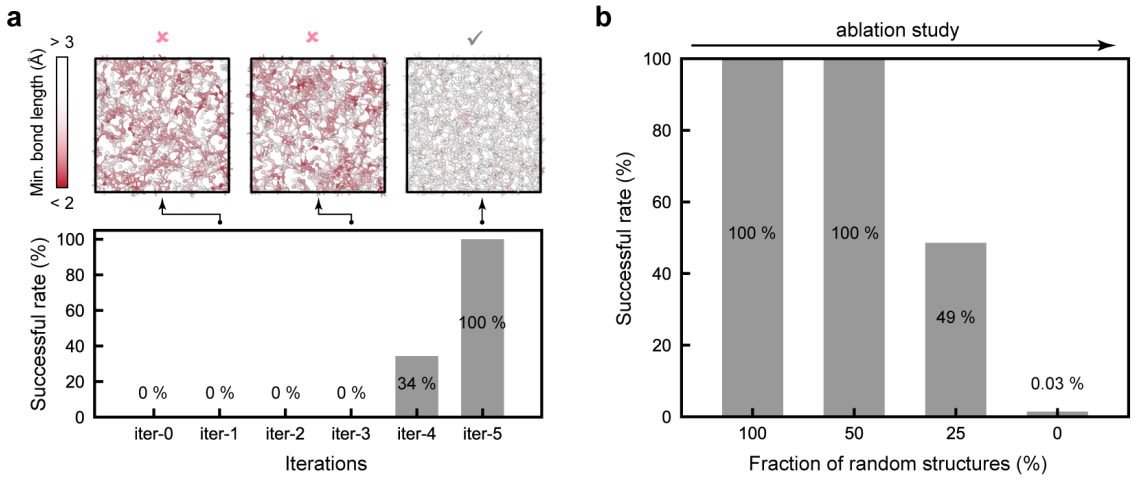

Figure 4.12: "Survival" metrics for ACE models. (**a**) The improvement of the model's robustness with the extension of the dataset through the iterative process. (**b**) Ablation study of the GST-ACE-24 model with respect to the number of random structures. Figure adapted with permission from Ref. 4.

4.6.3 Device-scale MD

The final model was used to study the SET process of cross-point PCM devices for phase-changing non-volatile random access memory (RAM) components. As the model I fitted provides a 400 times improvement in inference efficiency, a 20 ns simulation is now possible where only 50 ps could be performed before, due to the computational cost of the GAP model. In turn, this allows for the device-scale study of the SET process, whereby amorphous $GeSb_2Te_4$ is exposed to a longer, lower-voltage pulse to crystallise.

Non-volatile memory (NVM) devices encode bits, and each $GeSb_2Te_4$ PCM cell encodes either "0" or "1" dependent on its phase. The RESET process takes a 1 (crystalline) bit and reverts it to 0 (amorphous). The SET process takes a 0 (amorphous) bit and crystallises it to 1. However, the SET process takes a significantly longer time to complete than the RESET process, which can be achieved by melting the crystalline $GeSb_2Te_4$.

In order to complete a full-cycle simulation, the model is used to drive the RESET process from crystalline $GeSb_2Te_4$, before using the quenched amorphous model to drive the SET process. To do so, we simulate a $20 \times 20 \times 40$ nm^3 cell to represent the device. The resulting structure is comprised of over $500,000$ atoms.

The structure is heated for 10 ps by adding kinetic energy to the particles existing within the lower section of the structure, simulating contact with the buffer layer which is heated by the electrode. Periodic boundary conditions are used in the simulation, but a 10 Å frozen crystalline region in the z-axis is used to prevent the transfer of heat through the z-axis (where the buffer layer would be located). After heating, the structure is cooled for 40 ps. This structure is amorphous $GeSb_2Te_4$, and encodes a 0 bit. Then, the SET process is simulated by heating the structure to 600 K and annealing it for 20 ns. During this time, $GeSb_2Te_4$ nucleates and crystallises. This was the first ever full-cycle device-scale ML simulation of a non-volatile phase-change memory cell.

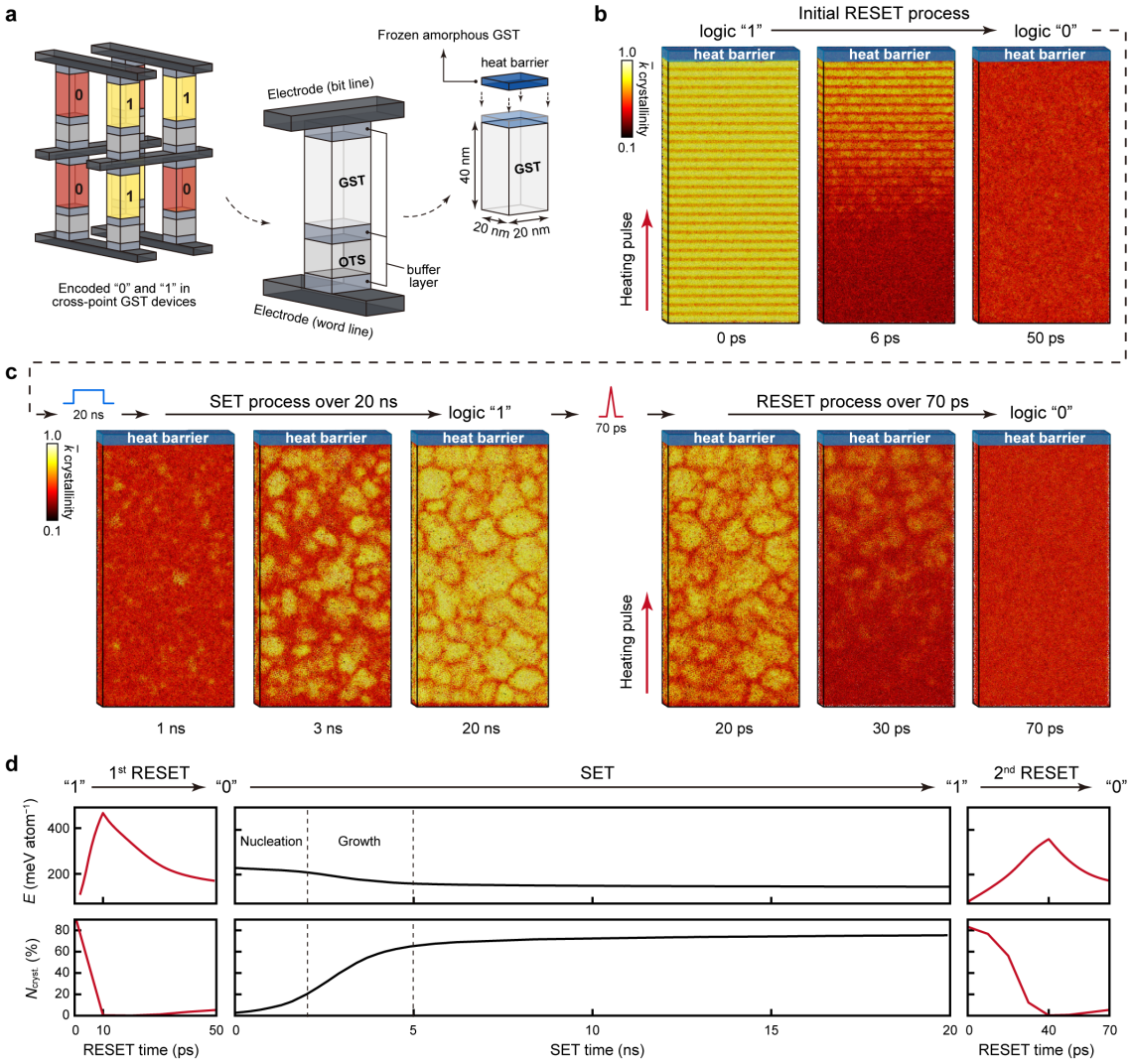

Figure 4.13: Full-cycle device-scale simulations. (**a**) Schematic of cross-point devices, in which logic "0" and "1" bits are encoded by amorphous and crystalline states of GST. In these devices, PCM layers and ovonic threshold switching (OTS) selector layers are sandwiched by buffer layers. A device-size structural model was built as in Ref. 43 ($20 \times 20 \times 40$ nm^3; 532,980 atoms); note that we here use amorphous GST as a heat barrier. (**b**) The initial RESET operation, simulated similar to Ref. 43 but now using the GST-ACE-24 model, starting from layered trigonal $GeSb_2Te_4$, triggered by a 10-ps heating pulse (0.064 pJ) from the bottom to the top. After the programming pulse, a 40 ps cooling was performed by removing the added kinetic energy from the structural model until it reached 300 K. (**c**) The subsequent SET operation at 600 K, simulated over 20 ns in the NVT ensemble. The resultant recrystallised GST structure contains 277 crystalline grains with an average diameter of ≈ 4.6 nm. A second RESET operation was then simulated, via a 40 ps heating pulse (0.036 pJ) and a 30 ps cooling process. Colour coding in panels (**b–c**) indicates the SOAP-based[60] crystallinity measure, $\overline{k}$ (see Ref. 188). (**d**) Computed potential energy and fraction of crystal-like atoms during the full-cycle simulations. A $\overline{k}$ cut-off of 0.57 was used to separate crystal-like and amorphous-like atoms.[188] Dashed lines indicate the nucleation and growth processes during crystallisation. Figure reproduced with permission from Ref. 4.

4.7 Tellurium

Having created an efficient and accurate optimised ACE model for the Ge–Sb–Te system, we turned our attention to elemental tellurium. Te is a promising material for selector devices, a critical component in high-density, non-volatile phase-change memory (PCM). Elemental ovonic threshold switching (OTS) devices are desirable as they promise greater stability, avoiding issues with phase separation seen in ZnTe devices.[193] These switches are defined by their switching behaviour, whereby upon removal of the voltage, the switch reverts to the off state. These selectors are essential for suppressing leakage currents in three-dimensional cross-point memory arrays, and are a candidate for OTS layers in PCM devices (Fig. 4.13). The switching mechanism in Te-based selectors relies on a rapid, reversible phase transition between a semiconducting crystalline "OFF" state and a metallic liquid "ON" state. This transition is triggered by Joule heating from an applied voltage pulse. After the pulse, the liquid Te spontaneously and rapidly recrystallises, returning the selector to the "OFF" state.

However, the extremely fast nature of Te crystallisation, occurring on nanosecond timescales, poses significant challenges for experimental observation at the atomic level. This has made it difficult to fully understand the formation mechanisms of its chiral crystal structures and the origin of performance variability, such as noise, in Te-based devices. To overcome these experimental limitations, large-scale, quantum-mechanically accurate atomistic simulations are crucial. Such simulations can reveal the microscopic pathways of crystallisation and phase transition, providing the fundamental understanding needed to explain and improve the electrical switching behaviour of Te-based selectors. By modelling the entire switching cycle, from melting a single crystal to the recrystallisation into a polycrystalline state, simulations can clarify key device characteristics, such as the difference between the initial and subsequent switching voltages and the impact of the material's microstructure on performance.

Model Development

At the time, Y. Zhou was developing a GAP model for Te, and we considered whether a similar uplift as was achieved for GST could be achieved for Te, enabling us to simulate Te-based OTS devices. The reference database was generated by augmenting a database of relevant configurations with GAP-RSS, followed by iterative GAP training. Full details of the dataset generation performed are published in Ref. 5.

We hypothesised that the hyperparameter values optimised for the Ge–Sb–Te model should provide a reasonable fit for Te, based on the fact that not only does the GST-ACE-24 database include some Te structures, but that the Ge–Sb–Te system must describe the behaviour of Te. Thus, I fit a model for Te using the same hyperparameter values as optimised for the GST-ACE-24 model. Comparing the fitted Te-ACE-24 model to the GAP model on a physically motivated test set comprised of structures derived from MD, I observe not only that the ACE model is more accurate across every structure category (Table 4.8), but it also has a significantly lower computational cost (by approximately 100 times).

Alongside this, the potential was stable up in small-scale (less than 1000 atom) melt-quench simulations up to 1500 K. As a final validation step, we compared the local structure predictions of both models to AIMD across a range of temperatures for local structure predictions, in a similar manner to the analysis performed for GST. We find that the ACE model has no notable differences from the GAP model in the local structure prediction, with RDF and ADF analysis both matching well the AIMD structure predictions.

Based on the findings of numerical accuracy and robustness in high temperature simulation, the Te-ACE-24 model was used to study the crystallisation process in OTS devices.

4.7.1 Ovonic threshold switch behaviour

The Te-ACE-24 potential was used to study the crystallisation process, chirality transfer, and the role of transient cubic motifs in these processes in our work in Ref. 5. Herein I focus on the investigation of Te OTS devices as a showcase for the transferability of XPOT-optimised potentials and the need for efficient MLIPs for large-scale simulation.

OTS devices enable the kind of cross-point devices studied in Fig. 4.13, and Te-based devices have been shown to exhibit large changes between the switch-on voltage for the first cycle vs. subsequent cycles.[194] Dependent on the annealing temperature used in the creation of the OTS, the first switch-on voltage, V_{fire}, increases.[194] In Fig. 4.14, a representative sketch (based on Ref. 195) of the different voltages are shown, where V_{th} is the threshold voltage for subsequent cycles. Full-scale device simulations of 102,000 atoms are performed to investigate the origin of these differences, and the dependence of V_{fire} on the annealing temperature.

To study the difference between V_{fire} and V_{th}, a perfect α-Te supercell is heated for 5 ns at a rate of 10^{11} K s^{-1} from 300–800 K. Then, cooling is performed at the same rate down to 300 K. 12 ns of annealing at 300 K are performed, before the heating cycle begins.

In Fig. 4.14b and e, the discrepancy between V_{fire} and V_{th} is shown. Note especially that the structural model in Fig. 4.14e contains grain boundaries comprised of amorphous tellurium, which become seeds for the melting process to begin at 523 K, as opposed to the perfect crystal, which only starts to melt at 723 K. As such, it is possible to understand that the difference between V_{fire} and V_{th} is most likely a consequence of the lack of higher temperature annealing between cycles, which would allow for the growth of larger crystalline grains and fewer grain boundaries from which the melting process could start. Further testing showed that for a polycrystal with only two grains, the temperature required for melting was 653 K, corresponding to a larger V_{th}.[5] The need for understanding the origin of V_{fire} and

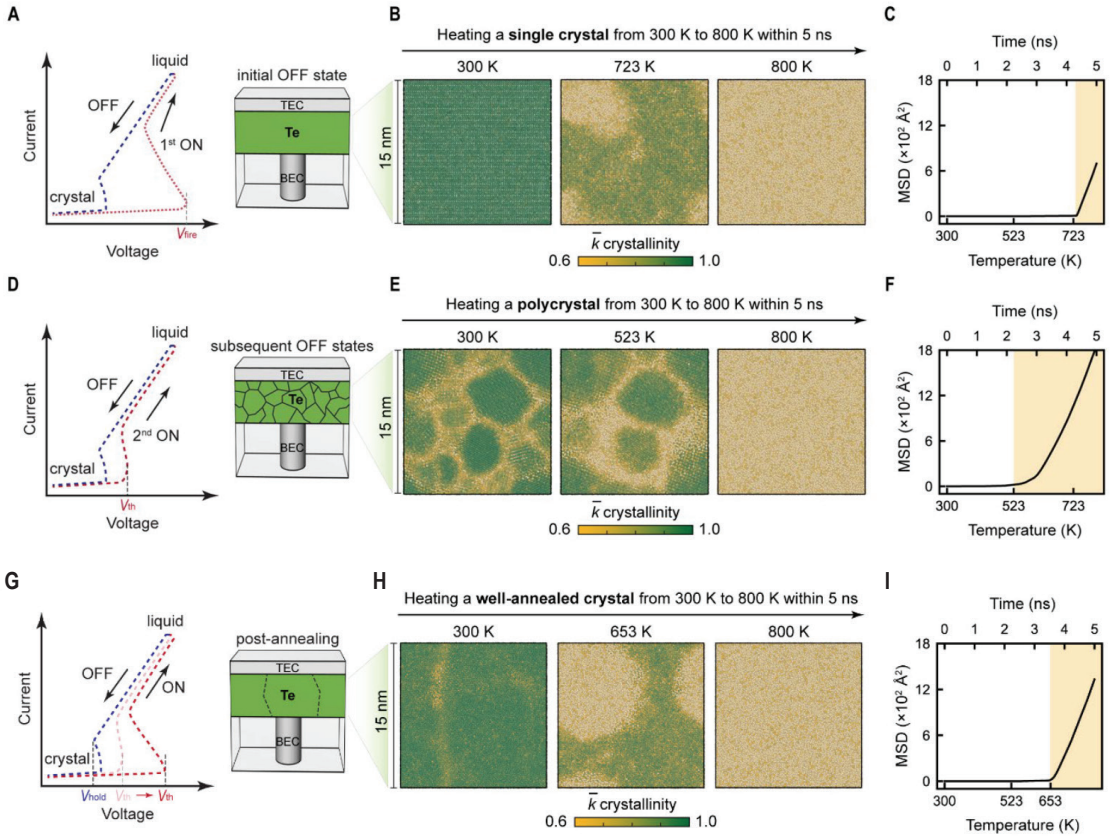

Figure 4.14: OTS device simulations. (**A, D, G**) Representative sketches of the different voltages in a Te OTS device. The difference between V_{fire} and V_{th} is shown dependent on the crystalline structure. (**B, E, H**) Representative snapshots of the melting process for various crystalline structures with varying levels of disorder. The middle snapshot is taken at the temperature at which the MSD begins to rise significantly. (**C, F, I**) MSD against temperature (time) for the melting simulations of all three structures. The onset of the melting process is shown to be related to the level of disorder in the crystalline structure. Figure adapted from Ref. 5.

V_{th} is due to the nature of the devices in which the OTS layers are used. Ideally, a threshold voltage of approximately 3 V is required to guarantee the programming window for the memory units such as the GST-based non-volatile RAM.[196] To increase the suitability for a Te OTS, annealing pulses may be needed to reduce the number of grain boundaries during crystallisation, in turn raising V_{th}.

4.7.2 Optimisation studies

Alongside the simulations performed, I investigated the difference between the GST-determined optimised hyperparameters, and new optimisations carried out for Te.

I did not optimise towards the test set, but rather towards a 10% hold-out from the training set. I chose this in order to provide a fairer comparison between the hyperparameters determined for GST and Te, ensuring that the optimisation for Te was not biased towards only the testing configurations.

Based on the GST hyperparameters as a starting point, I optimised the atomic property exponents, cutoff radius, and radial parameter for ACE models of 3000 basis functions, Opt-3k, and 750 basis functions, Opt-750. In both optimisations, the cutoff converged to the same 8 Å maximum that was optimal for GST-ACE-24, which confirms that this class of materials require much longer cutoffs than Si, where the optimal cutoff was less than 5 Å. In Table 4.9 I show that the Opt-3k model performs vastly similarly to the Te-ACE-24 model overall, with fluctuations in the errors between structure types. Opt-3k performs better on more ordered structures, where Te-ACE-24 is better on disordered structures. The consistency of the Te-ACE-24 model across structure types makes it a good candidate for use as a general-purpose Te model to study a wide range of crystallisation and melting processes. The Opt-750 model, as expected, performs worse than the existing models, with errors approaching 20% larger than for the more expensive models.

Overall, these findings suggest that the GST validation dataset was a good proxy for Te chemistry, and that in turn the optimisation of ACE models for GST yielded transferable hyperparameters between these materials. I note that the optimised basis functions, cutoff, and exponents between the Si Opt-3F model and the Te/GST models are very different and thus optimising hyperparameters for one system does not necessarily determine their optimal values for others, however, in the case of related chemistries there is clearly an overlap.

Additionally, it is possible that the hyperparameter space presented to the optimiser for the Opt-3k and Opt-750 simply did not include a better minimum, or that the 10% stratified sample held-out of the training data was not a representative optimisation target compared to the purpose-built test dataset that I use to

benchmark these models.

4.8 Outlook

One key takeaway from this work is that coupling an efficient and (relatively) accurate MLIP architecture (such as ACE) with hyperparameter optimisation techniques is important for fitting efficient MLIPs. Much work has been done to understand how databases affect ML models, and the best way to generate databases, especially for specific architectures.[88,197–199] However, hyperparameter optimisation has received less attention from the research community, with many seminal studies using hand-tuned[105,174] or grid-searched[121] hyperparameters. I show that for a range of disordered materials, XPOT's approach improves the accuracy of models, even when transferring from more expensive methodologies. I study silicon, wherein I find efficiency improvements upon the state-of-the-art models of the same architecture, and then consider the phase-space of Ge–Sb–Te. For GST, I extend accurate million-atom simulations from only being capable for a single composition to being able to access compositions along the GeTe–Sb_2Te_3 tie-line, using hyperparameter optimisation to create a model 2 times faster than the state-of-the-art, and which required 10 times fewer structures for training than recent DeepMD models describing GST-225.[44] Compared to the previous GAP model fitted by Y. Zhou, my optimised ACE model is 400 times faster at inference time, however, an extension of the training dataset was required. In extending this approach to Te, I study the transferability of optimised hyperparameter values, and enable studies into the crystallisation process in Te-based OTS devices. The models I fitted here also enabled further studies into the nature of chirality in Te, and simulations of mushroom-type GST devices, which I do not include here, but are discussed in great detail in Ref. 5 and Ref. 4 respectively. I believe that the use of hyperparameter optimisation (via XPOT or otherwise) to improve the accuracy of efficient MLIPs will continue to be a key component in the development of MLIPs for materials modelling, especially as the architectures of MLIPs become more complex[63,66,200] and the number of pa-

rameters continues to grow. Looking further, I view it as beneficial to combine these strategies with automated dataset generation and augmentation pipelines, such as `autoplex`[94] or `assyst`[201] to create a more automated solution to allow for more time to perform simulation and analysis instead of expending this valuable energy and effort on manual fitting procedures. I will explore these synergies in future work, and in the long run expect XPOT to become part of a larger framework (and set of workflows) for the automated and routine generation of machine-learned potentials for mainstream materials modelling.

