Spin-Aware Neural Network Interatomic Potential for Atomistic Simulation

by

David A. Bloore

B.S., Physics (2005), University of Massachusetts at Amherst
M.S., Nuclear Science and Engineering (2013), Massachusetts Institute of Technology

SUBMITTED TO THE DEPARTMENT OF NUCLEAR SCIENCE AND ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY IN NUCLEAR SCIENCE AND ENGINEERING

AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
JUNE 2021

Signature of the author:____
David A. Bloore
Massachusetts Institute of Technology
May 20, 2021

Certified by:____
Ju Li
Battelle Energy Alliance Professor of Nuclear Science and Engineering and Professor of Materials Science and Engineering
Thesis Advisor

Certified by:____
Koroush Shirvan
John Clark Hardwick (1986) Career Development Professor
Thesis Reader

Certified by:____
Michael J. Short
Class of '42 Associate Professor of Nuclear Science and Engineering
Thesis Committee Member

Certified by:____
Ju Li
Battelle Energy Alliance Professor of Nuclear Science and Engineering and Professor of Materials Science and Engineering
Chair, Department Committee on Graduate Students

Spin-Aware Neural Network Interatomic Potential for Atomistic Simulation

by

David A. Bloore

Submitted to the Department of Nuclear Science and Engineering
on May 20, 2021, in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

ABSTRACT

Computational modeling is key in materials science for developing mechanistic insight that enables new applications. *ab initio* methods capture exceptional phenomenological richness to high numerical accuracy, but at high cost and limited scale. Empirical potentials are faster and scale better, but cannot compare to *ab initio* in numerical and physical accuracy. Machine learning (ML) interatomic potentials (IPs) of recent years offer a balance: excellent phenomenology and accuracy, while scaling well and at moderate cost. Interatomic potentials are generally formulated as functions of atomic coordinates only—i.e. spin-agnostic. For materials whose structures or energetics are influenced by spin, this is insufficient. Iron's strong magnetism is coupled to its mechanical properties. This confounds spin-agnostic IPs because they implicitly use an expectation value across spin states for a given geometry.

Thus, this work offers a novel ML engine employing: (1) novel basis functions that translate spin information into neural network (NN) inputs, (2) and novel NN architectures that improve their ability to learn and express relationships between geometry, spin, and energy.

When applied to a broad dataset with high variance in both geometry and spin, the new bases achieve a 4x reduction in energy prediction error compared to the spin-agnostic Behler-Parrinello (BP) framework, and 5x using both the new bases and new NN architecture. When applied to a high spin-variance dataset, the new bases reduce energy prediction error by over 10x. Even when applied to a dataset with *low* spin-variance, the new bases reduce energy prediction error by 45%. These predictive improvements come at an increased computational cost of about 5% compared to spin-agnostic BP using only the new bases, but roughly 3x using both the new bases and NN.

This work presents two physical predictions to further elucidate the capabilities and value of the Spin-Aware NN IP (SANNIP). First, Monte Carlo (MC) spin relaxations using SANNIP exhibit behavior consistent with hysteresis in that the relaxed spin state is dependent on its initial alignment. Second, MC spin relaxations resolve the temperature beyond which ferromagnetically initialized systems lose their magnetization to between 1100 and 1150K, which is roughly consistent with experimental measurement of the Curie Temperature (T_C) of 1043K.

The evaluation of numerical accuracy and physical predictions demonstrate the utility of the novel bases and NN architectures. Future work can generate a broader dataset and deploy SANNIP potentials in molecular dynamics (MD) seeking insight into the role of spin in mechanical properties, defect interactions, etc. Additional bases and can explicitly treat externally applied electric and magnetic fields. Further NN architecture innovations can incorporate transfer learning into treatment of multi-component systems. This work is foundational to and enabling of many new avenues of investigation in computational materials science with the aim of improving materials design, fabrication, remediation, recycling, and disposal.

Thesis Advisor: Ju Li
Title: Battelle Energy Alliance Professor of Nuclear Science and Engineering and Professor of Materials Science and Engineering

Thesis Reader: Koroush Shirvan
Title: John Clark Hardwick (1986) Career Development Professor

Thesis Committee Member: Michael J. Short
Title: Class of '42 Associate Professor of Nuclear Science and Engineering

Acknowledgements

First, I'd like to thank Professor Ju Li for his input, opinions, and funding support. Within the Ju Li Group, I'd like to thank Dr. Qingjie Li for our discussions of science and practice.