Table 4.3: Energy and force RMSE values of silicon potentials either upfitted from the best initial potential (within the first 4 iterations, labelled Initial) or upfitted from the final optimised hyperparameters (labelled Opt). The errors are evaluated on the same three test sets. The number for the models refers to the number of atomic properties, P, from linear (1) to quaternary (4), excluding Opt-Ref, labelled after the number of functions in the potential. Adapted from Ref. 2

			Energy RMSE (meV at.$^{-1}$)			Force RMSE (meV Å^{-1})		
	P	# Func.	GAP-18	MQ-MD	RSS	GAP-18	MQ-MD	RSS
XPOT-optimised models with increasing complexity								
Initial-1	1	3000	4.2	4.0	28.7	76	110	160
Opt-1	1	3000	3.5	5.1	27.1	69	105	158
Initial-2	2	3000	6.5	3.5	192	70	104	1171
Opt-2	2	3000	2.5	5.0	23.1	63	97	150
Initial-3	3	3000	4.8	3.3	42.8	65	97	296
Opt-3	3	3000	318	5.2	$> 10^6$	300	98	$> 10^8$
Initial-4	4	3000	31.9	3.6	55.5	66	97	345
Opt-4	4	3000	4.8	5.5	62.6	63	99	274
XPOT-optimised models with varied numbers of functions								
Initial-3F	3	1375	8.0	3.8	50.9	71	104	224
Opt-3F	3	2000	4.6	4.1	20.5	65	97	139
Initial-4F	4	875	3.7	5.4	47.0	71	103	240
Opt-4F	4	1625	2.5	5.4	72.5	64	100	187
Reference models								
Opt-Ref	1	6827	3.0	4.4	34.5	63	104	179
Ref-ACE 133	1	6827	3.2	4.3	42.1	77	124	175
Si-GAP-18[55]	—	—	1.6	8.5	34.9	83	139	177

Table 4.4: Exponents of φ terms for the optimised form of each model, note that $\varphi^{(1)}$ exponents are not optimised, and are fixed at a value of 1.

Model	$\varphi^{(1)}$	$\varphi^{(2)}$	$\varphi^{(3)}$	$\varphi^{(4)}$
Opt-2	1	0.10	—	—
Opt-3	1	1.64	10	—
Opt-4	1	10	10	0.99
Opt-3F	1	0.10	0.10	—
Opt-4F	1	0.41	0.28	0.56

Table 4.5: Energy per-atom differences of quenched a-Si structures relative to (diamond-type) crystalline silicon. Structures were quenched and relaxed using the ML potentials in the top row, comparing the differences between XPOT-ACE and Ref-ACE.[133] The standard deviation across the sampled structures is reported using plus-minus notation.

Quench rate (K s^{-1})	ΔE (meV $at.^{-1}$) XPOT-ACE Quenched			
	XPOT	Ref	GAP	DFT
10^{14}	205 ± 6	209 ± 6	215 ± 7	210 ± 7
10^{13}	183 ± 4	187 ± 4	192 ± 4	185 ± 5
10^{12}	158 ± 3	161 ± 3	166 ± 4	156 ± 4
10^{11}	140 ± 4	142 ± 5	146 ± 5	137 ± 5
	Ref-ACE Quenched			
	XPOT	Ref	GAP	DFT
10^{14}	211 ± 4	208 ± 4	218 ± 3	217 ± 6
10^{13}	190 ± 4	187 ± 3	197 ± 4	185 ± 5
10^{12}	160 ± 4	158 ± 3	166 ± 4	155 ± 5
10^{11}	147 ± 4	145 ± 4	153 ± 4	144 ± 5

Table 4.6: Optimised hyperparameters, their ranges, and final values for GST.

Hyperparameters	**Description**	**Range**	**Opt. Value**
Cut-off	Cut-off distance (Å)	5.5 – 8.0	8.0
Radial parameter	Used in the construction of radial basis	4 – 10	10
Prefactor k_2	Used to define the embedding functions:	0.1 – 5	4.577
Prefactor k_3	$E_i = \varphi^{(1)} + k_2\sqrt{\varphi^{(2)}} + k_3{\varphi^{(3)}}^2$	0.1 – 5	0.101

Table 4.7: Numerical accuracy of the ablated models compared to the final model (reference). Errors are derived from the validation set. The MD speed is defined via single-core LAMMPS simulation.

	Quantities	**Energy RMSE** (meV $at.^{-1}$)	**Force RMSE** (meV $Å^{-1}$)	**MD speed**
GST-ACE-24 (reference)		**18**	**135**	**1.0**
Number of random structures	50%	17	129	1.0
	25%	16	126	1.0
	0%	15	122	1.0
Number of atomic properties, $\varphi^{(P)}$	$P = 2$	21	142	1.0
	$P = 1$	23	165	1.0
Number of basis functions, $\mathbf{B}_{iv}$	1500	22	144	1.5
	750	23	153	2.1
	300	23	168	3.1

Table 4.8: A comparison of GAP and ACE performance for models fitted to the same dataset. The GAP model required filtering of high-energy structures to achieve this accuracy, the ACE model benefitted from the higher-energy structures for robustness.

	GAP		**Te-ACE-24**	
	RMSE energies (meV atom^{-1})	RMSE forces (meV Å^{-1})	RMSE energies (meV atom^{-1})	RMSE forces (meV Å^{-1})
Liquid structures	13.8	164	8.1	124
Supercooled liquid structures	41.9	169	30.4	132
Amorphous structures	52.2	164	38.3	131
Growth configurations	26.6	142	16.7	107

Table 4.9: Comparison between different optimised ACE models for Te. The reference model is Te-ACE-24, and the two optimised models are those optimised with 3000 and 750 basis functions respectively. The RMSEs are derived from the external dataset in line with Table 4.8. Best errors are shown in **bold**.

	Energy RMSE (meV at.$^{-1}$)			**Force RMSE** (meV Å^{-1})		
	Ref.	Opt-3k	Opt-750	Ref.	Opt-3k	Opt-750
Liquid structures	**8.1**	8.7	10.1	124	125	**123**
Supercooled liquid structures	**30.4**	31.2	37.0	132	**116**	137
Amorphous structures	**38.3**	39.0	46.3	**131**	168	157
Growth configurations	16.7	**16.5**	18.3	107	**98**	114
Total	**25.6**	26.2	31.0	**124**	126	136

Chapter 5

Structural analysis and diffusivity studies of $Li_{14}SiP_6$

5.1 Acknowledgements

The work laid out in this chapter has been collated into a manuscript in preparation, from which some figures have been re-used. The work is co-authored by Christopher Davies (University of Oxford), Wilhelm Klein (Technical University of Munich), Prof. M. Saiful Islam (University of Oxford), Prof. Thomas Fässler (Technical University of Munich), and Prof. Volker L. Deringer (University of Oxford). C. Davies developed the specific site analysis package used in this work to perform site analysis (`https://github.com/ChrisDavi3s/site-analysis`) and the package used to generate Figure 5.6 (`https://github.com/ChrisDavi3s/DensMD`), and performed MSD and site analysis calculations. W. Klein performed new structural refinements. The initial creation of the Iter-0 – Iter-3 databases, and fitting of the respective GAP models is described fully in my Master's thesis, but is briefly described (as noted in the text) for completeness, as the work performed during my DPhil studies builds upon this.

5.2 Introduction

Structural disorder is an intriguing property of numerous crystalline materials [202] and is increasingly being exploited in practical applications. [203] In the field of battery materials, structural disorder plays a significant role in many crystalline cathode and solid-state ion conductors. [204,205] These materials frequently originate from relatively straightforward crystal-structure archetypes but gain complexity due to disorder. This is manifested by mixed occupancies on some or all lattice positions, and often by

vacancies that enable ion movement.[47,206] The macroscopic ionic conductivity is directly related to the microscopic mechanisms governing these atomic displacements. Examples are given by seemingly simple rocksalt-type (Li,V)O structures where Li and V atoms share the same crystallographic site,[207] by a wider-ranging class of cation-disordered oxide electrode materials,[208,209] and by $Li_{3-3x}Sc_xSb$ (structurally related to β-Li_3Sb[210]) which, despite its simple composition, exhibits exceptionally the highest reported ionic conductivity for a solid-state Li-ion conductor to date.[211]

The subject of this chapter, $Li_{14}SiP_6$,[168] is the first of a recently discovered family of disordered phosphorus-based compounds found to exhibit high Li-ion conductivity and easy synthetic access by ball-milling. These compounds are ternary Li–E–P systems, where E denotes a triel (group-13) or tetrel (group-14) element. Specific examples include the phosphidosilicate $Li_{14}SiP_6$[168] and its heavier analogues, $Li_{14}GeP_6$ and $Li_{14}SnP_6$,[212] as well as the phosphidoaluminate Li_9AlP_4[213] and the gallate Li_9GaP_4.[214] The crystal structures of these compounds can be described via a cubic close packing (ccp) lattice of P atoms, with a small (but varying) number of E atoms.

The crystal structures of these compounds are based on a cubic close packing (ccp) of P atoms, with a small fraction of tetrahedral ("T_d") voids occupied by E atoms in the structure, and a pronounced degree of disorder for the Li ions which occupy both T_d and octahedral ("O_h") voids. Understanding this disorder is important for understanding the mobility of Li ions and the conductivity trends in these materials.

When Strangmüller *et al.* first reported $Li_{14}SiP_6$,[168] density-functional-theory-(DFT-) based molecular dynamics (MD) simulations were performed to describe Li-ion mobility between T_d and O_h voids, in qualitative agreement with experiment.[168] Yet, based on DFT alone, it would not have been feasible to carry out simulations for more than a few hundred atoms or for extended timescales. Indeed, the original work contained a 5 ps DFT-MD simulation of 378 atoms. As explored in this thesis

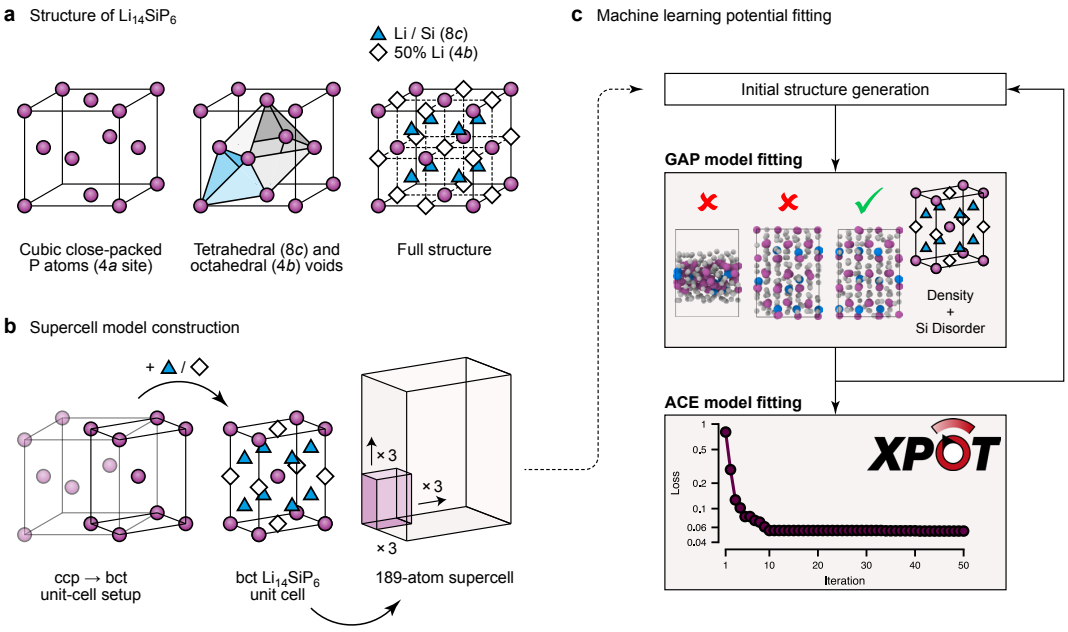

Figure 5.1: Machine-learning-driven atomistic modelling of $Li_{14}SiP_6$. (**a**) Crystal structure of $Li_{14}SiP_6$. From left to right, I show: a ccp arrangement of P atoms; examples of the tetrahedral and octahedral voids, which are centred at the 8c and 4b Wyckoff positions, respectively; and the distribution of Li and Si atoms on these Wyckoff positions. See Ref. 168 for more details. (**b**) The use of a body-centred tetragonal (bct) unit cell used to construct a supercell model of $Li_{14}SiP_6$ with discrete site occupations. A 3 by 3 by 3 supercell expansion allows us to maintain the correct stoichiometric composition. (**c**) Machine-learned interatomic potential (MLIP) fitting. In this simplified diagram, I illustrate the process using exemplary structural snapshots from early Gaussian approximation potential (GAP) model versions as well as the combined loss function for the subsequently fitted atomic cluster expansion (ACE) based models. Optimised hyperparameters for the final ACE model were determined with the help of the XPOT software.[1,2]

so far, MLIPs trained on DFT data enable significantly longer- and larger-scale MD simulations to study the atomistic structure of materials, and dynamical processes therein.[140,141,143,144] These characteristics lend themselves well to battery research, in particular alleviating the computational bottleneck for studying structures and the conduction of ions within them.[100,215]

In this chapter, I show how current bottlenecks in modelling disordered battery materials can be addressed by fitting fast, yet accurate MLIP models based on the atomic cluster expansion (ACE) framework[61,118,133] with optimised hyperparameters using XPOT.[2] I show that these ACE-driven simulations allow me to study the structure of the Li-ion disorder within the solid-state electrolyte $Li_{14}SiP_6$ by

unlocking sufficiently large structural models and long-term simulations which are not possible with *ab-initio* methods. New structure refinement is performed based on the findings of these ML simulations, improving the fit to the experimental data (as shown by reduced reliability factors). Although this study focuses on a single material as a proof-of-concept, I expect that the approach can be extended to other ion conductors, but specifically materials with related structures.

5.3 Methodology

5.3.1 Model structures and data generation

The structure of LSP follows the description of the Li–E–P systems above. Specifically, it is reported as a ccp lattice of P atoms with Si occupying ½ of the T_d (8c) sites, with Li occupying the remaining 8c sites, and half of the O_h (4b) sites.[168] Due to the complex stoichiometric composition of the structure, and the partial occupations of the 4b and 8c sites, modelling the structure computationally is not trivial, as atomistic simulations require structural models with discrete site occupations. Therefore, to meet stoichiometry, larger supercells (structures made up of several smaller repeating "unit" cells) must be constructed and Li and Si atoms placed on specific lattice sites. I now describe the construction of the original supercell structures.

In a previous study,[168] a 378-atom supercell was used for a DFT-driven MD simulation, based on a 3 by 3 by 3 expansion of the conventional (cubic) unit cell. The Si atoms were uniformly distributed throughout the supercell, whereby the locations of the Si atoms were determined by being placed at $(1/4, 1/4, 1/4)$ for two-thirds of the individual fcc unit cells. Then, the Si-containing unit cells were distributed as evenly as possible to form the 3 by 3 by 3 supercell by evenly spacing the remaining third of the unit cells which do not contain a Si atom.

Given the large number of highly-converged DFT reference computations that can be required for training MLIPs, I considered 378-atom models to be too large

for this purpose. Instead, I start from a smaller body-centred tetragonal (bct) unit cell choice (Fig. 5.1b), reducing the supercell size by half. I then use a 3 by 3 by 3 expansion of the bct unit cell to describe a large enough number of atoms (189 atoms per cell in total), retaining the correct stoichiometric composition (Li:Si:P = 14:1:6). Si atoms are uniformly distributed throughout the supercell, in accordance with the description of the structure in Ref. 168.

Neutron diffraction experiments and analysis using the maximum entropy method determined that Li atoms diffuse between T_{d}–O_{h} sites, while Si and P atoms remain in place except for normal thermal motion.[168] To represent the relevant areas of the PES for training, I displace the atoms from lattice points into physically relevant locations using two mechanisms: moving the Li atoms along the experimentally-determined pathway for Li-ion migration, and "rattling" all atoms (randomly displacing them by a short distance from their original positions) to increase structural diversity. Initially, 100 supercells of identical volume were created in this way to create an iteration ("Iter") 0 dataset.

Alongside these LSP supercells, elemental Li, Si, and P structures from the Materials Project (MP) database were added to the training set, as well as a number of binary compounds. The MP includes some computationally-based structures that are not experimentally known, which were also included. The entries for $Li_{21}P_5$ and $Li_{22}P_5$ (`mp-29720` and `mp-542598`) were excluded because their unit cells contain > 400 atoms, noting that phases with similar Li:Si ratios ($Li_{15}Si_4$ and Li_7Si_2) were present. The experimentally characterised $Li_{17}Si_4$ (Ref. 216) was also removed. Labels for all structures are generated using the PBEsol functional[217] as implemented in CASTEP.[218]

5.3.2 ML potential training

With this initial dataset in hand, I fitted a first Gaussian approximation potential (GAP) model,[52,101,149] with 20% of the LSP supercells held out as testing data. GAP is particularly efficient in the low-data regime, e.g., for early stages of a computa-

tional study where there are not yet large amounts of data available, which can be useful to kick-start exploration.[219] I iteratively fitted GAP models combining 2-body and Smooth Overlap of Atomic Positions (SOAP) terms,[60] scaled appropriately. In iterations 1–3, I performed GAP-driven constant-volume (NVT) MD and gradually increased the kinetic energy of the system, attempting to find the point at which the potential fails. When this occurred (i.e., where unreasonably short interatomic distances or the total breakdown of the structure were observed), I uniformly sampled 100 structures between the start of the simulation and a manually identified point of failure. These structures were labelled with DFT and added to the dataset for the next iteration.

After iteration ("Iter") 3, the GAP model was stable for LSP with uniformly distributed Si atoms at temperatures up to 1,800 K. By uniformly distributed, I mean that the locations of the Si atoms were determined using the same scheme as in the computational study in Ref. 168, where Si atoms were distributed by including either 1 Si atom in ⅔ of the unit cells (bct in the current work, but fcc in Ref. 168), which were then distributed evenly throughout the larger 3 by 3 by 3 supercell.

I removed the elemental and binary structures from the training data, upon which the model remained stable for LSP structures but gained accuracy. This work was completed during study for my Master's degree, but was described herein to provide context for the following work.

At this stage, all iterative LSP structures had been obtained from NVT simulations (based on experimentally determined cell parameters), and there was insufficient diversity in the training structures to perform accurate constant-*pressure* (NPT) MD up to 1,023 K. I thus focus on increasing the scope of the model to encompass varying density. I perform two iterations of NPT GAP-MD simulations. The first iteration significantly stabilised the potential, resulting in the iteration 5 structures which were closer to the room temperature density observed in experiment. Satisfied with the performance of the potentials under NPT, I turn my focus

to the possibility for Si disorder within LSP. Whereas existing structures have a pseudo-uniformly distributed Si population, I now randomly determine the location of the Si atoms within the supercell (within the constraint that Si atoms lie on 8c sites). This results in a final training dataset comprising 560 LSP structures, for a total of 105,840 atoms.

Having used GAP models to drive the iterative development of the training dataset, the cost of the model was still too high to perform nanosecond simulations of thousands of atoms given the computational resources available for this project. To resolve this, I fitted an optimised non-linear ACE model to the dataset including disordered structures. This approach was based upon the results from the previous chapter, and included 4 atomic property ($\varphi^{(p)}$) terms based on the success of a highly non-linear model by Erhard *et al.* for the Si–O system.[98] However, previous success with non-linear ACE models depended upon significantly larger datasets than the one developed for LSP, and I performed hyperparameter optimisation to explore the trade-off between accuracy and versatility of the model trained on this smaller dataset.

I optimised the ACE models with the help of my openly available "cross-platform"

Table 5.1: Force component mean absolute error (MAE) values of ML potential models fitted for $Li_{14}SiP_6$, showing the gradual improvement during iterative training. I test on three different types of data: "Ordered" structures are those generated according to my scheme adapted from Ref. 168; "Var. density" denotes NPT snapshots held out from iterations 4–5 with varying densities; and "Disordered" structures refer to the randomised Si distributions sampled for iteration 6.

	Force MAE (meV Å^{-1})		
Model	Ordered	Var. density	Disordered
GAP-Iter-0	41	301	271
GAP-Iter-1	47	163	238
GAP-Iter-2	44	143	233
GAP-Iter-3	43	136	223
GAP-Iter-4	33	96	217
GAP-Iter-5	37	74	205
ACE-Iter-6	19	39	124

optimisation package, XPOT.[1,2] By optimising hyperparameters of a non-linear ACE model, I fit a model which improves on the accuracy of my GAP-Iter-5 model, and retains the robustness up to 1800 K (Table 5.1). This particular parameterisation of an ACE model, comprising 1500 total basis functions, and four non-linear terms, reduces inference cost by over 100 times compared to the GAP-Iter-5 potential on CPU, and over 500 times vs. the MACE-MP-0b3 foundation model[67] on GPU. With my new, cheaper model in hand, I was able to study the structure of LSP using nanosecond simulations.