Special thanks to my thesis committee Professors Michael Short and Koroush Shirvan. Their advice and mentorship was invaluable with regard to setting measurable goalposts and objectively assessing progress in meeting them. I especially appreciate Professor Michael Short's input on my thesis defense and presentation style; it really made my defense an opportunity for growth and learning event that otherwise it would not have been.

Special thanks also to Professor Emeritus Sidney Yip for his advice and mentorship. His articulation of the purpose of the events leading up to graduation and emphasis on compactness and brevity in writing and presentation really helped me focus on the critical path.

Thank you Professor Kord Smith for teaching your first reactor physics course in 2011, and subsequently sponsoring my taking the Ph.D. qualification exam. You offered a combined challenge in physics and computational methods which opened new intellectual doors for me and broadened my view of science and engineering.

I'd like to thank the men and women of the United States Navy, for your continuing service, and supporting me during my time as a masters student at MIT. Special thanks to the Surface Warfare Officer and Engineering Duty Officer communities whose qualification pins I am authorized to wear. My service was a time of great personal growth and introduced me to people from all walks of American life, for which I will always be grateful.

I'd like to thank the United States Department of Energy Office of Energy Efficiency and Renewable Energy; it is an honor to be a part of work that enables a better future.

Thank you Dr. Alexandre Guion for your friendship over the years, from studying for the Ph.D. qualifying exam in 2012, to inviting me to the party where I met my wife, to being my best man. It's truly special to have your good influence at such vital times in my life.

Thank you to my friends old and new, and family. Mom and dad, thank you for raising me, supporting me, and encouraging me when I needed it and through difficult times.

Lastly, thank you to my wife Sabrina, for being there with me for the day-to-day, the ups and downs, and supporting my pursuit of dreams. I'll be more practical from now on, I promise.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Why is atomistic simulation useful? What does it matter to society? To what extent does it enrich the experience of being human?

Technology, to me, is a practical embodiment of science. It accomplishes some purpose. It is the harnessing of scientific knowledge to affect some kind of change in the physical world. Science absent any technological application is theory.

Technology starts with mathematics, because without mathematics one really cannot express quantitative hypotheses or derive meaning from measurements. Mathematics enables physics, which in turn enables chemistry, and then biology.

In the realm of technology, materials science turns out to be exceedingly broad in its applications. We clothe ourselves and live in structures that keep the elements away, transport ourselves around using somewhat durable objects with heat engines, thrusters, and/or batteries across land, sea, air, and space, and communicate using a broad range of devices and modalities.

All materials eventually fail, and physics of failure analyses inform strategies to make better materials. [1] Electron microscopy, x-ray diffraction, and other experimental techniques give us a window into the atomic scale, but we are limited in our ability to observe phenomena on ultra-short timescales and precisely manipulate individual atoms.

Conceptual understanding of the physics of failure, radiation damage, hydrogen embrittlement, catalysis, and other phenomena at the atomistic level is sought via simulation using the best theories science has to offer. As our technology develops farther from its origins in sticks, stones, and fire, atomistic simulation provides a key environment to run experiments *in silico* that are either impossible or prohibitively expensive.

1.1 Motivational Background

Dislocations, grain boundaries, and point defect interactions have been widely studied with regard to their influence on alloys' ductility, yield strength, and toughness. [2] Nuclear engineering is further concerned with radiation damage, transmutation, and hydrogen and helium affecting the mechanical properties of nuclear reactor pressure vessels (RPVs) throughout their service lives. [3]

Physical experiments on actual materials are expensive, time consuming, and produce radioactive waste. Samples must be fabricated, subjected to reactor-like conditions, and undergo a battery of mechanical property tests. These experiments must generally be performed across a wide range of fluences, oxidation pervasiveness, or other corrosion states. While informative, these experiments are not perfect because as-operated reactor conditions are hard to reproduce faithfully in all aspects.

Despite these challenges, regulatory agencies must be sure that new and novel materials will protect operators and the general public from the radioactivity of a fission reactor. It cannot be overstated the extent to which prevention of RPV materials failure concerns all life on Earth. Operating RPVs are therefore only built from materials licensed by regulatory agencies after their review of comprehensive testing programs.

Computational prediction of irradiated and corroded materials properties helps: (1) elucidate how and why materials failures occur, (2) motivate strategies to mitigate failure and minimize consequences, and (3) explore mechanical properties of RPV material candidates.

To make progress toward those ends, they must achieve a minimum level of accuracy that makes them useful. Once the threshold of usefulness is reached for accuracy, the next concern is computational efficiency (speed, hardware, and energy requirements). This largely determines the scale at which a method can be applied, and in turn what questions a given method can answer.

1.1.1 Atomistic Simulation and Scale in Time and Space

Different physics dominate different scales. General Relativity is concerned with the largest of scales, while Quantum Mechanics is concerned with the smallest. [4, 5] Typical human experiences on the surface of the Earth are Newtonian. [6] Since the physics dominating different scales are so different, the math used to describe the energetics and dynamics of each realm are also different.

$$G_{\mu\nu} \equiv= R_{\mu\nu} - \frac{1}{2} R g_{\mu\mu} = \frac{8\pi G}{c^4} T_{\mu\nu} \quad [4] \tag{1.1}$$

$$i\hbar \frac{\partial}{\partial t} \langle x|\psi\rangle = \int H(x, x')\psi(x')dx' \quad [4] \tag{1.2}$$

$$\vec{F} = m\vec{a} \quad [6] \tag{1.3}$$

First principles *ab initio* codes are in many ways the most accurate window into the Å-scale phenomenology of atomistic behavior. [7] By looking through this window, science and engineering have gleaned a wealth of mechanistic insight and practical prescriptions. However, atomistic simulation remains limited in its ability to tell us about human-scale phenomena (meters and minutes, for example). Quantum Mechanics solvers cannot directly model creep. They cannot easily tell us about large scale defects in metals—how they form and their affect on mechanical properties. [8]

But these questions matter. They are the concern of scientists and engineers who design and construct the technological artifice of the modern world. The physics of failure in materials science will always be a concern to any society with limited resources.

One way to connect the meso/marco-scale and the Å domain is molecular dynamics (MD) [9], which uses what is known as an interatomic potential (IP) to calculate the system's total energy and then uses that energy's derivatives to calculate atomic forces and evolve the system in time. [10]

Monte Carlo (MC) is another valuable tool that enables us to explore the configuration space of a system given a specific temperature and environment by using random "moves" and an IP to calculate the energy of the new configuration, and through this iteration sample the Boltzmann Distrubution. [11]

1.1.2 Known Unknowns in Atomistic Simulation

How can one know that what an IP is told to do is correct?

Specifically, how can one verify that the functional form of a given interatomic potential is correct and physical? In principle it is possible to execute x-ray diffraction experiments on materials of interest sampling a broad domain of temperatures and stresses, however those methods only obtain averages in time and space. It is not possible to exhaustively and conclusively verify the details of a prediction atom-by-atom for any given snapshot time, much less at scale across representative samplings of the Boltzmann Distrubution. Impurities and other confounders obscure reality further, since IPs have historically tended to treat very few elements—or even just one.

IPs can be used to make various predictions such as lattice constant, elastic tensor, melting point, boiling point, heat capacity, and so forth—however lattice and other constants can be included in the fit's objective function, and many predictions depend more on aggregate performance than high-precision and accuracy for each individual atom. [12]

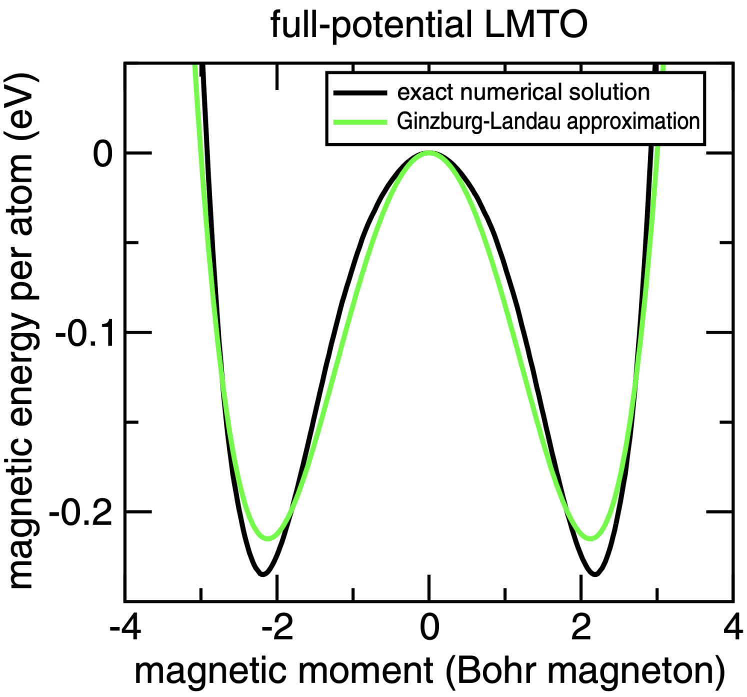
Questions of arbitrary displacements of single atoms from perfect lattices are nearly unanswerable experimentally. Perfect infinite medium experiments corresponding to cohesive energy curves are similarly impossible. When talking about IPs and thinking about how to construct them, it is important to always keep these ideas in mind. As exciting as it can be to formulate, fit, use, and evaluate interatomic potentials, it must be stated clearly what cannot readily be known about them and from them.