5.3.3 Assignment of chemical environments

The analysis of local environments in crystal structures is important for connecting atomistic simulations with experiments. In typical MD simulations, all atoms move freely in a cell that formally has $P1$ space-group symmetry (no symmetry operations other than translation), and discrete occupations are required (vis. Fig. 5.1). The outputted trajectory from an MD simulation must be reconciled *a posteriori* with the experimental crystal-structure refinement, where symmetry operations are indeed

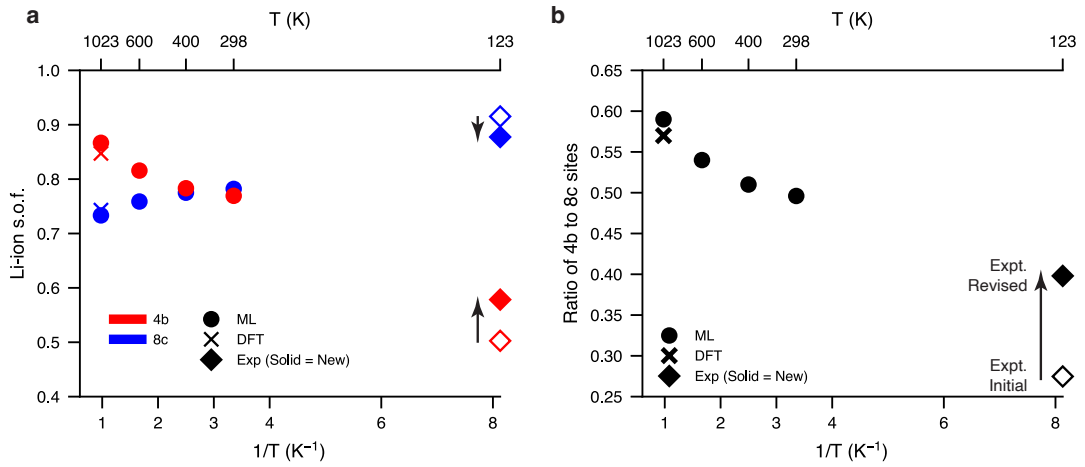

Figure 5.2: Site occupations in $Li_{14}SiP_6$ from simulations and experiment. (a) Site occupation factor (s.o.f.) values from my ML-driven simulations (circles), a previous DFT-MD study (crosses), and from the refinement of experimental data (diamonds). (b) Ratio of Li population of 4b to 8c sites: higher values indicate that O_h voids are preferred over T_d ones. In both panels, I show the initial experimental data from Ref. 168 as open symbols, and the revised values (from re-refinement informed by the simulation data; see Table 5.2) as filled symbols.

applied, and sites involving Li-ion mobility are represented by an average position. This is not a trivial task.

While there are many different schemes which can be used to assign atoms to particular sites (e.g., 8c or 4b) based on trajectories from MD,[205,220] in the case of LSP there are clearly defined phosphorus polyhedra which can be used to define the site upon which a Li-ion (the ions we are interested in) is located. Using a polyhedral mapping technique offers computational efficiency for analysing site occupation factors over the numerous structural snapshots making up MD trajectories. The site occupation factor (s.o.f.) refers to the fraction or percentage of a specific crystallographic site that is occupied by a particular type of atom or ion in the crystal structure. For each structure, the polyhedra used to assign crystallographic sites to each Li atom are recomputed based on the positions of P atoms at each timestep. No Si re-ordering is observed, in line with existing DFT analysis at 1023 K, thus, all analysis is focused on the Li ions.

C. Davies modified the `site-analysis` package,[221] to improve performance on large structures. It should be noted that there is the possibility for incorrect assignment of the site a Li atom is sitting within near the boundaries of these polyhedra (vis. Fig. 5.6a) due to the thermal motion of the P atoms. However, these events occur at the 100 fs timescale, and do not have an appreciable effect on the nanosecond-scale averages used by Chris's package to determine the s.o.f. values.

To predict Li s.o.f.s in LSP, I created a 10,206 atom supercell structure based upon the 378-atom structure used in a previous study.[168] This structure was a perfect crystal with uniformly distributed Si throughout the supercell, and Li s.o.f.s determined from the experimentally reported structure.

MD simulations were performed with a 1 fs timestep by heating under constant pressure conditions (NPT) for 50,000 steps from 298 K to the target temperature, followed by 50,000 steps of annealing to equilibrate the volume of the cell. Once this pre-equilibration occurs, the cell is annealed for at least 4 ns at the target

temperature under constant *volume* conditions (NVT) to equilibrate the system. In order to minimally disturb the dynamics of the system, the Nosé–Hoover thermostat is used, in line with best practices laid out in literature.[222]

5.4 Results and discussion

5.4.1 Li-ion site occupation factors from ML-driven simulations

Previous work[168] described a refinement of the structure from single-crystal X-ray diffraction experiments, based on the assumption of full 8c site occupancy, and DFT-driven MD at 1,023 K was performed to study the dynamics of the structure. I performed MD simulations between 123–1,023 K, using LAMMPS[178] with the ML-PACE package[133] to model the dependence of the s.o.f.s on temperature, and compare my findings to the previous DFT-MD simulation and s.o.f.s from structure refinement based on experimental data.[168] Unfortunately, simulations below 298 K were unsuitable for analysis of site occupation statistics due to a significantly reduced Li-ion "hop" frequency, even at simulation lengths of 40 ns (40 million steps), as the site occupation factor did not sufficiently converge.

I begin my analysis by comparing the 1,023 K simulation site occupancies predicted by the ACE model to the AIMD results reported in Ref. 168. The predicted occupancies lay within 3% of each other, and were significantly different to the experimentally determined s.o.f.s. As such, it seems that, despite having never been trained on varying Li occupations, the model is able to accurately extrapolate to this regime, when compared to DFT carried out with the same PBEsol functional.[223] From this, it seems that the model has not only captured the training data numerically but in terms of dynamic process behaviour too (Fig. 5.2). In Fig. 5.2b I show the ratio of the s.o.f.s to highlight the preference for each site.

5.4.2 Improving the experimental structure refinement

With decreasing temperature, my model predicts a gradually increasing occupation of 8c over 4b sites. However, the trend observed in my ML-driven MD simulations does not seem to converge towards the existing refined figures for 123 K, despite all structural models being initialised on structures with s.o.f. values determined by Ref. 168. Due to this disparity, I suggested that the Rietveld refinement of the structure based on the diffraction data of a single crystal at 123 K be re-run without preconditions on the site occupation factors of lithium. The initial refinement had fixed the s.o.fs of Li at the 8c and 4b sites to $^{11}/_{12}$ and $^{1}/_{2}$, respectively, in line with the included computational study, ensuring full occupation of the 8c sites (Fig. 5.3).[168]

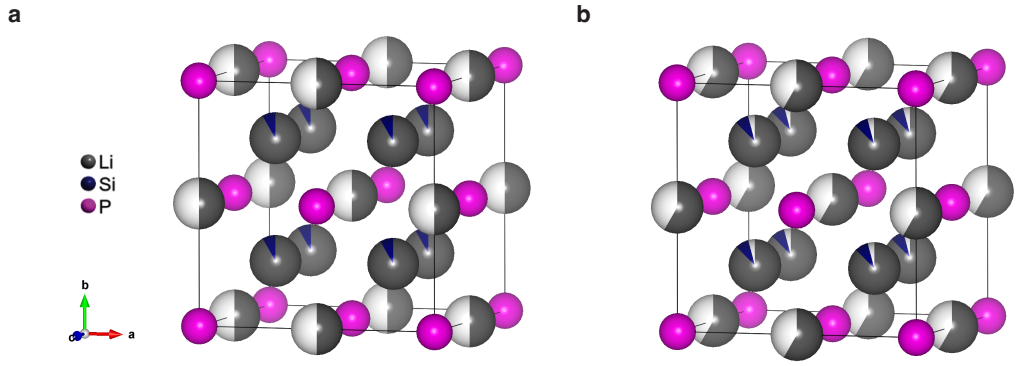

Figure 5.3: Structure of LSP. (**a**) The unit cell of LSP at 4 K as defined by the initial Reitveld refinement. 4b sites (shown as large spheres half silver, half light grey), and 8c sites (smaller spheres, $^{11}/_{12}$ silver, $^{1}/_{12}$ blue) are shown with their initial occupations. (**b**) The newly refined structure derived in this work, with free lithium occupations.

Finally, to plot the Li diffusion pathway (Fig. 5.6), I performed a 1 ns simulation at 1,023 K with the phosphorus and silicon lattice frozen in place, to enable visualisation of the Li-ion pathway w.r.t. a fixed Si/P arrangement. In relieving this constraint, a free refinement of the s.o.f.s is performed by W. Klein, and resulted in an almost 20% increase in the s.o.f. for 4b sites. Furthermore, the reliability factors (R_1 and R_2) decrease, indicating a better fit to the diffraction data. This is much closer to the trend predicted by the model, and based on the reduced relia-

bility factors, I suggest that it is a more accurate representation of the structure of $Li_{14}SiP_6$.

Beyond only LSP, I expect that ML-driven MD simulations could be useful in future work where determination of the s.o.f.s is challenging based on X-ray diffraction data alone, and that trajectory analysis can be used to guide structural models. MLIPs can be particularly useful in cases where experimental data availability is limited (e.g., in terms of resolution), or where the computational steps can be easily automated.

5.4.3 Li-ion mobility and conductivity

Simulations to investigate the ionic conductivity of LSP were performed according to the protocol discussed in Section 5.3.3. For temperatures above 400 K, NVT was run for 12 ns at the target temperature to equilibrate the structure and provide sufficient statistics to study site occupations independently of the starting configuration. For temperatures below 400 K, I ran MD for up to 40 ns to try and reach the ergodic regime and ensure the validity of the simulations for use in Arrhenius analysis. This was based on initial studies of 4 ns not sufficiently reaching the ergodic regime (in this case, failure to reach the Fickian diffusion regime).

Table 5.2: Results of refinements of single-crystal X-ray data recorded at 123 K for $Li_{14}SiP_6$, with Li s.o.f.s (from left to right): fixed to obtain completely filled tetrahedral voids (Ref. 168); freely refined; or fixed with values obtained from MD simulations at 298 K; **bold numbers** represent refined values.

	Ref. 168	Free refinement	MD (298 K)
s.o.f.(P)	1	1	1
s.o.f.(Si)	1/12	1/12	1/12
s.o.f.(Li1)[8c]	11/12	**0.8775(4)**	0.7820
s.o.f.(Li2)[4b]	1/2	**0.5784(7)**	0.7693
U_{iso}(P)	**0.0263(5)**	**0.0263(5)**	**0.0265(6)**
U_{iso}(Si/Li1)[8c]	**0.0191(9)**	**0.0180(9)**	**0.0158(10)**
U_{iso}(Li2)[4b]	**0.09(2)**	**0.12(2)**	**0.28(7)**
R_1 [$I > 2\sigma(I)$]	0.0410	0.0397	0.0429
wR_2 [all data]	0.0907	0.0897	0.1026

Reliable diffusion coefficients via the Nernst–Einstein relation, and subsequent calculations of the conductivity, σ, necessitate ergodic simulations in the diffusive regime. This condition is confirmed by observing linear Mean Squared Displacement (MSD) versus time lag (MSD $\propto t$), and the vanishing of the non-Gaussian parameter (NGP), $\alpha_2(t)$, at the long time limit of the simulation.[224] The non-Gaussian parameter helps to characterise the timescale over which particles most strongly deviate from Gaussian statistics during the ballistic-to-Fickian transition. A single peak in $\alpha_2(t)$ indicates maximum heterogeneity. For LSP this timescale becomes shorter as temperature increases, supporting a single, thermally activated hopping process visualised by the tranisition from caged to mobile states.

Simulations in the high-temperature regime (600 K and above) achieved these within 12 ns (c.f. Fig. 5.5). However, lower temperature simulations, while reaching equilibrated site occupations, and the MSD $\propto t$, show a persistent non-zero NGP, even after 40 ns of simulation. This indicates a breakdown in ergodicity due to inaccessibililty of sufficiently long timescales, i.e., individual particle trajectories maintain distinct behaviours despite ensemble-averaged Fickian diffusion (MSD $\propto t$). Therefore, the simulations below 400 K cannot be used for calculating valid conductivities, and therefore, I exclude them from my analysis of the activation energy via the Arrhenius equation. These simulations, however, *do* equilibrate the site occupations, and are therefore valid for studying the Li-ion s.o.f.s.

LSP has an approximately 10 times lower ionic conductivity than the recently discovered $Li_{2.55}Sc_{0.15}Sb$, and compounds such as these with higher conductivities may converge upon the ergodic regime more quickly. However, materials with very high conductivities often display highly correlated ionic motion, which can invalidate the Nernst–Einstein relation, and require alternate techniques such as Green–Kubo analysis.

Initial insights into Li-ion hopping dynamics were gained statistically. Specifically, C. Davies examined the self-part of the Van Hove correlation function (VHCF),

$$G_s(r,t) = \frac{1}{N}\langle \sum_{i=1}^{N} \delta(r - |\mathbf{r}_i(t) - \mathbf{r}_i(0)|)\rangle,$$

to identify characteristic hopping events through the emergence of distinct peaks at increasing time lags t.

At 1,023 K, Li-ion "hopping" events (diffusion) occur on the timescale of $\approx$ 1 ps, visualised by the NGP and van Hove analysis. By contrast, at 123 K single hops only occur on the $\approx$ 1 ns timescale, and insufficient diffusion was observed to be able to draw conclusions of significance. The van Hove analysis (Fig. 5.4) shows that diffusion is centred around the 4b–8c distance of $\approx$ 2.5 Å, the location of the secondary peak in the 1,023 K spectrum.

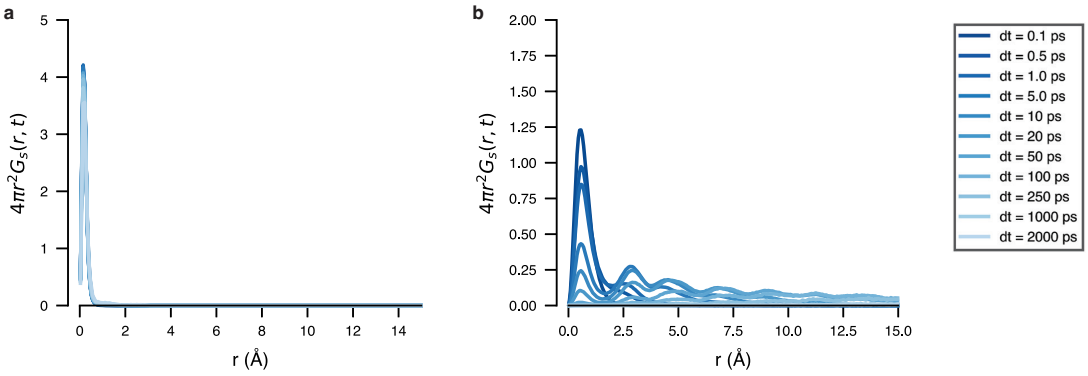

Figure 5.4: van Hove analysis for LSP. (**a**) shows the van Hove correlation function for Li-ion diffusion in LSP at 1,023 K, (**b**) shows the van Hove correlation function for Li-ion diffusion in LSP at 123 K. The distribution of the distance of each Li-ion from its initial positions at increasing time lags is shown, demonstrating the lack of diffusion at low temperatures.

Finally, to plot the Li diffusion pathway (Fig. 5.6), I performed a 1 ns simulation at 1,023 K with the phosphorus and silicon lattice frozen in place, to enable visualisation of the Li-ion pathway w.r.t. a fixed Si/P arrangement.

From my MD simulations between 600–1,023 K, I plot ionic conductivity against temperature (Fig. 5.5b). Based on the fit to the ionic conductivity at each temper-

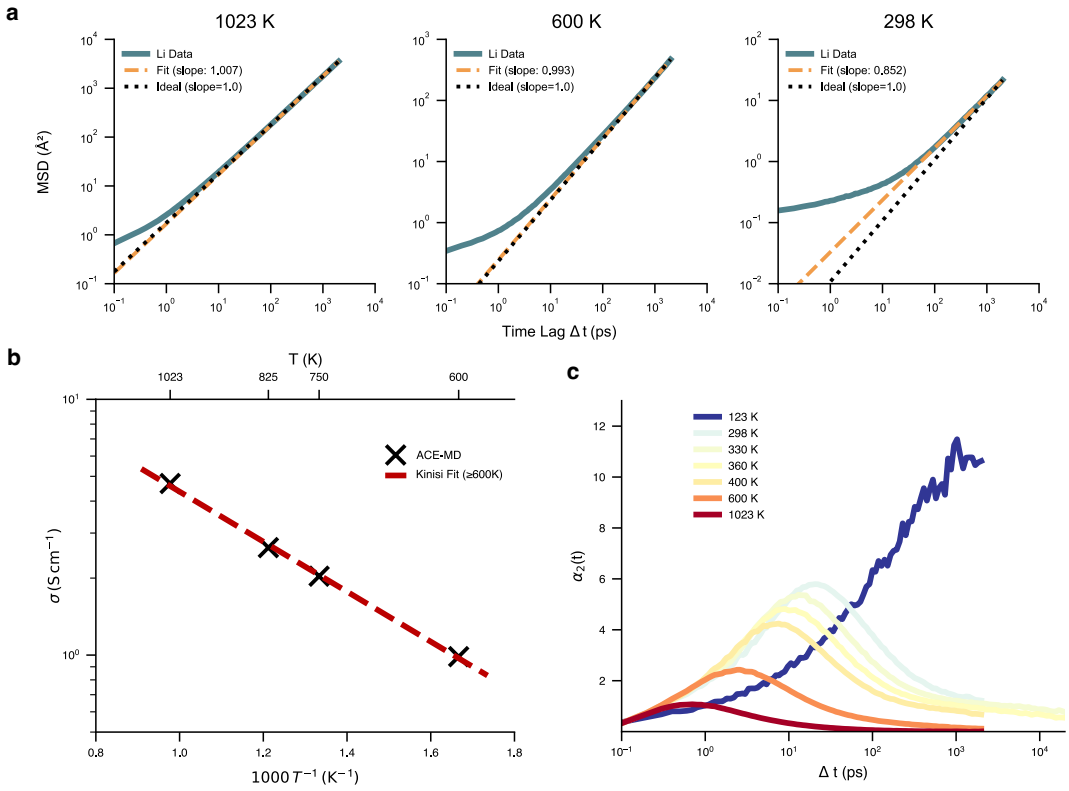

Figure 5.5: MSD, NGP, and Arrhenius plots for MD simulations. (**a**) The MSD against time lag plots used to determine the validity of the Nernst–Einstein relation. (**b**) Arrhenius plot for the validated temperatures, and activation energy determined from temperature dependence of conductivity. Activation energies were estimated for ACE-MD trajectories using the Kinisi package (v1.1.1).[225] (**c**) NGP of lithium in 10k-atom, trajectories (330 K/360 K runs extended to 40 ns to show NGP non-convergence at long timescales). The lack of vanishing NGP at long timescales for simulations below 600 K invalidates their use for determining the activation energy.

ature, I derive an activation energy of 25.4 kJ mol^{-1} (0.264 eV at.$^{-1}$) The errors defined by Kinisi are $< 10^{-5}$ eV at.$^{-1}$. Experimental values from temperature dependence impedance give an activation energy of 32.2 $\pm$ 0.6 kJ mol^{-1}, while NMR-derived values via the Waugh–Fedin relation predict an activation energy of 30 kJ mol^{-1}.[168] Further NMR studies via temperature-dependent determination of the activation energy predict that $E_a = 15$ kJ mol^{-1}.[226] The spread in these experimental results is indicative of the different techniques and synthesis outcomes. In relation, our value sits between these values, and is broadly comparable.

Due to the differences between the experimental and simulated material (introduced through approximations required by simulation size), I do not expect the activation energies to match exactly. Firstly, the experimental pellet used to measure the conductivity is synthesised via ball-milling, resulting in a non-zero number of grain boundaries. In this project, we do not include grain boundaries in the structures, and indeed the structure is a single crystal, directly comparable to the sample created for X-ray diffraction measurement. Secondly, there is no determination of the local Si distribution within any of the synthesised structure, but it is an idealised case to assume that the silicon atoms will be distributed evenly throughout the structure. As such, the computational structure is a proxy for the idealised case of a single crystal synthesised with ideal Si distribution. This is done in this study due to the length and size of simulation required to perform robust conductivity measurements for the activation energy. In reality, if there is significant variance in the local distributions of Si atoms, pathway blocking, and other effects may come into play, affecting the bulk ionic conductivity. As such, based on these two factors, I expect that the activation energy will be reduced, and significantly higher conductivities would be predicted at low temperatures owing to the lack of insurmountable energy barriers to Li-ion transport.

5.4.4 Microscopic insights into diffusion pathways

Having established the performance of the model and investigated the nature of the Li-ion movement and distribution in LSP, I analysed the Li-ion diffusion pathway in more detail. Based on the large scale simulations, 4b–4b and 8c–8c transitions do not occur at any meaningful frequency and as such, I do not study the energy barriers (c.f. Fig. 5.4, Fig. 5.5c). To study the Li-ion diffusion pathway between 4b–8c sites, I first perform DFT of representative structures along the pathway, and compare them to predictions from my model.

To define the relevant structures, I select two pathways. The first is the experimentally derived "indirect" (bent) pathway, which proceeds along an offset path

(shown in white in Fig. 5.6a).[168] Secondly, based on qualitative observations of the diffusion pathway in my simulations, I define a "direct" pathway which connects the 4b and 8c sites via the shortest possible route through the centre of the triangular face shared by both coordination polyhedra (this pathway is shown in teal in Fig. 5.6a).

To better understand the predicted energetic differences of these pathways, I perform small-scale computational analysis on the LSP system. I investigated the energy profile of Li-ion diffusion from the 8c site to an empty 4b site in a perfect crystal cell. The structure is a perfect crystal using the 4 K lattice parameter, and the Li-ion is moved along the pathway by a single atom. I note that compared to MD simulations, where all atoms move freely and the lattice experiences thermal motion, this is a simplified model of a single Li-ion diffusion event, as if the lattice were static. Thus, the absolute values of $\Delta\varepsilon_{ML}$ are not directly comparable to those in the experimentally derived one-particle potential (OPP) calculated from neutron scattering data taken at 1,023 K in a previous study.[168]

To confirm that any energy predictions were not an artefact of the ML model, I performed a DFT study of the potential energy throughout both Li-ion diffusion

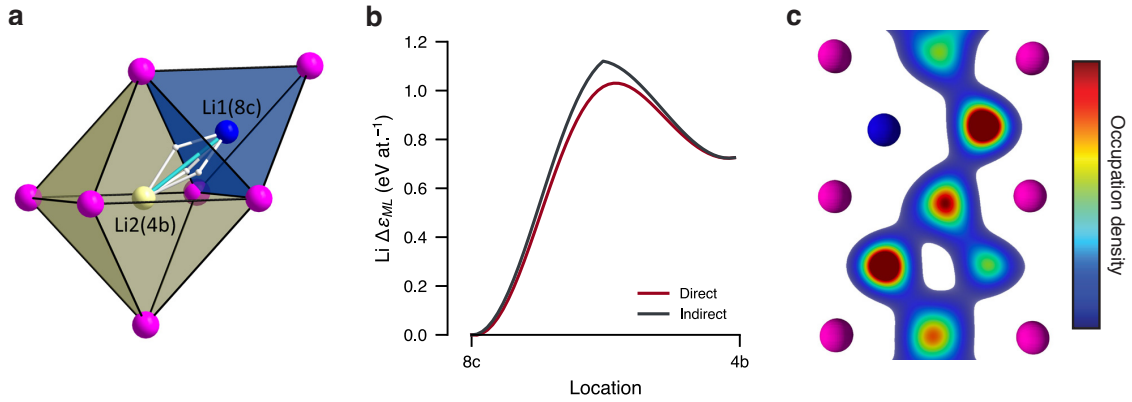

Figure 5.6: Microscopic insights into the Li-ion diffusion pathway in LSP. (a) A structural model of the indirect pathway proposed in Ref. 168 (white) and the direct pathway observed in ML-driven MD (teal), drawn similar to Ref. 168. (b) Li-atom atomic energy for movement from T_{d}–O_{h} site. (c) A heatmap of the Li-atom locations w.r.t. the fixed phosphorus lattice, viewed at (110) with P and Si atoms as violet and blue spheres, respectively. I observe Li diffusion along the T_d–O_h–T_d pathway.

pathways from a T_d to an O_h site, providing a benchmark for my model. As DFT can normally only give access to the total (per-cell, not per-atom) energy, I use this quantity to test my ACE model, resulting in an MAE $< 0.1\,\mathrm{meV\,at.}^{-1}$ for structural snapshots along the pathways when compared to DFT. Thus, at least on a per-cell basis, the model accurately recreates the DFT predicted PES.

I discern the energy profile of the Li-ion along the diffusion pathways via per-atom energy predictions from the ACE model. I achieved this by querying 50 linearly distributed snapshots of the Li-ion along each pathway, querying the energy of the atom at each snapshot. In Fig. 5.6b, I show that the direct pathway is predicted to be energetically favourable by my model when compared to the offset pathway introduced in Ref. 168. The comparison between the two pathways suggests a ≈ 0.1 eV energy barrier increase for the indirect pathway, and explains the qualitative finding that a direct pathway seemed to be preferred in simulation.

As I mentioned above, it is not necessarily possible to directly compare to the OPP, but I note that the relative energies of the 4b and 8c sites are seen to be significantly further apart. Qualitatively however, the general form of the activation barrier is retained, with the maximum energy predicted at the point at which the Li atom passes through the face-sharing plane of the polyhedra, along both the indirect and direct pathways.

To understand the effect of this finding on simulation, I perform a simulation wherein the P atoms are fixed, such that a heatmap of the Li-ion motion in relation to a fixed lattice and the heatmap directly represents the Li-ion atomic positions. I visualise these occupation densities, using a heatmap plot (Fig. 5.6c) along the 110 $d = 1$ plane, of Li-atom positions throughout simulation, demonstrating the preferential 4b–8c Li-ion diffusion and the lack of a clearly offset pathway. It is possible to observe the effect of a Si atom on the Li-ion diffusion on the neighbouring pathways, whereby the neighbouring Li-ions occupying 4b sites are slightly offset away from the Si atom. Additionally, the occupation (i.e., peak value in the heatmap) of each 8c

site may be different as the data was collected from a single 1 ns simulation used to generate such an average, resulting in specific sites (of the same type) experiencing different occupancies.

5.5 Outlook

Machine-learning-driven atomistic simulations are increasingly central to modelling and understanding battery materials, and they can be particularly useful in cases where ionic mobility is linked to structural disorder. In this chapter, I built upon the findings of the last chapter in fitting ACE models to large, existing datasets, by fitting ML interatomic potential models to a significantly smaller, more targeted dataset taking into account crystallographic information about mixed site occupations. I ran large-scale simulations using computationally efficient optimised non-linear ACE potentials, and I showed that the results of my simulations were able to improve the Rietveld refinement of experimental data for $Li_{14}SiP_6$, and the understanding of the compound's structure.

I expect that simulations such as those performed in this chapter, alongside systematic protocols and workflows, will become more commonplace in modelling disordered ion conductors: accessing descriptions of partial site occupations and high-temperature phases in a realistic manner. In doing so, one aims to, through comparison with experiment, further our understanding of these materials, and in time create materials which offer improved safety, performance, and longevity in batteries. For future work, I believe uncertainty-based sampling will be useful to efficiently create wide-ranging datasets which can account for grain-boundaries and other structural features found in experimental samples. While I studied here a member of the phosphidotetrelate class of ion conductors, I believe that this approach will be used more generally for studying ion conductors, especially in the creation of initial datasets via sampling from experimental findings. For example, work is underway to describe the $Li_{3-3x}Sc_xSb$ ion conductor, described in Ref. 211, and the field at large is investigating a number of battery materials and ion diffusion

processes, ranging from Li–P–S phases,[206] to the red phosphorus-based anodes for sodium-ion batteries.[227]

Chapter 6

Distillation using Synthetic Data

6.1 Acknowledgements

The work laid out in this chapter has been pre-printed on arXiv,[3] and is under review, some text, and figures labelled as "adapted" have been re-used from this manuscript. This work was joint-first authored by John Gardner (University of Oxford) and I. We conceived the study, designed experiments, and coordinated the project, with input from Prof. V. L. Deringer. J. Gardner developed the synthetic data generation (`augment-atoms`), performed the fine-tuning of the foundation models, and performed distillation of the GNN models. The contributing authors were Chiheb Ben Mahmoud, Zoé Faure Beaulieu, Veronika Juraskova, Laura-Bianca Paşca, and Louise A. M. Rosset. Primary investigators were Fernanda Duarte, Fausto Martelli, Chris J. Pickard, who provided individual expertise on the different applications. Z. Faure Beaulieu performed the processing of the simulation data for water, and chose the most relevant benchmarks with the help of F. Martelli. I trained all ACE models, and performed the analysis of the results included in this chapter.

6.2 Introduction

Thus far, I have focused on the modelling of *ab-initio*-derived potential energy surfaces, taking hundreds or thousands of DFT-labelled structures to train an MLIP without any data-derived coefficient priors. The cost of labelling such quantities of training data, and generating a sufficiently diverse dataset, continues to be a significant hurdle for the use of MLIPs in materials study. Work continues towards democratising MLIPs by reducing the barriers to entry, including via automated database generation such as hyperactive learning or GAP-RSS,[88,228] advances in

model architectures such as MACE or non-linear ACE models,[63,118] and a new class of "foundation models" that can be used as a baseline from which to learn.[16,67]

The term "foundation model" (FM) is used to describe machine-learning models trained on large, diverse, and therefore general datasets, which demonstrate high accuracy and generalisation behaviour out-of-the-box.[229] Such models have been developed in many fields, transforming the way in which research is carried out across natural language processing, computer vision, meteorology, medical and physical sciences.[16–21,230,231]

As an example, recently the Aurora model was trained on data comprising weather forecasts, ocean-wave and hurricane dynamics to create a foundation model for the Earth system as a whole.[18] Not only did this consolidate predictive power into a single model, but Bodnar *et al.* report improved accuracy vs. specialised state-of-the-art models for wave height, cyclone paths, and wind forecasting.

In the field of atomistic simulations of materials and molecular systems, foundation models are specifically machine-learned interatomic potential (MLIP) models that have been trained on very large datasets of diverse chemical systems, although these often still have a speciality on either materials or molecules.

For solid-state materials, notable examples include MACE-MP-0,[67] MatterSim,[232] Orb,[69,70] OMat24,[71] and the current numerical state-of-the-art: eSEN.[79]

In the realm of molecular chemistry, the ANI and AIMNet model series are well established,[72,233] while the MACE-OFF[73] and OMol25[87] models have been proposed more recently, after their materials counterparts were successfully introduced.

While the current norm is for FMs which specialise in either molecular or materials chemistry, in recent months some models which can provide accurate predictions for both have been published.[74,75] The accuracy of these foundation models, and MLIPs in general, continues to improve over time. Powering this are a series of improvements from architectural innovations[67,78,79] to training protocols,[85] improved datasets,[71,86,87,234] and enhanced validation and benchmarking techniques.[82,83]

In order to harness the benefits of these advances, it is crucial to find ways to reduce the computational cost of predictions using these FMs, as to capture the wide-ranging chemistry of these datasets at an acceptable accuracy large, expensive, models are used. The problem of cost is not a new one in the field of machine learning, and significant resource has been spent finding ways to mitigate this, including by transferring information from expensive models to cheaper ones which can be used to make predictions. Hinton *et al.* introduced "knowledge distillation": a process whereby information learned by an expensive model is transferred to another, cheaper, model.[235] These models are often labelled the "teacher" and "student" respectively, and the technique has been applied to ML models in many fields to great success.

For MLIPs, research has focused on using synthetic data, the same technique which propelled the disruptive DeepSeek family of large language models to fame in 2024.[10] Synthetic data is data which has been labelled by an approximation of the "ground-truth", in this case by a "teacher" model instead of the DFT ground truth method. Initially, aiming to reproduce a teacher model's total-energy and force predictions was considered, and this is the most classical approach (that is, directly asking the model to learn the values it would from DFT, but here with synthetic labels).[59,70,75] However, further developments also include extending the approach to focusing on the teacher's *local* energy predictions,[114,236] learning the teacher model's Hessians,[237] aligning the internal representations of the teacher and student models,[238] or learning from an ensemble of teachers.[239]

The use of synthetic data for MLIP training was initially explored in the context of fast linear models,[59] then of graph-neural-network models,[116] and more recently for the DPA models.[240] However, most work in this space reported to date has focused on specific MLIP architectures, and often used domain-specific data-generation protocols (such as custom MD simulations in Refs. 59 and 114).

The key approach in this work is to determine a general, efficient, architecture-agnostic protocol for distilling any FM to target any specific chemical system (Figure 6.1) with a small initial investment in *ab-initio* labelling. The approach outlined in this chapter produces fast, accurate, and stable "student" models to drive large-scale simulations without a significant loss in accuracy or needing the power of a state-of-the-art HPC system. Being based on generalised data-driven knowledge transfer, this approach transcends specific MLIP fitting frameworks, and indeed one of its most promising applications is one that does *not* require a computationally expensive graph-network architecture for production simulations. Building on previous work that has shown that (i) MLIPs can be trained based on local atomic environments,[51,52] (ii) graph-based MLIPs provide unprecedented accuracy and flexibility,[62,63,67] and that (iii) graph-based MLIPs can be used to fit chemically comprehensive models that scale with data,[67,71,75,232,241] I posit that distillation will become a crucial step in democratising MLIPs for computational study of chemistry.

I focus herein on the application of the method to ACE models, as well as emphasising the validating of XPOT hyperparameters for synthetic training and compare to existing parametrisations of ACE for simulating water and ice, as these are the MLIPs that I directly trained, however, further information on the variety of chemistries we have studied using this distillation approach can be found in Ref. 3.

6.3 Methods

6.3.1 Synthetic data generation

To significantly expand the original training database, a synthetic data generation protocol is implemented. It iteratively augments each structure from the initial, smaller dataset (of DFT-labelled structures used to fine-tune the FM) to create a large and diverse collection of atomic configurations. To do so, a "rattle-relax-repeat" protocol is defined, and applied for each structure, where structures are added to

the dataset after relaxation. Crucially, this circumvents the need for MD using the FM, and instead only requires single-point labelling, which is computationally inexpensive using an FM (especially when compared to DFT).

The exact form of the selection of the protocol is described in detail in Ref. 3, and, as the protocol used is the same for all distillation applications in this chapter, I do not further describe it here. However, it can be summarised as follows: through iterations which build upon each other, the protocol generates an increasingly diverse set of structures where exploration can be tuned via changing of parameters in the protocol.

6.3.2 Fine-tuning foundation models

Due to the nature of foundation models (i.e., that they are generalising across the entire periodic table), and the fact that a single set of DFT parameters is normally used for the entire dataset for consistency, the predictive power for zero-shot FM-MD is often not sufficiently accurate for domain experts. Fine-tuning of foundation models (even with small numbers of structures) can improve accuracy significantly for the system of interest.[75,116]

Additionally, this allows one to describe the PES using a functional, and DFT parameters, of choice (i.e., those that are most suitable for replicating experimental results or are best suited to the system of interest). For this purpose, we use only existing datasets which were used for the study of their respective chemical systems, and which are all using different levels of theory or parameters to the large, general-purpose dataset which the foundation models were trained on.

To demonstrate the architecture-agnostic nature of not only the distillation, but also the fine-tuning approach, we use `MatterSim-v1.0.0-1M`[232], `MACE-MP-0b3`[67], `MACE-OFF24`,[73] and `orb-v3-direct-20-omat`[70] as teacher models.

In the following work, all foundation models are fine-tuned using `graph-pes`, with 25 representative structures used for training, and 5 for validation, unless otherwise stated. All FMs were fine-tuned with a batch size of 2. A fresh AdamW[242] optimiser

was used with an initial learning rate of 1×10^{-3} with a decay on plateau factor of 0.8, with patience 25 in relation to the validation loss.

6.3.3 Student models

Distillation of the GNN-based models using PaiNN[243] and TensorNet[244] is discussed comprehensively in Ref. 3. I performed training of all ACE student models with `pacemaker`.[118]

Two classes of ACE model were trained:

- **XPOT models**: These models were trained using XPOT-determined optimised hyperparameters.
- **Reference Models**: These models are based on the parameterisation of Ref. 173, and are discussed in Section 6.4.3

I performed optimisation of hyperparameters based on a fixed cutoff of 5.5 Å, training on 333 structures and dimers, with 250 structures used for validation. The number of structures is selected to act as a cost-effective intermediate value from the distribution used for ablation studies.

50 iterations were performed with 8 initially sampled points in hyperparameter space. The hyperparameters, their ranges, and their optimised values are summarised in Table 6.1. An embedding of the form:

$$E_i = \varphi^{(1)} + \sqrt{\varphi^{(2)}} + \varphi^{(3)\,\frac{3}{4}} + \varphi^{(4)\,\frac{1}{4}} + \varphi^{(5)x_1} + \varphi^{(6)x_2} \tag{6.1}$$

is used, where x_1 and x_2 are the exponents for $\varphi^{(5)}$ and $\varphi^{(6)}$ which I optimised with XPOT.

A customised version of `pacemaker` was used to perform the training. I added the AdamW[242] optimiser via its `keras` implementation into `tensorpotential`, the evaluator used in `pacemaker` training protocols. No changes are made to the loss function, and coefficient updates are made batch-wise instead of epoch-wise as in

Table 6.1: Hyperparameters optimised for the ACE student models.

Hyperparameter	Values	Optimised value
$\varphi^{(5)}$ exponent	0.375, 0.125	0.375
$\varphi^{(6)}$ exponent	0.625, 0.875, 2	0.625
Radial basis	SBessel, ChebExpCos, ChebPow	SBessel
Basis functions	100–1000	540

the BFGS implementation. Thus, I reduce the disparity in training time required when increasing the size of the training database.

With optimised hyperparameters and customised `pacemaker` in hand, I train the reference and XPOT models using the same procedure. For the reference models, I take the final model provided by Ibrahim *et al.*,[173] and reset all coefficients to zero, (i.e., the model is in its initial state). The training process then involves two steps: firstly, the model is trained using a fresh AdamW optimiser with an initial learning rate of 1×10^{-3}, batch size of 32, and $\kappa = 0.75$. All models have an automatically determined Ziegler–Biersack–Littmark (ZBL) repulsive term dependent on the shortest interatomic distance in the training set. I implement an early stopping criterion based on the validation loss, whereby if the model fails to improve by 1×10^{-5} per epoch for any 50 epochs, the training is terminated. This is added as insurance against overfitting. To converge the energy predictions of the model, I take the fitted model and perform BFGS optimisation for 50 epochs with $\kappa = 0.01$.

6.4 Results and discussion

6.4.1 Proof of concept: water

As a proof of concept, we distil the `MACE-MP-0b3` foundation model[67] upon structures representative of liquid water at room temperature and pressure, aiming to create much faster "student" MLIPs at the hybrid-DFT (revPBE0-D3)[223,245–247] level of theory. Figure 6.1a illustrates the protocol used, and the steps involved.

Before beginning the process, the dataset for the fine-tuning must be selected.

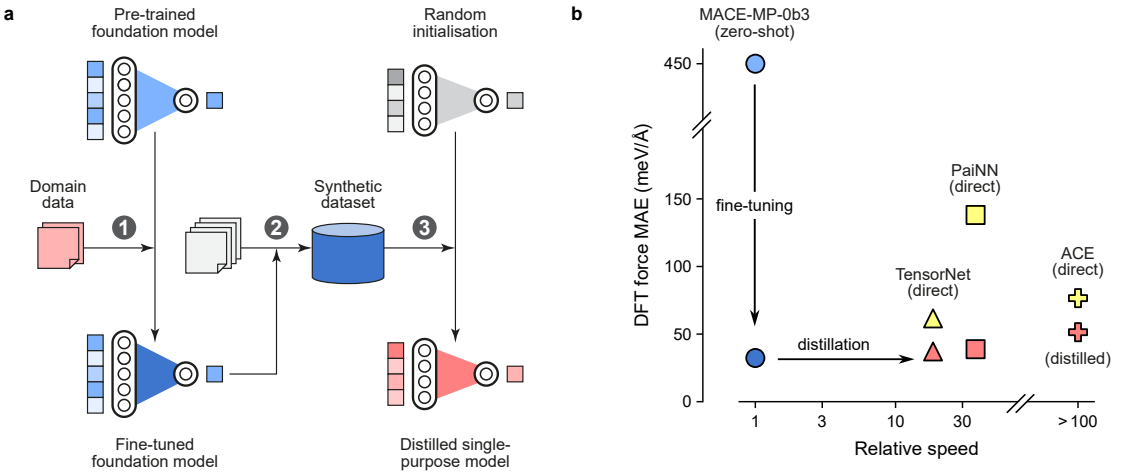

Figure 6.1: Distilling atomistic foundation models via synthetic data. (**a**) Illustration of the three-step-procedure used throughout the present work. **1**: An existing foundational MLIP model (*light blue*) is fine-tuned on a small amount of domain-specific structures with quantum-mechanical energy and force labels (*red*). **2**: This fine-tuned 'teacher' model (*dark blue*) is then used to cheaply generate and label a much larger dataset of synthetic structures, similar to Ref. 59 but now without the need for MD simulations (Methods). **3**: Finally, a small and fast 'student' model is trained on this synthetic dataset in a near-automated fashion, yielding the distilled single-purpose MLIP model highlighted in red. (**b**) Proof-of-concept for this protocol for liquid water. Relative to the foundation model, distilled models with the `TensorNet`, `PaiNN` and `ACE` architectures are significantly faster for nearly the same accuracy on the DFT-labelled test set. We include models directly trained on the domain-specific DFT dataset for comparison (*yellow markers*): these have seen only small amounts of data and are uniformly worse than their respective distilled counterparts (*red*). Figure reproduced from Ref. 3.

Here, the DFT-labelled dataset from Ref. 248 is used, which is a high-quality dataset for bulk liquid water, labelled at the revPBE0-D3 level of theory.