1.1.3 Magnetism in Atomistic Simulation

The vast majority of IPs developed are functions of atomic coordinates and element/redox only—i.e. spin-agnostic. When IPs are modeling chemical Groups 1, 2, and 3, noble gasses, or otherwise modeling weakly magnetic systems, this is a reasonable or even preferable approximation. To explicitly treat atomic spins in those cases would dramatically escalate costs for little to no gain.

At this point the reader may reasonably ask, what's the big problem with IP spin-agnosticism? The first part of the answer to this question is that strongly magnetic materials have an energy landscape that is highly dependent on their magnetic state. Fig. 1.1 shows energy vs. magnetic moment for iron. The energy difference between local energy extrema is $>$ 200 meV/atom—which is about eight times the average kinetic energy of particles at room temperature, or about five times what is often called "chemical accuracy" or 1 kcal/mol. [13, 14] That is an enormous magnitude of error for calculations in atomistic simulation.

Figure 1.1: Magnetic energy per atom vs. magnetic moment for Fe. [15]



Imagine the following thought experiment. Let's say there exist N sets of atomic coordinates $\vec{R}_i$, and then for each set of coordinates there is a set of M atomic magnetic moments $\{\vec{S}_{i,0} \ldots \vec{S}_{i,M}\}$. A dataset of this type can be characterized:

$$\vec{R}_i \wedge \{\vec{S}_{i,0} \ldots \vec{S}_{i,M}\} \quad \forall\, i \in \{1, \ldots, N\} \tag{1.4}$$

Then the energy of each combination of coordinates and atomic magnetic moments is calculated, that set of energy calculations is used to train a spin-agnostic IP, and the IP learns everything perfectly. What will it predict as the potential energy of a system when it only sees the atoms' coordinates? It will output the average value of the energies for that set of coordinates across all of its spin configurations, and therefore nearly always be wrong.

$$E(\vec{R}_i) = \frac{1}{M} \sum_{j=1}^{M} E(\vec{R}_i, \vec{S}_{i,j}) \tag{1.5}$$

This could be improved by using an expectation value in place of an average, but that requires careful measure of the probability of each spin configuration.

Even if an IP assumes and is fitted using a ferromagnetic ground state for all geometries, the magnetization of a real material is a partial function of temperature. That means that identical geometries, which can occur at different temperatures (albeit with different frequencies), will have different spin states and therefore energies.

This weakness in all spin-agnostic IPs for α-Fe is well known to many groups. [15, 16] Computaitonal modeling of α-Fe that is both accurate and precise in expressing the full range of its physical phenomenology remains a grand challenge to this day.

There are further confounders. Defects and clusters, for example, are known to differ in spin from bulk lattice. [17–19] The situation worsens in multi-element cases involving transition metals, for example considerable magnetic frustration effects occur in defect-free iron-chromium systems. [20]

Figure 1.2: Collinear vs. anti-ferromagnetic Fe_{13}; the collinear arrangement is 37.8 meV lower in system energy. [18]

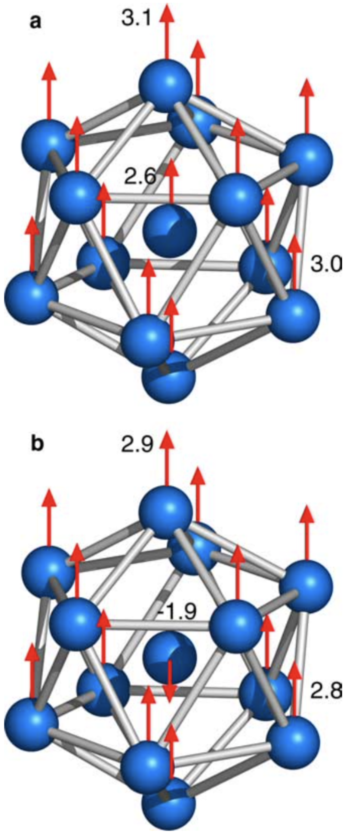


Figure 1.3: Local minima in structure, magnetic moment, and energy for Fe_5. [18]