The dataset is split into a training set of 25 structures, a validation set of 5 structures, and a test set of 1,563 structures. The training set is extremely small for two reasons: one, previous studies on fine-tuning MACE models has demonstrated accurate results at similar scale,[77] two, a small, high-quality dataset exemplifies the case in which distillation is most useful: to quickly generate a large dataset of synthetic structures without the need for expensive DFT calculations.

Step 1 (vis. Fig. 6.1) is to fine-tune the foundation model on the dataset. `MACE-MP-0b3` was trained using on data labelled with the PBE+U functional. PBE+U is a GGA functional with an additional Hubbard U term to account for the on-site Coulomb repulsion.[249] However, the rev-PBE0-D3 functional is a hybrid DFT func-

tional that includes a fraction of exact exchange, and is therefore more accurate than PBE+U, requiring the model to learn this higher level of theory in order to predict labels from our fine-tuning dataset. As a result, fine-tuning drastically improves accuracy on the test set (component-wise force R^2: 0.944 $\rightarrow$ 0.9991, component-wise force MAE: 450 meV Å^{-1} $\rightarrow$ 32 meV Å^{-1}, Fig. 6.1b).

Step 2 is to generate structures and label them to form the synthetic dataset. The fine-tuned FM is used to augment the starting dataset of 25 + 5 training and validation structures via the iterative sequence of rattling and relaxation steps mentioned Section 6.3.1 and described in detail in Ref. 3. This protocol is extremely sample-efficient, requiring an average of five model inference calls to produce each new, uncorrelated structure - at least an order of magnitude more efficient than MD. The fine-tuned FM is then used to label the resulting 10k structures, as well as 50 structures for each H–H, O–O and O–H dimer.

Finally, step 3 is to train a variety of efficient student MLIPs on the synthetic dataset, allowing one to take advantage of the improved computational speed compared to the FM. As the teacher model is treated as a "black box" in this approach, it is *architecture agnostic*, and results are presented for a number of architectures upon the water dataset. `TensorNet`, `PaiNN`, and `ACE` models are fitted, with force errors relative to DFT that approach those of the fine-tuned FM (component-wise force MAEs of 37, 39, and 51 meV Å^{-1} respectively); see Fig. 6.1b. These frameworks were chosen to represent a range of architectures and possible speed-ups from 10 times to 100 times faster than the FM, enabling these simulations to be performed on a single GPU for large system sizes, unlike the FM (Fig. 6.2). Additionally, `EDDP` potentials were trained as a comparison to a method which only trained on energies, and not forces.[104]

When comparing the numerical errors for "direct" training (training on the 25 DFT-labelled training structures) versus distillation, an up to 70% reduction in errors is observed. However, the numerical errors do not sufficiently describe the

differences in behaviour of the models. While all the direct student models consistently fail in MD, predicting unphysical structures and losing atoms, the distilled models are capable of driving MD simulations that are stable.

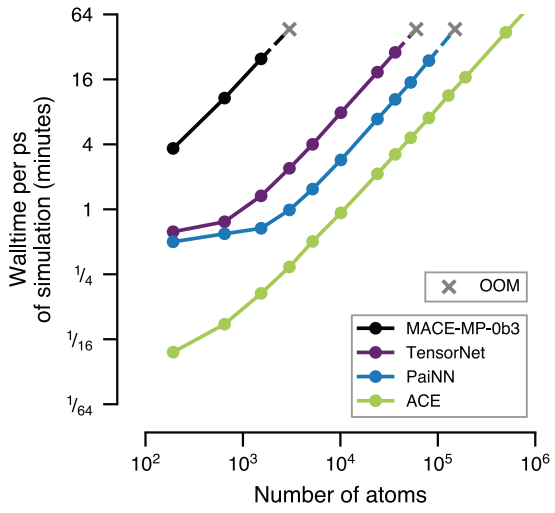

Figure 6.2: Computational efficiency. Inference costs for different water models. Speeds are measured during 300K NVT simulations run on a single Nvidia RTX A6000 GPU). The (fine-tuned) `MACE-MP-0b3` model is memory-intensive, and quickly runs out of memory (OOM) as system size increases beyond 1,000 atoms. In contrast, the smaller, distilled, GNN models can continue to scale system size by up to another 2 orders of magnitude. Finally, ACE provides another order of magnitude speed-up, and allows for scaling to extremely large systems (OOM occurs at $\approx 6.7 \times 10^7$ atoms). Figure reproduced with permission from Ref. 3.

In total, (assuming known hyperparameters), distillation via our approach takes under 8 hours to go from starting FM to distilled model on a single, mid-level GPU (Nvidia RTX A6000): 30 minutes to fine-tune the foundation model, 3 hours to generate and label 10,000 synthetic structures, and $\approx$ 4 hours to train the distilled model. Hence, with a comparably small investment of resource and energy, and using only a single GPU, the user can speed up the simulation as compared to the expensive foundation model.

Hyperparameter optimisation increases the cost of the training process, and performing optimisation with XPOT took 24 hours (that is, $\approx$ 30 minutes per model

fit). The fitting process was undertaken by reducing the number of structures trained upon (333), and the total number of epochs (500 for XPOT optimisation fits, up to 2500 for final model fitting). I provide further details on ACE models specifically, and comparisons of the various parameterisations in Section 6.4.3.

Beyond numerical validation, further tests are required to ascertain the appropriateness of an MLIP for any given task.[79,102] For the models trained on water, we define the quality of a model dependent on its description of liquid water in MD simulations. Both the distilled models and `MACE-MP-0b3` provide sensible local structure via MD in terms of the nearest-neighbour environment, as shown by the radial distribution function (Fig. 6.3a). Consistent with their respective speed-ups, the PaiNN model (approximately 10 times faster than the teacher) yields a very similar local structure to experiment and the teacher; the ACE model (approximately 100 times faster than the teacher) is different and appears marginally over-structured: the first peak in the RDF, indicating hydrogen-bonded O–H$\cdots$O contacts, is more pronounced compared to experiment.

Further understanding can be gained from the shortest-path ring distribution which indicates the medium-range ordering of the liquid (Fig. 6.3b), and from the tetrahedral order parameter of Ref. 251 for local environments (Fig. 6.3c). The trend observed in the RDF, whereby PaiNN underorders compared to experiment and FM, and ACE overorders, is visible across all tests in Figure 6.3. The increased prevalence of 6-membered rings for the ACE-driven structure in Fig. 6.3b indicates again the higher order, and a perfect hexagonal ice would solely consist of 6-memebered rings. In Fig. 6.3c, the tetrahedral order parameter reinforces the observed predictive biases of PaiNN and ACE.

In Fig. 6.3 we inspect the hydrogen-bonding (HB) connectivity, which allows for yet more nuanced insight into the simulated structures and therefore for more careful validation. The hydrogen-bond acceptor (A) and donor (D) interactions are counted for each individual water molecule in the simulation box. The nomenclature "AxDy"

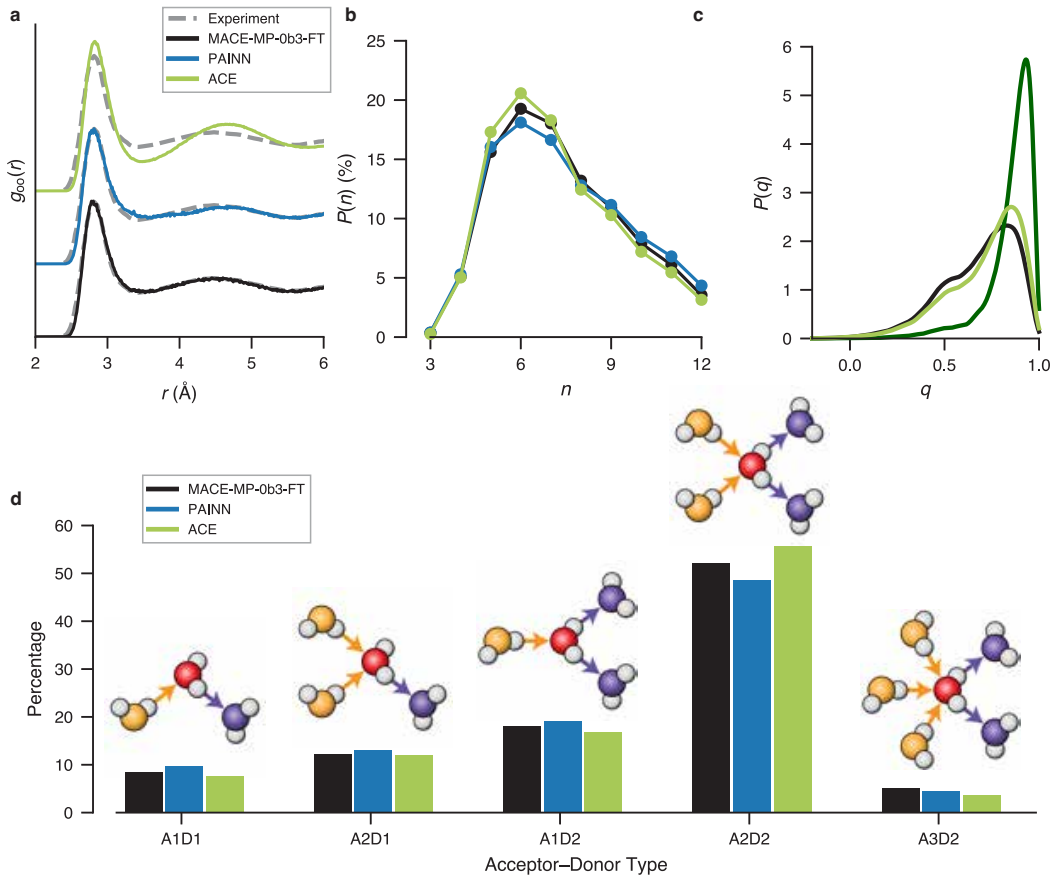

Figure 6.3: Physics-guided validation of distilled MLIPs. All analyses are derived from MD simulations of liquid water at 300 K. (**a**) Radial distribution function for O–O interactions. The distilled PaiNN model behaves similarly to the FM, which in turn agrees well with experimental data (taken from Ref. 250). (**b**) Distribution of n-membered rings in the simulated liquid. (**c**) Tetrahedral order parameter[251] distribution in each structural model. (**d**) Hydrogen-bonding network connectivity as presented in Ref. 252. We count different categories of topological connectivity, based on the number of H-bond acceptor (A) and donor (D) interactions. Figure reproduced with permission from Ref. 3.

refers to a water molecule with x acceptor interactions and y donor interactions.

In crystalline ice all water molecules have the "A2D2" HB topology, whereas liquid water contains many molecules with fewer HBs, and a few with more.[252] The descriptions of `MACE-MP-0b3` and its student models agree well across the range of topologies investigated; the PaiNN model shows slightly fewer crystal-like environments, in line with the less pronounced structuring evident from the RDF and density plots; conversely, the ACE model leads to a slightly more ordered (A2D2-type) water structure. However, both PaiNN and ACE predict fewer overly coordinated

water molecules ("A3D2").

To better understand the behaviour of the distilled ACE model, I performed a simulation at 340 K instead of 300 K and compared the resulting RDF to the 300 K simulation (Fig. 6.4). I observed a substantially improved match with experiment, and this result is consistent with the behaviour of a differently parameterised ACE model in Ref. 173, where increased temperatures of 305 K were required to prevent freezing for water-ice mixtures. Additionally, peaks are shifted to the right due to a reduced density as a result of the higher temperature causing the model to predict a less dense structure during the NPT equilibration process.

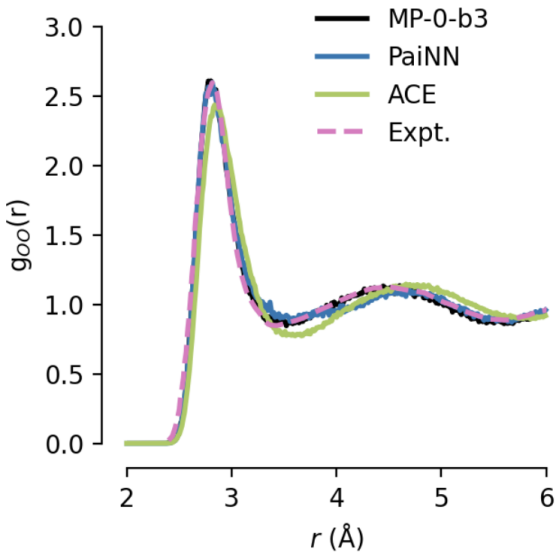

Figure 6.4: Effect of temperature for ACE RDF. RDF of the optimised ACE model from the 340 K simulation of water, compared to the 300 K simulations presented in Figure 6.3a. The reduced first peak, and improved match to experimental RDF are shown. Offsets in peaks are due to the higher temperature driving a less dense structure during the NPT equilibration phase.

6.4.2 Ablation studies

We perform ablation studies for key components and hyperparameters of the protocol: investigating the effects of altering the input database size and student models. Numerical experiments are used to determine the scaling laws for prediction errors

across student architectures. In doing so, we aim to better understand the limitations of the distillation.

The first experiment is to ablate (reduce) the number of synthetic structures used to distil the fine-tuned `MACE-MP-0b3` from the previous section (Fig. 6.5a). In this way, there is no change to the fine-tuned FM (and thus its predictions), but rather the amount of data used to distil the model, ensuring that the *quality* of the synthetic labels is not affected, but just their quantity and thus the coverage of the structure space. It is immediately clear that the ACE models do not achieve a comparable improvement to the other student models with increasing training set size. I posit that the hyperparameters being optimised for a training set size of 333 structures may constrain the accuracy by enforcing a high level of robustness (i.e., a reduced number of functions was found to be optimal c.f. Table 6.1). Further study is included in Section 6.4.3. However, to fully understand the origin and extend of the effect, further study is required.

In addition to the architectures mentioned above, ephemeral data-derived potential (EDDP) models are included in this series of experiments.[104] These are trained only using the energy labels of the synthetic dataset (due to practical limitations), and so are not directly comparable to the other models in terms of accuracy. For all architectures, increasing the amount of synthetic data improves final performance on the DFT-labelled test set, approaching, but not reaching, the accuracy of the FM in the large-data limit. The increase in amount of synthetic data comes with a time cost. For this specific combination of system and foundation model, generation of a new, uncorrelated structure occurs once every second.

These distilled models are trained to mimic the foundation model's outputs on the synthetic dataset: they have never been directly trained on any DFT labels. In Fig. 6.5b, the relationship between the distilled model's accuracy relative to the foundation model's labels (x-axis) and the DFT ground truth (y-axis) is shown. For reference, the naïve scaling law one would obtain if these two errors were uncorre-

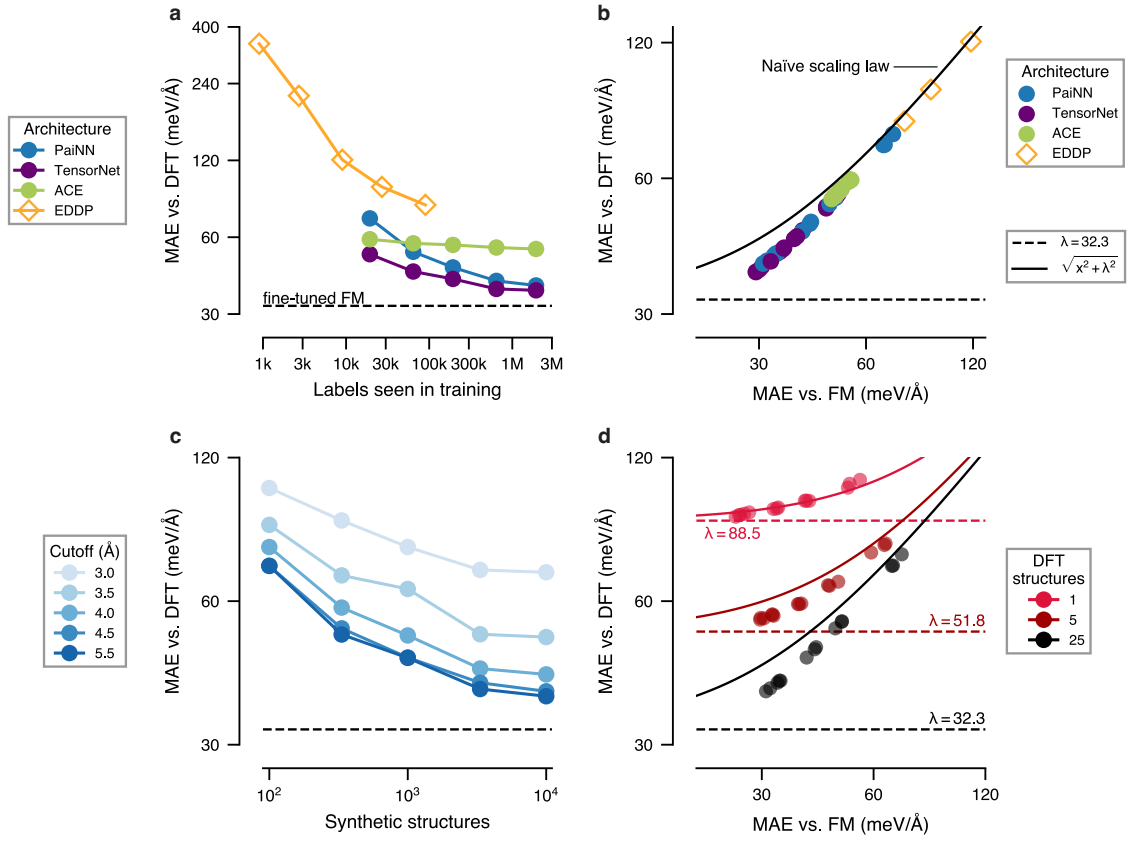

Figure 6.5: Ablation studies. All panels relate to the `MACE-MP-0b3` foundation model (FM) fine-tuned on water and distilled into various MLIP architectures (see Section 6.4.1). (**a**) Learning curves. The number of synthetic data labels used for distillation is varied: energy data for small cells in the case of EDDP training (*open diamonds*); energy and force data for larger cells in the case of all other models (*filled circles*). We plot the distilled models' component-wise force MAE on the DFT-labelled test set while varying the amount of synthetic data used in the distillation step. All points show the best-of-three, PaiNN and TensorNet models use a radial cut-off of 4.5 Å, while ACE and EDDP use 5.5Å. (**b**) Scaling laws. We plot the accuracy of each model relative to the fine-tuned FM (x-axis) and DFT (y-axis) on the hold-out test set. A solid line indicates the expected distribution if these two accuracies were uncorrelated; a dashed line shows the error of the FM against DFT (denoted λ). While the plot is truncated at 130 meV Å^{-1}, the remaining EDDP models behave similarly up to 400 meV Å^{-1}. (**c**) Cut-off radius. We take the same fine-tuned FM as in panel (a), and distil into the `PaiNN` architecture using various radial cut-offs. Again, all points show the best-of-three. (**d**) Amount of ground-truth domain data ('n-shot' performance). We vary the amount of data used to fine-tune `MACE-MP-0b3` (and hence its error against DFT), and plot the MAE of `PaiNN` models distilled from these fine-tuned FMs relative to the FM itself (x-axis) and to DFT (y-axis). Figure reproduced with permission from Ref. 3.

lated, *i.e.*, $y = \sqrt{x^2 + \lambda^2}$ is also plotted, where λ is the MAE of the FM on the DFT labels. For all architectures and amounts of synthetic data, but especially where errors are < 60 meV Å^{-1}, the distilled models are **more accurate** on the DFT than

would be expected. Indeed, there is a negative correlation between the quantities $F_{\mathrm{FM}} - F_{\mathrm{DFT}}$ and $F_{\mathrm{distilled}} - F_{\mathrm{FM}}$. Physically, this means that the distilled model (on average) under-predicts forces when the FM over-predicts them with respect to DFT. This can be thought of as a negative bias towards the true (DFT-labelled) PES, and it improves the data efficiency of synthetic distillation significantly. No claims are made about the origins of this phenomenon in this thesis, and further research is required to understand this behaviour and its origins. Further numerical testing across a range of systems, FM architectures, and distilled model architectures is required to fully characterise the uniformity and scale of this behaviour.

Studies of the cutoff (for PaiNN) and fine-tuning dataset size (for MACE-MP-0b3) are presented in Fig. 6.5c and d, justifying the cutoff of 4.5 Å chosen for the distilled model. Increasing the amount of fine-tuning data leads to more accurate foundation models, in turn providing more accurate distilled models. Fine-tuning offers the trade-off of requiring up to 100 times fewer datapoints, enabling higher-level DFT to be performed.[114]

6.4.3 ACE parameterisation

I now compare the performance of my XPOT-ACE optimised models to models fitted with the same parameterisation as from Ref. 173. The hyperparameters used by Ibrahim *et al.* are chosen to best fit their ACE model to their dataset, which comprises over 170,000 atomic environments, and due to the difference in nature of their dataset and the synthetic dataset generated here, there will be a difference in the optimal parameterisation. Additionally, the hyperparameters are chosen for a different level of DFT, which may cause discrepancies in the form of the PES which is modelled. However, the parameterisation in Ref. 173 is designed to describe liquid water and ice phases, and so should extend to the system contained here, and inform further the behaviour of ACE as an architecture vs. varying model shape.

In Fig. 6.6, I compare the performance of the XPOT-ACE optimised models to models fitted with the same parameterisation as from Ref. 173. Firstly, I observe

that the XPOT-optimised ACE models (henceforth, ACE) are more accurate than those from the parameterisation of Ref. 173 (Ibrahim-ACE) for the synthetic dataset sizes investigated. This is to be expected as the XPOT optimisation was performed upon a representative dataset of 333 structures from the synthetic dataset, thus providing a parameterisation which only has to describe liquid water (and at the level of theory here), and not be extendable to ordered and disordered ice phases.

Secondly, I observe the significant difference in data dependency based on the differences between the two parametrisations. The optimised ACE models are less dependent on increasing the amount of data used to train the model, or, looked at through another lens, they are less capable of leveraging additional data to improve their accuracy.

The formulation of the Ibrahim-ACE model comprises a larger number of basis functions (1774 vs. 1080), but a lower parameter count (4844 vs. 8048) than my ACE models as a consequence of the additional non-linear terms included (Equation 6.1). The maximum possible number of functions allowed by the optimisation was 2000, so it is not a case of user-enforced constraints which reduced the descriptor length for each atom, but rather that the optimisation found the best accuracy with a reduced number of basis functions.

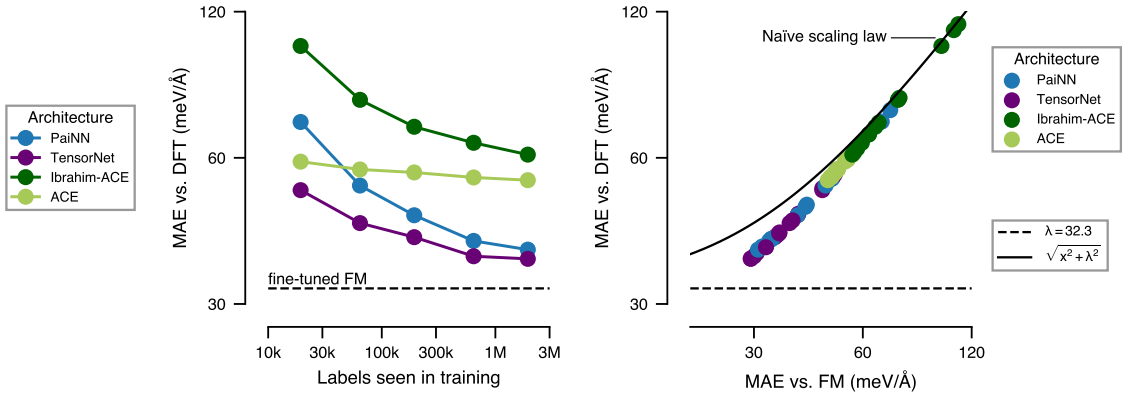

Figure 6.6: Study of the ACE parameterisation. A comparison of the XPOT-optimised ACE models (ACE) to those fitted on our synthetic dataset using the parameterisation of Ref. 173. The XPOT-optimised ACE models are more accurate than the Ibrahim-ACE models, and are less data-dependent. Figure drawn in the style of Figure 5 panels (**a**) and (**b**) in Ref. 3.

Based on the trend observed here, I expect that the Ibrahim-ACE models might achieve reduced numerical errors compared to my ACE models in the high data limit. However, it demonstrates that the ACE models fitted in this work are unable to compete with the accuracy of the GNN-based student models in the high data limit. Of course ACE models are more efficient and can be scaled to larger systems (Figure 6.2), but the discrepancy in accuracy is significant and points to the need for further research on whether there are specific dataset characteristics which can be leveraged to improve the accuracy of ACE models in comparison, or whether it is a fundamental limitation of the ACE architecture for this particular type of dataset.

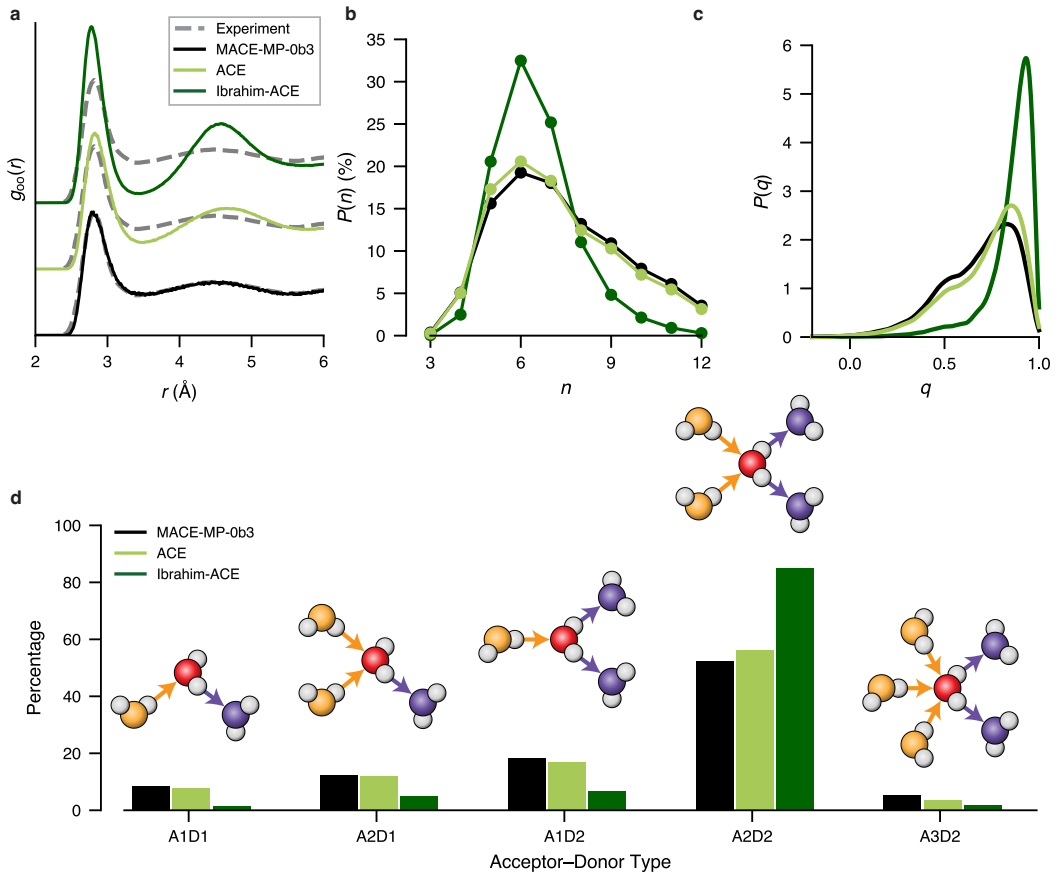

Figure 6.7: Comparison of ACE models. Plotted in the same style as Figure 6.3a, the XPOT-optimised ACE model (ACE) is compared to the Ibrahim-ACE model (Ibrahim-ACE) and the fine-tuned FM (MACE-MP-0b3). (**a**) RDF where the significant over-ordering of the Ibrahim-ACE model is shown. (**b**) Ring analysis, note the large overabundance of 6-membered rings. (**c**) Tetrahedral order parameter.[251] (**d**) Hydrogen-bonding network connectivity.[252] Figure drawn in the style of Figure 4 in Ref. 3.

Neither parametrisations of ACE are capable of driving stable MD simulations for over 1 ns until over 250,000 labels are included in the training set (1000 structures). While the error values between the distilled ACE do not suggest such a significant difference, the degradation in physical accuracy of structural prediction at 300 K is significant. Based on MSD analysis, the Ibrahim-ACE model predicts a frozen phase (with minimal MSD over 1.6 ns of simulation time) at 300 K, while the XPOT-optimised ACE model predicts a liquid where MSD is over 100 times higher, and comparable to the PaiNN model and FM under the same conditions.

The outcome of this "frozen" phase is shown in Figure 6.7. The RDF shows significant over-ordering of the structure predicted by the Ibrahim-ACE model, and this trend is visible in the ring analysis, tetrahedral order parameter, and hydrogen-bonding network connectivity. The difference between the structures is not slight, and it is not possible to say that the Ibrahim-ACE model has learned the correct behaviour from the synthetic dataset.

This result only highlights the importance of the choice of hyperparameters for a model. Although the system (water) is not changed from what the ACE model from Ibrahim *et al.* was trained on, the choice of hyperparameters for the ACE model from Ibrahim *et al.* do not provide a model which proves capable of learning the behaviour of the fine-tuned FM from the synthetic dataset. It must be noted that the DFT functional and level of theory for which the hyperparameters were chosen are different, but the dataset is preprocessed according to recommendations in Ref. 118, and the XPOT-optimised model retains a much more similar physical prediction to the fine-tuned FM. Indeed, I did not expect such a disparity in accuracy between the two models, especially given the reduced basis complexity of the XPOT-optimised ACE model, which would normally suggest (in the absence of overfitting) a less sensitive, and often accurate, model.

6.5 Outlook

In this chapter, I have presented a generalised approach for the architecture-agnostic distillation of atomistic foundation models to target specific chemical domains. This approach enables the creation of a "student" MLIP of any architecture that maintains the predictive power of their foundation model "teacher" with significantly reduced computational requirements: the distilled models can be substantially faster and less memory-intensive at inference time, enabling large-scale atomistic simulations. Although not discussed here, distilled models were trained in our work in Ref. 3 on a variety of chemical systems from liquid hydrogen to promising solar energy materials, demonstrating the generalisability and robustness of this approach.

The distilled models obtained with this approach are quick and cheap to produce: the only human input required is a small number of (< 50) DFT-labelled structures to fine-tune an existing FM to the desired level of accuracy on a given task. With the input structures provided, each stage of the process is fully automated, implemented in open-sourced code, and can leverage GPU acceleration. As such, the distillation process can take as little as 4 hours on a single Nvidia A6000 GPU. This process takes advantage of efficient implementation and the conceptual advantages of model distillation to provide cheaper access to accurate, robust, and efficient MLIPs. Given the substantial computational effort required for very-large-scale atomistic simulations, and the associated energy requirements and (indirectly) CO_2 emissions, providing access to cheaper models where possible is important. Just as with hyperparameter optimisation, we view distillation as key in reducing the overall cost of simulations going forward.

The results in this chapter already demonstrate the potential of knowledge distillation for atomistic MLIPs, and we expect that the approach will only become more useful as the field progresses. FM development continues at a breakneck pace, with the "state-of-the-art" FM changing regularly, and new architectures continue to be developed for efficient description of the potential energy surface. Distillation at

large, and our approach in particular, will benefit from these continued developments in these areas, owing to the architecture-agnostic nature of our findings. In the case of FMs, as the discrepancy between DFT and FM predictions decrease, the value of synthetic data will only increase, providing a better basis for distillation. From the perspective of student models, increasing efficiency will continue to drive larger and longer simulations to study complex chemical processes at an atomistic scale. In this way, distillation may be viewed as another major step — following foundational work showing accuracy and data scalability — in creating truly mainstream MLIPs for scientific research.

However, there are still questions to be answered about what makes a good student model architecture. Notably, the distilled ACE models here fail to live up to the GNN-based architectures in terms of physical accuracy. While this is to be expected due to the reduced cost of ACE models, the reduced data dependence of the optimised model in particular requires further study. I expect that distillation approaches will continue to be used with non-graph-based architectures, especially as the key tenet behind distillation is to provide a more efficient model that can be used in large-scale simulations. In particular, studies into the dependence of optimal hyperparameters on the number, and density (in descriptor space), of training structures will be key in understanding the promise of distillation for new and existing efficient model architectures.

Chapter 7

Conclusion and outlook

Over the four years during which I have worked on the research contained within this thesis, ML-driven materials modelling has boomed in popularity and scope. Throughout this period, one of the main limitations has continued to be the long, often iterative, process of fitting models to study individual material chemistries. To help address this, significant resource and effort has been expended to automate and streamline the process of fitting these models. However, the cost of including new chemistries such as dopants, impurities, or related compounds still remains high. Existing DFT-labelled datasets are often incompatible and are not re-used due to the lack of a single, defined standard for DFT functionals and settings used in data labelling, as well as the fact that datasets (as they are often created iteratively) are viewed to be "designed" for a specific model architecture.[199] As such, this diversity in label quality and data sampling results in little to no re-use of purpose-built datasets in the field.

In this thesis, I have fitted models to purpose-built datasets for Si, Te, Ge–Sb–Te, and $Li_{14}SiP_6$ to show that expertly-curated datasets are inherently transferable between model architectures, at a reduced cost compared to fitting from scratch, even if in some cases further modification may be required, and that hyperparameter optimisation enables this efficient transfer between model architectures. I have also shown that using Bayesian optimisation, it is possible improve the performance of a potential (where performance is defined as a combination of the accuracy, robustness, and inference cost) compared to existing models in literature.

Against the backdrop of improving purpose-specific models, foundation models have generated significant drive to make use of fine-tuning these models to perform pilot studies, cutting time frames from months to days. However, the inference cost of these models is still significant compared to the ACE models I have fitted and

discussed herein, leaving inaccessible the sort of long-timescale simulations that have been undertaken to study GST and Te.[4,5] Leveraging these models to reduce the cost of fitting smaller, faster models for specific chemistries, as explored in Chapter 6, is a promising avenue, and I look forward to further developments in this area. By using FMs to label data, astonishingly large datasets can be produced, and with this abundance of data, it is possible to fit models which can generate new chemical insights without thousands of DFT calculations, with reduced human effort.

Beyond the work herein, future study of hyperparameter values for fine-tuning, and improving the accuracy of these fine-tuned FMs will enable the field to efficiently study and describe materials processes, and XPOT (or other optimisation frameworks) will be key tools in this process. Further development of the optimisation strategies, loss function, and GNN-based model interfaces in XPOT will provide tools for the field to investigate model fitting from scratch, distillation, and fine-tuning. As model architectures and data availability improve, I believe further insights into best practices for leveraging their predictive power and robustness will only help us to provide new insight on processes to support, or explain, experimental findings.

More broadly, foundation models bring great promise, but also great risk. The benchmarks for such models often rely strongly on a numerical basis, which, while informative, does not provide a description of the quality and usefulness of these models for atomistic MD, which is likely to be their most common application. I am encouraged by the advances in the Matbench Discovery benchmark suite,[253] a project which aims to provide a leaderboard which continues to extend beyond solely numerical accuracy on testing labels, but includes further metrics of interest which enhance the description of any given FM's behaviour. Throughout my DPhil I have seen significant advances in the adaptability and usability of software developed for the computational study of materials, and I hope this trend continues. The democratisation of MLIPs for chemistry and materials science is critical in my mind

to ensuring that we accelerate materials discovery to reduce the footprint we create on the planet.

The ability to accurately describe the PES of disordered functional materials promises new insights which I hope will allow us to build better batteries to reduce our reliance on fossil fuels, design more efficient neuromorphic chips to reduce the energy consumption of AI, and develop theories on classes of materials to better determine which materials may hold promise. Efficient MLIPs, such as those I have fitted in this thesis, offer the possibility to study in depth the influence of local effects within disordered materials, as well as access length-scales capable of describing the processes which determine their efficacy. While new models will most likely be derived from fine-tuning (and distillation) of FMs, I believe that the research focused on the direct fitting of models to comprehensive datasets will only help to ensure that the validation and use of these new models continues to be robust.

For solid-state ion conductors, the ability to reliably fit accurate and efficient models capable of simulations which are long enough to enable study of Li-ion mobility in the ergodic regime is exciting for future work in the field for characterising new ion-conductors. While I was limited in temperature accessibility due to the non-convergence of the NGP in this work, the ever-increasing efficiency of MLIPs offers a path to true room temperature simulations of SSEs in the coming years.

Applying XPOT and my learnings to model fitting for Te and Ge–Sb–Te further crystallised the utility of hyperparameter optimisation in model development for complex systems. In less than 4 months, a GAP model was transferred into an equivalent ACE model which was over 400 times faster, and enabled billion-atom simulations of phase-changing memory devices, and the study of crystallisation at device-scale! I have great faith that the field will continue to grow, and that specifically the effects of dopants, direct simulation of interfaces, and eventual full-device simulation will enhance our understanding of PCMs in use in memory and neuromorphic computing hardware today.

MLIPs continue to prove their use as a structural analysis tool, delivering atomistic insight into the behaviour of materials. I envisage that the usage of MLIPs will continue to boom, and that they will become truly mainstream tools for scientific researchers from all backgrounds. As such, the development of the tools to enable easy fitting of models to new chemistries will be critical to the continued growth of the field.

Bibliography

[1] Thomas du Toit, D. F. & Deringer, V. L. Cross-platform hyperparameter optimization for machine learning interatomic potentials. *The Journal of Chemical Physics* **159**, 024803 (2023).

[2] Thomas du Toit, D. F., Zhou, Y. & Deringer, V. L. Hyperparameter Optimization for Atomic Cluster Expansion Potentials. *J. Chem. Theory Comput.* **20**, 10103–10113 (2024).

[3] Gardner, J. L. A. *et al.* Distillation of atomistic foundation models across architectures and chemical domains (2025). 2506.10956.

[4] Zhou, Y., du Toit, D. F. T., Elliott, S. R., Zhang, W. & Deringer, V. L. Full-cycle device-scale simulations of memory materials with a tailored atomic-cluster-expansion potential (2025). 2502.08393.

[5] Zhou, Y., Elliott, S. R., du Toit, D. F. T., Zhang, W. & Deringer, V. L. The pathway to chirality in elemental tellurium (2024). 2409.03860.

[6] Nicholas, T. C. *et al.* The structure and topology of an amorphous metal-organic framework (2025). 2503.24367.

[7] McKinsey and Company. The State of AI: Global survey. https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai.

[8] von Garrel, J. & Mayer, J. Artificial Intelligence in studies—use of ChatGPT and AI-based tools among students in Germany. *Humanit Soc Sci Commun* **10**, 799 (2023).

[9] Kalla, D., Smith, N., Samaah, F. & Kuraku, S. Study and Analysis of Chat GPT and its Impact on Different Fields of Study (2023). 4402499.

[10] DeepSeek-AI *et al.* DeepSeek-V3 Technical Report (2025). 2412.19437.

[11] Anthropics/claude-code. Anthropic (2025).

[12] Phan-Minh, T. *et al.* DriveIRL: Drive in Real Life with Inverse Reinforcement Learning. In *2023 IEEE Int. Conf. Robot. Autom. ICRA*, 1544–1550 (2023).

[13] Wu, T. *et al.* A Brief Overview of ChatGPT: The History, Status Quo and Potential Future Development. *IEEE/CAA JAS* **10**, 1122–1136 (2023).

[14] What drives progress in AI? Trends in Compute. https://futuretech.mit.edu/news/what-drives-progress-in-ai-trends-in-compute.

[15] Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

[16] Pyzer-Knapp, E. O. *et al.* Foundation models for materials discovery – current state and future directions. *npj Comput Mater* **11**, 1–10 (2025).

[17] Fu, X. *et al.* A foundation model of transcription across human cell types. *Nature* **637**, 965–973 (2025).

[18] Bodnar, C. *et al.* A foundation model for the Earth system. *Nature* **641**, 1180–1187 (2025).

[19] Parker, L. *et al.* AstroCLIP: A Cross-Modal Foundation Model for Galaxies. *Mon. Not. R. Astron. Soc.* **531**, 4990–5011 (2024). Comment: 18 pages, accepted in Monthly Notices of the Royal Astronomical Society, Presented at the NeurIPS 2023 AI4Science Workshop, 2310.03024.

[20] Hollmann, N. *et al.* Accurate predictions on small data with a tabular foundation model. *Nature* **637**, 319–326 (2025).

[21] Zhou, Y. *et al.* A foundation model for generalizable disease detection from retinal images. *Nature* **622**, 156–163 (2023).

[22] Hopfield, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci.* **79**, 2554–2558 (1982).

[23] Fahlman, S. E., Hinton, G. E. & Sejnowski, T. J. Massively parallel architectures for AI: Netl, thistle, and boltzmann machines. In *Proc. Third AAAI Conf. Artif. Intell.*, AAAI'83, 109–113 (AAAI Press, Washington, D.C., 1983).

[24] Ackley, D. H., Hinton, G. E. & Sejnowski, T. J. A learning algorithm for boltzmann machines. *Cognitive Science* **9**, 147–169 (1985).

[25] Kohn, W. Nobel Lecture: Electronic structure of matter—wave functions and density functionals. *Rev. Mod. Phys.* **71**, 1253–1266 (1999).

[26] Müser, M. H., Sukhomlinov, S. V. & Pastewka, L. Interatomic potentials: Achievements and challenges. *Adv. Phys. X* **8**, 2093129 (2023).

[27] Thiemann, F. L., O'Neill, N., Kapil, V., Michaelides, A. & Schran, C. Introduction to machine learning potentials for atomistic simulations. *J. Phys.: Condens. Matter* **37**, 073002 (2024).

[28] Fermi, E., Pasta, P., Ulam, S. & Tsingou, M. STUDIES OF THE NONLINEAR PROBLEMS. Tech. Rep. LA-1940, Los Alamos National Laboratory (LANL), Los Alamos, NM (United States) (1955).

[29] Rahman, A. Correlations in the Motion of Atoms in Liquid Argon. *Phys. Rev.* **136**, A405–A411 (1964).

[30] Jones, J. E. & Chapman, S. On the determination of molecular fields.—I. From the variation of the viscosity of a gas with temperature. *Proc. R. Soc. Lond. Ser. Contain. Pap. Math. Phys. Character* **106**, 441–462 (1997).