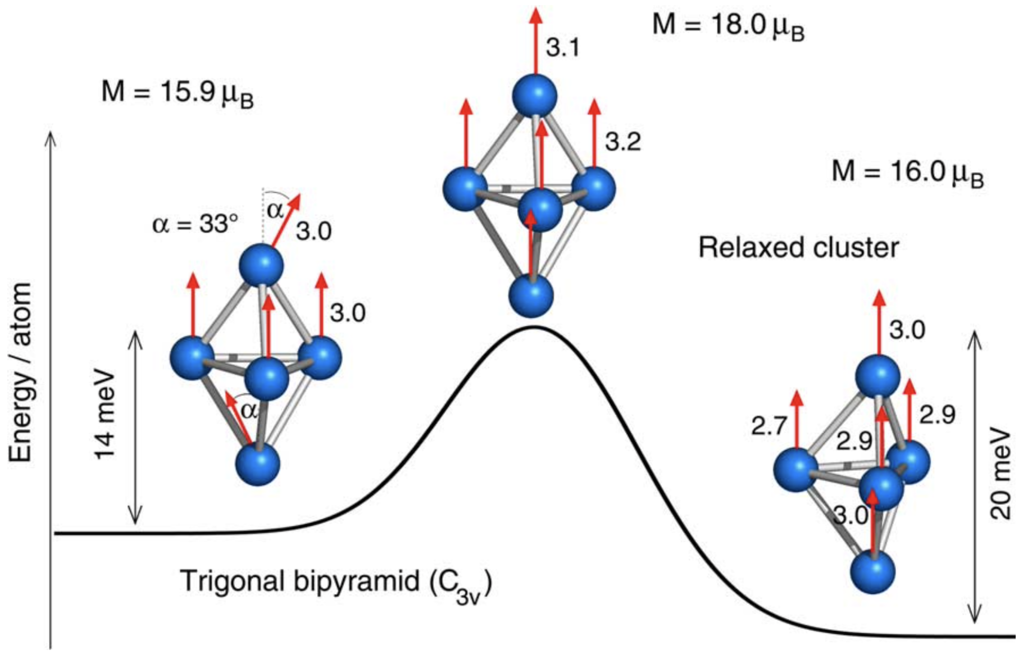
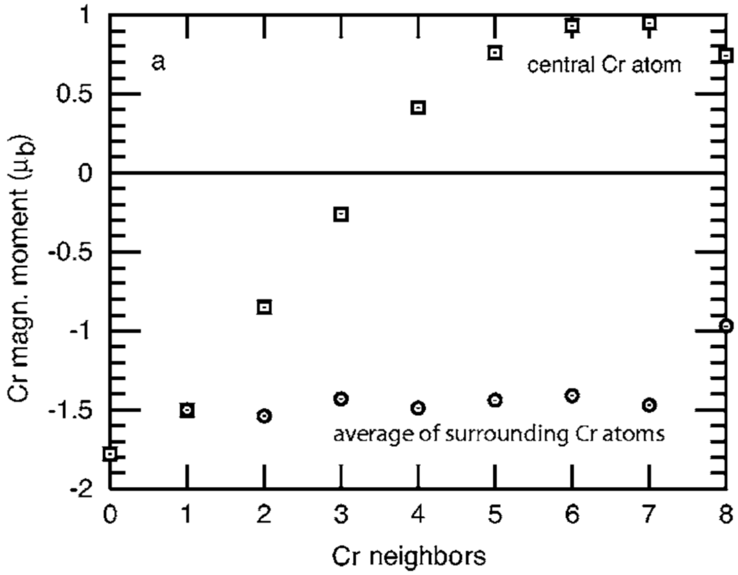


Figure 1.4: Magnetic moment of Cr atoms in Fe as a function of Cr nearest neighbors. [20]