[31] Lennard-Jones, J. E. Cohesion. *Proc. Phys. Soc.* **43**, 461 (1931).

[32] Stillinger, F. H. & Weber, T. A. Computer simulation of local order in condensed phases of silicon. *Phys. Rev. B* **31**, 5262–5271 (1985).

[33] Daw, M. S. & Baskes, M. I. Semiempirical, Quantum Mechanical Calculation of Hydrogen Embrittlement in Metals. *Phys. Rev. Lett.* **50**, 1285–1288 (1983).

[34] Daw, M. S. & Baskes, M. I. Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. *Phys. Rev. B* **29**, 6443–6453 (1984).

[35] Kohn, W., Becke, A. D. & Parr, R. G. Density Functional Theory of Electronic Structure. *J. Phys. Chem.* **100**, 12974–12980 (1996).

[36] The Nobel Prize in Chemistry 1998. `https://www.nobelprize.org/prizes/chemistry/1998/summary/`.

[37] ARCHER2-HPC/usage-data. ARCHER2, UK National Supercomputing Service (2025).

[38] Agrawal, A. & Choudhary, A. Perspective: Materials informatics and big data: Realization of the "fourth paradigm" of science in materials science. *APL Materials* **4**, 053208 (2016).

[39] Caro, M. A., Deringer, V. L., Koskinen, J., Laurila, T. & Csányi, G. Growth Mechanism and Origin of High sp^3 Content in Tetrahedral Amorphous Carbon. *Phys. Rev. Lett.* **120**, 166101 (2018).

[40] Deringer, V. L. *et al.* Origins of structural and electronic transitions in disordered silicon. *Nature* **589**, 59–64 (2021).

[41] Zhou, Y., Elliott, S. R. & Deringer, V. L. Structure and Bonding in Amorphous Red Phosphorus. *Angew. Chem. Int. Ed.* **62**, e202216658 (2023).

[42] Rosset, L. A. M., Drabold, D. A. & Deringer, V. L. Signatures of paracrystallinity in amorphous silicon from machine-learning-driven molecular dynamics. *Nat Commun* **16**, 2360 (2025).

[43] Zhou, Y., Zhang, W., Ma, E. & Deringer, V. L. Device-scale atomistic modelling of phase-change memory materials. *Nat Electron* **6**, 746–754 (2023).

[44] Abou El Kheir, O. & Bernasconi, M. Million-Atom Simulation of the Set Process in Phase Change Memories at the Real Device Scale. *Adv. Electron. Mater.* **n/a**, e2500110.

[45] Jorn, R., Kumar, R., Abraham, D. P. & Voth, G. A. Atomistic Modeling of the Electrode–Electrolyte Interface in Li-Ion Energy Storage Systems: Electrolyte Structuring. *J. Phys. Chem. C* **117**, 3747–3761 (2013).

[46] Staacke, C. G., Huss, T., Margraf, J. T., Reuter, K. & Scheurer, C. Tackling Structural Complexity in Li2S-P2S5 Solid-State Electrolytes Using Machine Learning Potentials. *Nanomaterials (Basel)* **12**, 2950 (2022).

[47] Landgraf, V. *et al.* Disorder-Mediated Ionic Conductivity in Irreducible Solid Electrolytes. *J. Am. Chem. Soc.* (2025).

[48] Tusar, M., Zupan, J. & Gasteiger, J. Neural networks and modelling in chemistry. *J. Chim. Phys.* **89**, 1517–1529 (1992).

[49] Sumpter, B. G., Getino, C. & Noid, D. W. A neural network approach to the study of internal energy flow in molecular systems. *The Journal of Chemical Physics* **97**, 293–306 (1992).

[50] Skinner, A. J. & Broughton, J. Q. Neural networks in computational materials science: Training algorithms. *Modelling Simul. Mater. Sci. Eng.* **3**, 371 (1995).

[51] Behler, J. & Parrinello, M. Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Phys. Rev. Lett.* **98**, 146401 (2007).

[52] Bartók, A. P., Payne, M. C., Kondor, R. & Csányi, G. Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons. *Phys. Rev. Lett.* **104**, 136403 (2010).

[53] Sosso, G. C. *et al.* Fast Crystallization of the Phase Change Compound GeTe by Large-Scale Molecular Dynamics Simulations. *J. Phys. Chem. Lett.* **4**, 4241–4246 (2013).

[54] Deringer, V. L. & Csányi, G. Machine learning based interatomic potential for amorphous carbon. *Phys. Rev. B* **95**, 94203–94203 (2017).

[55] Bartók, A. P., Kermode, J., Bernstein, N. & Csányi, G. Machine Learning a General-Purpose Interatomic Potential for Silicon. *Phys. Rev. X* **8**, 041048 (2018).

[56] Willman, J. T. *et al.* Machine Learning Interatomic Potential for Simulations of Carbon at Extreme Conditions. *ArXiv220501209 Cond-Mat* (2022). 2205.01209.

[57] Guo, Z. *et al.* Extending the limit of molecular dynamics with ab initio accuracy to 10 billion atoms. In *Proc. 27th ACM SIGPLAN Symp. Princ. Pract. Parallel Program.*, PPoPP '22, 205–218 (Association for Computing Machinery, New York, NY, USA, 2022).

[58] Johansson, A. *et al.* Micron-scale heterogeneous catalysis with Bayesian force fields from first principles and active learning (2022). Comment: 10 pages, 9 figures, 2204.12573.

[59] Morrow, J. D. & Deringer, V. L. Indirect learning and physically guided validation of interatomic potential models. *J. Chem. Phys.* **157**, 104105 (2022).

[60] Bartók, A. P., Kondor, R. & Csányi, G. On representing chemical environments. *Phys. Rev. B* **87**, 184115 (2013).

[61] Drautz, R. Atomic cluster expansion for accurate and transferable interatomic potentials. *Phys. Rev. B* **99**, 014104 (2019).

[62] Batzner, S. *et al.* E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat Commun* **13**, 2453 (2022).

[63] Batatia, I., Kovács, D. P., Simm, G. N. C., Ortner, C. & Csányi, G. MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields (2022). 2206.07697.

[64] Gasteiger, J., Groß, J. & Günnemann, S. Directional Message Passing for Molecular Graphs (2022). Comment: Published as a conference paper at ICLR 2020. Author name changed from Johannes Klicpera to Johannes Gasteiger, 2003.03123.

[65] Geiger, M. & Smidt, T. E3nn: Euclidean Neural Networks (2022). Comment: draft, 2207.09453.

[66] Bochkarev, A., Lysogorskiy, Y. & Drautz, R. Graph Atomic Cluster Expansion for semilocal interactions beyond equivariant message passing (2024). 2311.16326.

[67] Batatia, I. *et al.* A foundation model for atomistic materials chemistry (2024). Comment: 119 pages, 63 figures, 37MB PDF, 2401.00096.

[68] Kovács, D. P. *et al.* MACE-OFF23: Transferable Machine Learning Force Fields for Organic Molecules (2023). 2312.15211.

[69] Neumann, M. *et al.* Orb: A Fast, Scalable Neural Network Potential (2024). 2410.22570.

[70] Rhodes, B. *et al.* Orb-v3: Atomistic simulation at scale (2025). Comment: 21 pages, 2504.06231.

[71] Barroso-Luque, L. *et al.* Open Materials 2024 (OMat24) Inorganic Materials Dataset and Models (2024). Comment: 19 pages, 2410.12771.

[72] Zubatyuk, R., Smith, J. S., Leszczynski, J. & Isayev, O. Accurate and transferable multitask prediction of chemical properties with an atoms-in-molecules neural network. *Sci. Adv.* **5**, eaav6490 (2019).

[73] Kovács, D. P. *et al.* MACE-OFF: Short-Range Transferable Machine Learning Force Fields for Organic Molecules. *J. Am. Chem. Soc.* **147**, 17598–17611 (2025).

[74] Shoghi, N. *et al.* From Molecules to Materials: Pre-training Large Generalizable Models for Atomic Property Prediction (2024). 2310.16802.

[75] Zhang, D. *et al.* DPA-2: A large atomic model as a multi-task learner. *npj Comput Mater* **10**, 1–15 (2024).

[76] Zhang, S. *et al.* Exploring the frontiers of condensed-phase chemistry with a general reactive machine learning potential. *Nat. Chem.* **16**, 727–734 (2024).

[77] Kaur, H. *et al.* Data-efficient fine-tuning of foundational models for first-principles quality sublimation enthalpies. *Faraday Discuss.* **256**, 120–138 (2025).

[78] Liao, Y.-L., Wood, B., Das, A. & Smidt, T. EquiformerV2: Improved Equivariant Transformer for Scaling to Higher-Degree Representations (2024). Comment: Published as a conference paper at ICLR 2024, 2306.12059.

[79] Fu, X. *et al.* Learning Smooth and Expressive Interatomic Potentials for Physical Property Prediction (2025). Comment: 20 pages, 14 figures, 6 tables, 2502.12147.

[80] Batatia, I. *et al.* The design space of E(3)-equivariant atom-centred interatomic potentials. *Nat Mach Intell* **7**, 56–67 (2025).

[81] Zuo, Y. *et al.* A Performance and Cost Assessment of Machine Learning Interatomic Potentials. *J. Phys. Chem. A* **124**, 731–745 (2020). 1906.08888.

[82] Póta, B., Ahlawat, P., Csányi, G. & Simoncelli, M. Thermal Conductivity Predictions with Foundation Atomistic Models (2024). Comment: 15 pages, 10 figures, 2408.00755.

[83] Riebesell, J. *et al.* Matbench Discovery – A framework to evaluate machine learning crystal stability predictions (2024). Comment: Please see online leaderboard at: https://matbench-discovery.materialsproject.org/, 2308.14920.

[84] Leimeroth, N., Erhard, L. C., Albe, K. & Rohrer, J. Machine-learning interatomic potentials from a users perspective: A comparison of accuracy, speed and data efficiency (2025). 2505.02503.

[85] Shiota, T., Ishihara, K., Do, T. M., Mori, T. & Mizukami, W. Taming Multi-Domain, -Fidelity Data: Towards Foundation Models for Atomistic Scale Simulations (2024). 2412.13088.

[86] Schmidt, J., Wang, H.-C., Cerqueira, T. F. T., Botti, S. & Marques, M. A. L. A dataset of 175k stable and metastable materials calculated with the PBEsol and SCAN functionals. *Sci Data* **9**, 64 (2022).

[87] Levine, D. S. *et al.* The Open Molecules 2025 (OMol25) Dataset, Evaluations, and Models (2025). Comment: 60 pages, 8 figures, 2505.08762.

[88] van der Oord, C., Sachs, M., Kovács, D. P., Ortner, C. & Csányi, G. Hyperactive learning for data-driven interatomic potentials. *npj Comput Mater* **9**, 1–14 (2023).

[89] Bernstein, N., Csányi, G. & Deringer, V. L. De novo exploration and self-guided learning of potential-energy surfaces. *npj Comput Mater* **5**, 1–9 (2019).

[90] Podryabinkin, E. V. & Shapeev, A. V. Active learning of linearly parametrized interatomic potentials. *Computational Materials Science* **140**, 171–180 (2017).

[91] Jinnouchi, R., Miwa, K., Karsai, F., Kresse, G. & Asahi, R. On-the-Fly Active Learning of Interatomic Potentials for Large-Scale Atomistic Simulations. *J. Phys. Chem. Lett.* **11**, 6946–6955 (2020).

[92] Stenczel, T. K. *et al.* Machine-learned acceleration for molecular dynamics in CASTEP. *The Journal of Chemical Physics* **159**, 044803 (2023).

[93] Menon, S. *et al.* From electrons to phase diagrams with machine learning potentials using pyiron based automated workflows (2024).

[94] Liu, Y. *et al.* An automated framework for exploring and learning potential-energy surfaces (2024). 2412.16736.

[95] Fiedler, L. *et al.* Training-free hyperparameter optimization of neural networks for electronic structures in matter. *Mach. Learn.: Sci. Technol.* **3**, 045008 (2022).

[96] Poelking, C., Faber, F. A. & Cheng, B. BenchML: An extensible pipelining framework for benchmarking representations of materials and molecules at scale. *Mach. Learn.: Sci. Technol.* **3**, 040501 (2022).

[97] Chen, C. *et al.* Accurate force field for molybdenum by machine learning large materials data. *Phys. Rev. Materials* **1**, 043603 (2017).

[98] Erhard, L. C., Rohrer, J., Albe, K. & Deringer, V. L. Modelling atomic and nanoscale structure in the silicon–oxygen system through active machine learning. *Nat Commun* **15**, 1927 (2024).

[99] Kirsz, M., Daramola, A., Hermann, A., Zong, H. & Ackland, G. J. Tadah! A Swiss Army Knife for Developing and Deployment of Machine Learning Interatomic Potentials (2025). 2502.02211.

[100] Deringer, V. L. Modelling and understanding battery materials with machine-learning-driven atomistic simulations. *J. Phys. Energy* **2**, 041003 (2020).

[101] Deringer, V. L. *et al.* Gaussian Process Regression for Materials and Molecules. *Chem. Rev.* **121**, 10073–10141 (2021).

[102] Morrow, J. D., Gardner, J. L. A. & Deringer, V. L. How to validate machine-learned interatomic potentials. *The Journal of Chemical Physics* **158**, 121501 (2023).

[103] Work With New Electronic "Brains" Opens Field for Army Math Experts. *The Times* 65 (1957).

[104] Pickard, C. J. Ephemeral data derived potentials for random structure search. *Phys. Rev. B* **106**, 014102 (2022).

[105] Deringer, V. L., Caro, M. A. & Csányi, G. A general-purpose machine-learning force field for bulk and nanostructured phosphorus. *Nat Commun* **11**, 5461 (2020).

[106] El-Machachi, Z., Wilson, M. & Deringer, V. L. Exploring the configurational space of amorphous graphene with machine-learned atomic energies. *Chem. Sci.* **13**, 13720–13731 (2022).

[107] Vandermause, J. *et al.* On-the-fly active learning of interpretable Bayesian force fields for atomistic rare events. *npj Comput Mater* **6**, 1–11 (2020).

[108] Zhang, L., Lin, D.-Y., Wang, H., Car, R. & E, W. Active learning of uniformly accurate interatomic potentials for materials simulation. *Phys. Rev. Mater.* **3**, 023804 (2019).

[109] Kulichenko, M. *et al.* Uncertainty-driven dynamics for active learning of interatomic potentials. *Nat Comput Sci* **3**, 230–239 (2023).

[110] Ashukha, A., Lyzhov, A., Molchanov, D. & Vetrov, D. Pitfalls of In-Domain Uncertainty Estimation and Ensembling in Deep Learning (2021). 2002.06470.

[111] Grasselli, F., Kapil, V., Bonfanti, S., Rossi, K. & Chong, S. Uncertainty in the era of machine learning for atomistic modeling. *Digital Discovery* (2025).

[112] Lysogorskiy, Y., Bochkarev, A., Mrovec, M. & Drautz, R. Active learning strategies for atomic cluster expansion models. *Phys. Rev. Mater.* **7**, 043801 (2023).

[113] Vandermause, J., Xie, Y., Lim, J. S., Owen, C. J. & Kozinsky, B. Active learning of reactive Bayesian force fields applied to heterogeneous catalysis dynamics of H/Pt. *Nat Commun* **13**, 5183 (2022).

[114] Gardner, J. L. A., Beaulieu, Z. F. & Deringer, V. L. Synthetic data enable experiments in atomistic machine learning. *Digital Discovery* **2**, 651–662 (2023).

[115] Blank, T. B., Brown, S. D., Calhoun, A. W. & Doren, D. J. Neural network models of potential energy surfaces. *The Journal of Chemical Physics* **103**, 4129–4137 (1995).

[116] Gardner, J. L. A., Baker, K. T. & Deringer, V. L. Synthetic pre-training for neural-network interatomic potentials. *Mach. Learn.: Sci. Technol.* **5**, 015003 (2024).

[117] Morrow, J. D. *et al.* Understanding Defects in Amorphous Silicon with Million-Atom Simulations and Machine Learning. *Angew. Chem.* **136**, e202403842 (2024).

[118] Bochkarev, A. *et al.* Efficient parametrization of the atomic cluster expansion. *Phys. Rev. Mater.* **6**, 013804 (2022).

[119] Thompson, A. P., Swiler, L. P., Trott, C. R., Foiles, S. M. & Tucker, G. J. Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials. *Journal of Computational Physics* **285**, 316–330 (2015).

[120] De, S., Bartók, A. P., Csányi, G. & Ceriotti, M. Comparing molecules and solids across structural and alchemical space. *Phys. Chem. Chem. Phys.* **18**, 13754–13769 (2016).

[121] Zhou, Y., Zhang, W., Ma, E. & Deringer, V. L. Unlocking device-scale atomistic modelling of phase-change memory materials (2022). 2207.14228.

[122] Darby, J. P., Kermode, J. R. & Csányi, G. Compressing local atomic neighbourhood descriptors. *npj Comput Mater* **8**, 1–13 (2022).

[123] Drautz, R. & Fähnle, M. Spin-cluster expansion: Parametrization of the general adiabatic magnetic energy surface with ab initio accuracy. *Phys. Rev. B* **69**, 104404 (2004).

[124] Musil, F. *et al.* Physics-Inspired Structural Representations for Molecules and Materials. *Chem. Rev.* **121**, 9759–9815 (2021).

[125] Kovács, D. P. *et al.* Linear Atomic Cluster Expansion Force Fields for Organic Molecules: Beyond RMSE. *J. Chem. Theory Comput.* **17**, 7696–7711 (2021).

[126] Shapeev, A. V. Moment Tensor Potentials: A Class of Systematically Improvable Interatomic Potentials. *Multiscale Model. Simul.* **14**, 1153–1173 (2016).

[127] Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A. & Müller, K.-R. SchNet – A deep learning architecture for molecules and materials. *The Journal of Chemical Physics* **148**, 241722 (2018).

[128] Zhang, L., Han, J., Wang, H., Car, R. & E, W. Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics. *Phys. Rev. Lett.* **120**, 143001 (2018).

[129] Batatia, I., Kovacs, D. P., Simm, G., Ortner, C. & Csanyi, G. MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields. In Koyejo, S. *et al.* (eds.) *Adv. Neural Inf. Process. Syst.*, vol. 35, 11423–11436 (Curran Associates, Inc., 2022).

[130] Wood, M. A. & Thompson, A. P. Extending the accuracy of the SNAP interatomic potential form. *The Journal of Chemical Physics* **148**, 241721 (2018).

[131] Bartók, A. P. Gaussian Approximation Potential: An interatomic potential derived from first principles Quantum Mechanics (2010). Comment: PhD thesis, University of Cambridge 2009, 1003.2817.

[132] Finnis, M. W. & Sinclair, J. E. A simple empirical N-body potential for transition metals. *Philos. Mag. A* **50**, 45–55 (1984).

[133] Lysogorskiy, Y. *et al.* Performant implementation of the atomic cluster expansion (PACE) and application to copper and silicon. *npj Comput Mater* **7**, 1–12 (2021).

[134] Witt, W. C. *et al.* ACEpotentials.jl: A Julia implementation of the atomic cluster expansion. *The Journal of Chemical Physics* **159**, 164101 (2023).

[135] Rohskopf, A. *et al.* FitSNAP: Atomistic machine learning with LAMMPS. *J. Open Source Softw.* **8**, 5118 (2023).

[136] Zou, H. & Hastie, T. Regularization and Variable Selection Via the Elastic Net. *Journal of the Royal Statistical Society Series B: Statistical Methodology* **67**, 301–320 (2005).

[137] Hvarfner, C., Hellsten, E. O. & Nardi, L. Vanilla Bayesian Optimization Performs Great in High Dimensions (2024). 2402.02229.

[138] Jones, D. R., Schonlau, M. & Welch, W. J. Efficient Global Optimization of Expensive Black-Box Functions. *Journal of Global Optimization* **13**, 455–492 (1998).

[139] Head, T. *et al.* Scikit-optimize/scikit-optimize: V0.5.2. Zenodo (2018).

[140] Behler, J. First Principles Neural Network Potentials for Reactive Simulations of Large Molecular and Condensed Systems. *Angew. Chem. Int. Ed.* **56**, 12828–12840 (2017).

[141] Deringer, V. L., Caro, M. A. & Csányi, G. Machine Learning Interatomic Potentials as Emerging Tools for Materials Science. *Adv. Mater.* **31**, 1902765 (2019).

[142] Noé, F., Tkatchenko, A., Müller, K.-R. & Clementi, C. Machine Learning for Molecular Simulation. *Annu. Rev. Phys. Chem.* **71**, 361–390 (2020).

[143] Unke, O. T. *et al.* Machine Learning Force Fields. *Chem. Rev.* **121**, 10142–10186 (2021).

[144] Friederich, P., Häse, F., Proppe, J. & Aspuru-Guzik, A. Machine-learned potentials for next-generation matter simulations. *Nat. Mater.* **20**, 750–761 (2021).

[145] Qamar, M., Mrovec, M., Lysogorskiy, Y., Bochkarev, A. & Drautz, R. Atomic Cluster Expansion for Quantum-Accurate Large-Scale Simulations of Carbon. *J. Chem. Theory Comput.* **19**, 5151–5167 (2023).

[146] Ouyang, Y. *et al.* Accurate description of high-order phonon anharmonicity and lattice thermal conductivity from molecular dynamics simulations with machine learning potential. *Phys. Rev. B* **105**, 115202 (2022).

[147] Rinaldi, M., Mrovec, M., Bochkarev, A., Lysogorskiy, Y. & Drautz, R. Non-collinear magnetic atomic cluster expansion for iron. *npj Comput Mater* **10**, 1–12 (2024).

[148] Li, F. *et al.* Stable All-Solid-State Lithium Metal Batteries Enabled by Machine Learning Simulation Designed Halide Electrolytes. *Nano Lett.* **22**, 2461–2469 (2022).

[149] Klawohn, S. *et al.* Gaussian approximation potentials: Theory, software implementation and application examples. *The Journal of Chemical Physics* **159**, 174108 (2023).

[150] Chen, C. & Ong, S. P. A universal graph deep learning interatomic potential for the periodic table. *Nat Comput Sci* **2**, 718–728 (2022).

[151] Miwa, K. & Ohno, H. Interatomic potential construction with self-learning and adaptive database. *Phys. Rev. Mater.* **1**, 053801 (2017).

[152] Menon, S. *et al.* From electrons to phase diagrams with machine learning potentials using pyiron based automated workflows. *npj Comput Mater* **10**, 261 (2024).

[153] Blum, L. C. & Reymond, J.-L. 970 Million Druglike Small Molecules for Virtual Screening in the Chemical Universe Database GDB-13. *J. Am. Chem. Soc.* **131**, 8732–8733 (2009).

[154] Rupp, M., Tkatchenko, A., Müller, K.-R. & von Lilienfeld, O. A. Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning. *Phys. Rev. Lett.* **108**, 058301 (2012).

[155] Csanyi, G. *et al.* Expressive Programming for Computational Physics in Fortran 950+. *NCP* 1–24 (2007).

[156] Huber, P. J. Robust Estimation of a Location Parameter. *Ann. Math. Stat.* **35**, 73–101 (1964).

[157] Abu-Mostafa, Y. S., Magdon-Ismail, M. & Lin, H.-T. *Learning from Data: A Short Course* (AMLbook.com, S.l., 2012).

[158] Hoffmann, R., Kabanov, A. A., Golov, A. A. & Proserpio, D. M. Homo Citans and Carbon Allotropes: For an Ethics of Citation. *Angew. Chem. Int. Ed.* **55**, 10962–10976 (2016).

[159] Jain, A. *et al.* The Materials Project: A materials genome approach to accelerating materials innovation. *APL Mater.* **1**, 011002 (2013).

[160] Frazier, P. I. A Tutorial on Bayesian Optimization (2018). 1807.02811.

[161] Wong, T.-T., , L., Wai-Shing & and Heng, P.-A. Sampling with Hammersley and Halton Points. *J. Graph. Tools* **2**, 9–24 (1997).

[162] Stuke, A., Rinke, P. & Todorović, M. Efficient hyperparameter tuning for kernel ridge regression with Bayesian optimization. *Mach. Learn.: Sci. Technol.* **2**, 035022 (2021).

[163] Park, H. *et al.* End-to-end AI framework for interpretable prediction of molecular and crystal properties. *Mach. Learn.: Sci. Technol.* **4**, 025036 (2023).

[164] Rowe, P., Deringer, V. L., Gasparotto, P., Csányi, G. & Michaelides, A. An accurate and transferable machine learning potential for carbon. *J. Chem. Phys.* **153**, 034702 (2020).

[165] Muhli, H. *et al.* Machine learning force fields based on local parametrization of dispersion interactions: Application to the phase diagram of ${\mathrm{C}}_{60}$. *Phys. Rev. B* **104**, 054106 (2021).

[166] Deringer, V. L. *et al.* Realistic Atomistic Structure of Amorphous Silicon from Machine-Learning-Driven Molecular Dynamics. *J. Phys. Chem. Lett.* **9**, 2879–2885 (2018).

[167] de Tomas, C. *et al.* Transferability in interatomic potentials for carbon. *Carbon* **155**, 624–634 (2019).

[168] Strangmüller, S. *et al.* Fast Ionic Conductivity in the Most Lithium-Rich Phosphidosilicate Li14SiP6. *J. Am. Chem. Soc.* **141**, 14200–14209 (2019).

[169] Zhou, Y., Kirkpatrick, W. & Deringer, V. L. Cluster Fragments in Amorphous Phosphorus and their Evolution under Pressure. *Adv. Mater.* **34**, 2107515 (2022).

[170] George, J., Hautier, G., Bartók, A. P., Csányi, G. & Deringer, V. L. Combining phonon accuracy with high transferability in Gaussian approximation potential models. *The Journal of Chemical Physics* **153**, 044104 (2020).

[171] McInnes, L., Healy, J., Saul, N. & Großberger, L. UMAP: Uniform Manifold Approximation and Projection. *J. Open Source Softw.* **3**, 861 (2018).

[172] Training — mace 0.1.0 documentation. https://mace-docs.readthedocs.io/en/latest/guide/training.html#swa-and-ema.

[173] Ibrahim, E., Lysogorskiy, Y. & Drautz, R. Efficient Parametrization of Transferable Atomic Cluster Expansion for Water. *J. Chem. Theory Comput.* **20**, 11049–11057 (2024).

[174] Erhard, L. C., Rohrer, J., Albe, K. & Deringer, V. L. A machine-learned interatomic potential for silica and its relation to empirical models. *npj Comput Mater* **8**, 1–12 (2022).

[175] Laaziri, K. *et al.* High-energy x-ray diffraction study of pure amorphous silicon. *Phys. Rev. B* **60**, 13520–13533 (1999).

[176] Xie, R. *et al.* Hyperuniformity in amorphous silicon based on the measurement of the infinite-wavelength limit of the structure factor. *Proc. Natl. Acad. Sci.* **110**, 13250–13254 (2013).

[177] Pandey, K. K., Garg, N., Shanavas, K. V., Sharma, S. M. & Sikka, S. K. Pressure induced crystallization in amorphous silicon. *Journal of Applied Physics* **109**, 113511 (2011).

[178] Thompson, A. P. *et al.* LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Physics Communications* **271**, 108171 (2022).

[179] McMillan, P. F., Wilson, M., Daisenberger, D. & Machon, D. A density-driven phase transition between semiconducting and metallic polyamorphs of silicon. *Nature Mater* **4**, 680–684 (2005).

[180] Rao, F. *et al.* Reducing the stochasticity of crystal nucleation to enable subnanosecond memory writing. *Science* **358**, 1423–1427 (2017).

[181] Ding, K. *et al.* Phase-change heterostructure enables ultralow noise and drift for memory operation. *Science* **366**, 210–215 (2019).

[182] Sosso, G. C., Miceli, G., Caravati, S., Behler, J. & Bernasconi, M. Neural network interatomic potential for the phase change material GeTe. *Phys. Rev. B* **85**, 174103 (2012).

[183] Mocanu, F. C. *et al.* Modeling the Phase-Change Memory Material, Ge2Sb2Te5, with a Machine-Learned Interatomic Potential. *J. Phys. Chem. B* **122**, 8998–9006 (2018).

[184] Slater, M. D., Kim, D., Lee, E. & Johnson, C. S. Sodium-Ion Batteries. *Adv. Funct. Mater.* **23**, 947–958 (2013).

[185] Sosso, G. C., Donadio, D., Caravati, S., Behler, J. & Bernasconi, M. Thermal transport in phase-change materials from atomistic simulations. *Phys. Rev. B* **86**, 104301 (2012).

[186] Konstantinou, K., Mocanu, F. C., Lee, T.-H. & Elliott, S. R. Revealing the intrinsic nature of the mid-gap defects in amorphous Ge2Sb2Te5. *Nat Commun* **10**, 3065 (2019).

[187] Konstantinou, K., Mavračić, J., Mocanu, F. C. & Elliott, S. R. Simulation of Phase-Change-Memory and Thermoelectric Materials using Machine-Learned Interatomic Potentials: Sb2Te3. *Phys. Status Solidi B* **258**, 2000416 (2021).

[188] Xu, Y. *et al.* Unraveling Crystallization Mechanisms and Electronic Structure of Phase-Change Materials by Large-Scale Ab Initio Simulations. *Adv. Mater.* **34**, 2109139 (2022).

[189] Zhang, B. *et al.* Element-resolved atomic structure imaging of rocksalt Ge2Sb2Te5 phase-change material. *Applied Physics Letters* **108**, 191902 (2016).

[190] Wang, J. *et al.* Enhancing thermoelectric performance of Sb2Te3 through swapped bilayer defects. *Nano Energy* **79**, 105484 (2021).

[191] Caravati, S., Bernasconi, M., Kühne, T. D., Krack, M. & Parrinello, M. Coexistence of tetrahedral- and octahedral-like sites in amorphous phase change materials. *Appl. Phys. Lett.* **91**, 171906–3 (2007).

[192] Raty, J. Y. *et al.* Aging mechanisms in amorphous phase-change materials. *Nat Commun* **6**, 7467 (2015).

[193] Jang, G. *et al.* Bidirectional-nonlinear threshold switching behaviors and thermally robust stability of ZnTe selectors by nitrogen annealing. *Sci Rep* **10**, 16286 (2020).

[194] Shen, J. *et al.* Elemental electrical switch enabling phase segregation–free operation. *Science* **374**, 1390–1394 (2021).

[195] Zhu, M., Ren, K. & Song, Z. Ovonic threshold switching selectors for three-dimensional stackable phase-change memory. *MRS Bulletin* **44**, 715–720 (2019).

[196] Cheng, H.-Y., Carta, F., Chien, W.-C., Lung, H.-L. & BrightSky, M. J. 3D cross-point phase-change memory for storage-class memory. *J. Phys. D: Appl. Phys.* **52**, 473002 (2019).

[197] Ben Mahmoud, C., Gardner, J. L. A. & Deringer, V. L. Data as the next challenge in atomistic machine learning. *Nat Comput Sci* 1–4 (2024).

[198] Kulichenko, M. *et al.* Data Generation for Machine Learning Interatomic Potentials and Beyond. *Chem. Rev.* **124**, 13681–13714 (2024).

[199] Niblett, S. P., Kourtis, P., Magdău, I.-B., Grey, C. P. & Csányi, G. Transferability of Data Sets between Machine-Learned Interatomic Potential Algorithms. *J. Chem. Theory Comput.* (2025).

[200] Musaelian, A. *et al.* Learning local equivariant representations for large-scale atomistic dynamics. *Nat Commun* **14**, 579 (2023).

[201] Poul, M., Huber, L. & Neugebauer, J. Automated generation of structure datasets for machine learning potentials and alloys. *npj Comput Mater* **11**, 174 (2025).

[202] Keen, D. A. & Goodwin, A. L. The crystallography of correlated disorder. *Nature* **521**, 303–309 (2015).

[203] Simonov, A. & Goodwin, A. L. Designing disorder into crystalline materials. *Nature Reviews Chemistry* **4**, 657–673 (2020).

[204] Morgan, D. & Jacobs, R. Opportunities and Challenges for Machine Learning in Materials Science. *Annu. Rev. Mater. Res.* **50**, 71–103 (2020).

[205] Morgan, B. J. Mechanistic Origin of Superionic Lithium Diffusion in Anion-Disordered Li6PS5X Argyrodites. *Chem. Mater.* **33**, 2004–2018 (2021).

[206] Szczuka, C. *et al.* Forced Disorder in the Solid Solution Li3P–Li2S: A New Class of Fully Reduced Solid Electrolytes for Lithium Metal Anodes. *J. Am. Chem. Soc.* **144**, 16350–16365 (2022).

[207] Liu, H. *et al.* A disordered rock salt anode for fast-charging lithium-ion batteries. *Nature* **585**, 63–67 (2020).

[208] Lee, J. *et al.* Unlocking the Potential of Cation-Disordered Oxides for Rechargeable Lithium Batteries. *Science* **343**, 519–522 (2014).

[209] Clément, R. J., Lun, Z. & Ceder, G. Cation-disordered rocksalt transition metal oxides and oxyfluorides for high energy lithium-ion cathodes. *Energy Environ. Sci.* **13**, 345–373 (2020).

[210] Brauer, G. & Zintl, E. Konstitution von Phosphiden, Arseniden, Antimoniden und Wismutiden des Lithiums, Natriums und Kaliums. (23: Mitteilung ueber Metalle und Legierungen.). *Z. Fuer Phys. Chem. Abt. B Chem. Elem. Aufbau Mater.* **37**, 323–352 (1937).

[211] Jiang, J. *et al.* Scandium Induced Structural Disorder and Vacancy Engineering in Li3Sb – Superior Ionic Conductivity in Li3-3xScxSb. *Adv. Energy Mater.* **n/a**, 2500683.

[212] Strangmüller, S. *et al.* Modifying the Properties of Fast Lithium-Ion Conductors—The Lithium Phosphidotetrelates Li14SiP6, Li14GeP6, and Li14SnP6. *Chem. Mater.* **32**, 6925–6934 (2020).

[213] Restle, T. M. F. *et al.* Fast Lithium Ion Conduction in Lithium Phosphidoaluminates. *Angew. Chem. Int. Ed.* **59**, 5665–5674 (2020).

[214] Restle, T. M. F. *et al.* Fast Lithium-Ion Conduction in Phosphide Li9GaP4. *Chem. Mater.* **33**, 2957–2966 (2021).

[215] Artrith, N. Machine learning for the modeling of interfaces in energy storage and conversion materials. *J. Phys. Energy* **1**, 032002 (2019).

[216] Zeilinger, M., Baran, V., van Wuellen, L., Haeussermann, U. & Faessler, T. F. Stabilizing the phase Li15 Si4 through lithium-aluminum substitution in Li15-x Alx Si4 ($0.4 < x < 0.8$)-single crystal X-ray structure determination of Li15 Si4 and Li14.37 Al0.63 Si4. *Chem. Mater.* **25**, 4113–4121 (2013).

[217] Perdew, J. P. *et al.* Restoring the density-gradient expansion for exchange in solids and surfaces. *Phys. Rev. Lett.* **100**, 136406 (2008). Comment: 4 pages, 2 figures, `0711.0156`.

[218] Clark, S. J. *et al.* First principles methods using CASTEP. *Z. Für Krist. - Cryst. Mater.* **220**, 567–570 (2005).

[219] El-Machachi, Z. *et al.* Accelerated First-Principles Exploration of Structure and Reactivity in Graphene Oxide. *Angew. Chem. Int. Ed.* **63**, e202410088 (2024).

[220] Kahle, L., Musaelian, A., Marzari, N. & Kozinsky, B. Unsupervised landmark analysis for jump detection in molecular dynamics simulations. *Phys. Rev. Mater.* **3**, 055404 (2019).

[221] Morgan, B. Bjmorgan/site-analysis (2025).

[222] Maginn, E. J., Messerly, R. A., Carlson, D. J., Roe, D. R. & Elliot, J. R. Best Practices for Computing Transport Properties 1. Self-Diffusivity and Viscosity from Equilibrium Molecular Dynamics [Article v1.0]. *Living J. Comput. Mol. Sci.* **1**, 6324–6324 (2019).

[223] Ernzerhof, M. & Scuseria, G. E. Assessment of the Perdew–Burke–Ernzerhof exchange-correlation functional. *The Journal of Chemical Physics* **110**, 5029–5036 (1999).

[224] Song, S. *et al.* Transport dynamics of complex fluids. *Proc. Natl. Acad. Sci.* **116**, 12733–12742 (2019).

[225] McCluskey, A. R., Coles, S. W. & Morgan, B. J. Accurate Estimation of Diffusion Coefficients and their Uncertainties from Computer Simulation. *J. Chem. Theory Comput.* **21**, 79–87 (2025).

[226] Wankmiller, B. & Hansen, M. R. Observation of Li+ jumps in solid inorganic electrolytes over a broad dynamical range: A case study of the lithium phosphidosilicates Li8SiP4 and Li14SiP6. *Journal of Magnetic Resonance Open* **14–15**, 100098 (2023).

[227] Wu, L. & Deringer, V. L. The Zintl-Klemm Concept in the Amorphous State: A Case Study of Na-P Battery Anodes. https://arxiv.org/abs/2504.04920v1 (2025).

[228] L. Deringer, V., M. Proserpio, D., Csányi, G. & J. Pickard, C. Data-driven learning and prediction of inorganic crystal structures. *Faraday Discuss.* **211**, 45–59 (2018).

[229] Bommasani, R. *et al.* On the Opportunities and Risks of Foundation Models (2022). Comment: Authored by the Center for Research on Foundation Models (CRFM) at the Stanford Institute for Human-Centered Artificial Intelligence (HAI). Report page with citation guidelines: https://crfm.stanford.edu/report.html, 2108.07258.

[230] Brown, T. *et al.* Language Models are Few-Shot Learners. In *Adv. Neural Inf. Process. Syst.*, vol. 33, 1877–1901 (Curran Associates, Inc., 2020).

[231] Kirillov, A. *et al.* Segment Anything (2023). Comment: Project web-page: https://segment-anything.com, 2304.02643.

[232] Yang, H. *et al.* MatterSim: A Deep Learning Atomistic Model Across Elements, Temperatures and Pressures (2024). 2405.04967.

[233] Smith, J. S., Nebgen, B., Lubbers, N., Isayev, O. & Roitberg, A. E. Less is more: Sampling chemical space with active learning. *The Journal of Chemical Physics* **148**, 241733 (2018).

[234] Kaplan, A. D. *et al.* A Foundational Potential Energy Surface Dataset for Materials (2025). Comment: The first three listed authors contributed equally to this work. For training data, see http://matpes.ai or https://materialsproject-contribs.s3.amazonaws.com/index.html#MatPES_2025_1/, 2503.04070.

[235] Hinton, G., Vinyals, O. & Dean, J. Distilling the Knowledge in a Neural Network (2015). Comment: NIPS 2014 Deep Learning Workshop, 1503.02531.

[236] Matin, S. *et al.* Teacher-student training improves accuracy and efficiency of machine learning inter-atomic potentials. https://arxiv.org/abs/2502.05379v1 (2025).

[237] Amin, I., Raja, S. & Krishnapriyan, A. Towards Fast, Specialized Machine Learning Force Fields: Distilling Foundation Models via Energy Hessians. https://arxiv.org/abs/2501.09009v2 (2025).

[238] Ekström Kelvinius, F., Georgiev, D., Toshev, A. & Gasteiger, J. Accelerating Molecular Graph Neural Networks via Knowledge Distillation. *Adv. Neural Inf. Process. Syst.* **36**, 25761–25792 (2023).

[239] Matin, S. *et al.* Ensemble Knowledge Distillation for Machine Learning Interatomic Potentials (2025). 2503.14293.

[240] Wang, R., Gao, Y., Wu, H. & Zhong, Z. PFD: Automatically Generating Machine Learning Force Fields from Universal Models (2025). Comment: 9 pages, 9 figures, 2502.20809.

[241] Merchant, A. *et al.* Scaling deep learning for materials discovery. *Nature* **624**, 80–85 (2023).

[242] Loshchilov, I. & Hutter, F. Decoupled Weight Decay Regularization (2019). Comment: Published as a conference paper at ICLR 2019, 1711.05101.

[243] Schütt, K., Unke, O. & Gastegger, M. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *Proc. 38th Int. Conf. Mach. Learn.*, 9377–9388 (PMLR, 2021).

[244] Simeon, G. & de Fabritiis, G. TensorNet: Cartesian Tensor Representations for Efficient Learning of Molecular Potentials (2023). Comment: NeurIPS 2023, camera-ready version, 2306.06482.

[245] Adamo, C. & Barone, V. Toward reliable density functional methods without adjustable parameters: The PBE0 model. *J. Chem. Phys.* **110**, 6158–6170 (1999).

[246] Grimme, S., Antony, J., Ehrlich, S. & Krieg, H. A consistent and accurate ab initio parametrization of density functional dispersion correction (DFT-D) for the 94 elements H-Pu. *J. Chem. Phys.* **132** (2010).

[247] Goerigk, L. & Grimme, S. A thorough benchmark of density functional methods for general main group thermochemistry, kinetics, and noncovalent interactions. *Phys. Chem. Chem. Phys.* **13**, 6670 (2011).

[248] Cheng, B., Engel, E. A., Behler, J., Dellago, C. & Ceriotti, M. Ab initio thermodynamics of liquid and solid water. *Proc. Natl. Acad. Sci.* **116**, 1110–1115 (2019).

[249] Pakdel, S., Olsen, T. & Thygesen, K. S. Effect of Hubbard U-corrections on the electronic and magnetic properties of 2D materials: A high-throughput study. *npj Comput Mater* **11**, 1–9 (2025).

[250] Skinner, L. B., Benmore, C. J., Neuefeind, J. C. & Parise, J. B. The structure of water around the compressibility minimum. *J. Chem. Phys.* **141**, 214507 (2014).

[251] Errington, J. R. & Debenedetti, P. G. Relationship between structural order and the anomalies of liquid water. *Nature* **409**, 318–321 (2001).

[252] DiStasio, R. A., Jr., Santra, B., Li, Z., Wu, X. & Car, R. The individual and collective effects of exact exchange and dispersion interactions on the ab initio structure of liquid water. *The Journal of Chemical Physics* **141**, 084502 (2014).

[253] Riebesell, J. *et al.* A framework to evaluate machine learning crystal stability predictions. *Nat Mach Intell* **7**, 836–847 (2025).