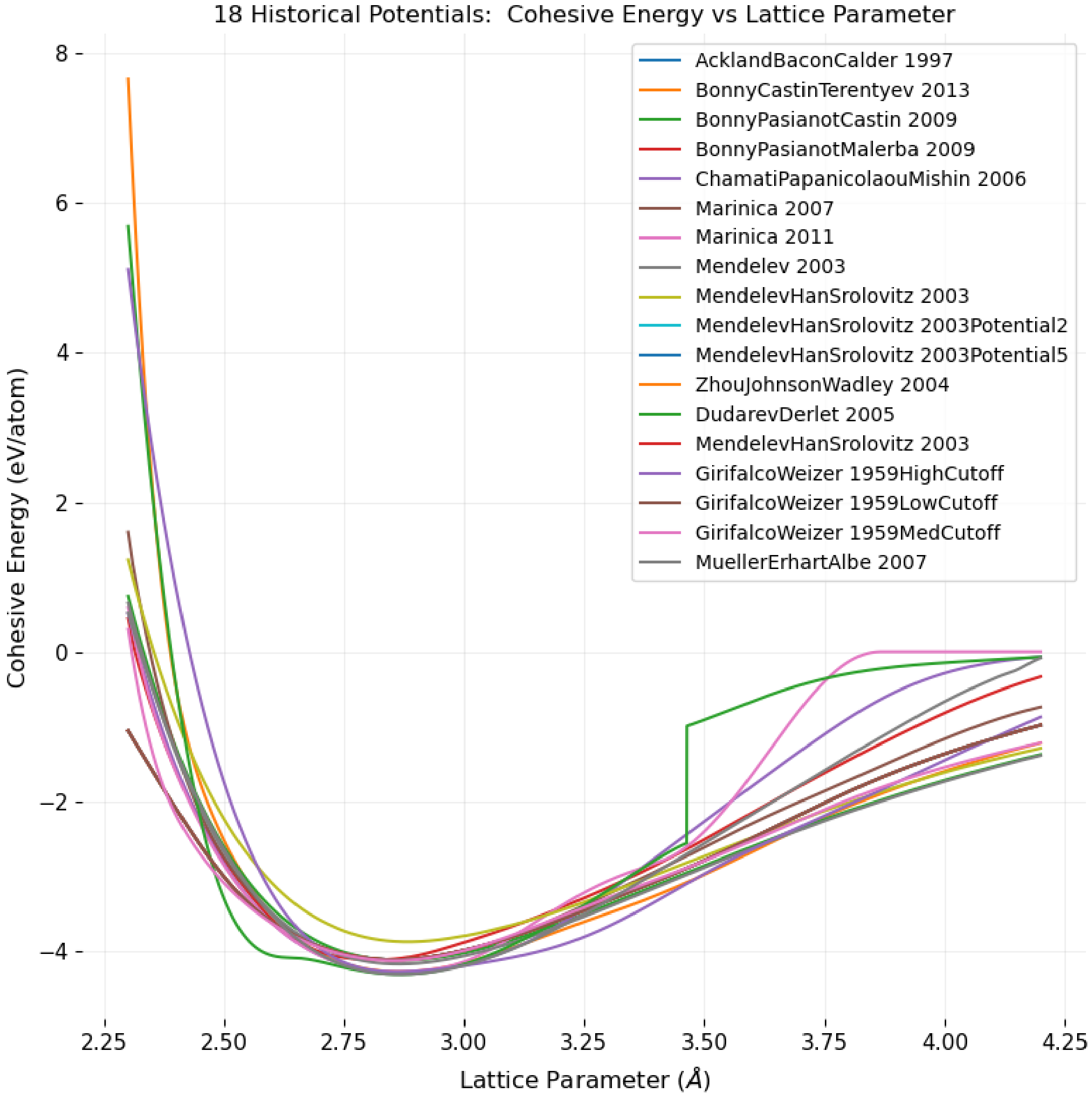
Another complexity comes from the idea that spin and lattice excitations have similar characteristic relaxation times. [21] There is also the phenomenon of magnetic hysteresis. Given this reality, time-dependent defect interaction spin artifacts further diminish the physicality of spin-agnostic IPs.

1.1.4 Historical Potentials for Fe

Fig. 1.5 shows us a plot of cohesive energy vs lattice parameter for 18 different interatomic potentials developed for Fe. [15, 17, 22–33] They are available from either NIST or openKIM. [34, 35]

There exists considerable diversity in terms of the global minimum energy, lattice parameter, and overall shape, smoothness, and values and derivatives near the ends of the displayed domain. Each one of these is the result of a significant effort in terms of resources and time, with different: approaches to generating data, strategies for fitting, and intended applications. Some might be specifically aiming to produce a physical Peierl's barrier, while others may be looking for reasonable defect production via radiation cascade. Other significant aims are modeling of defect formation energy, defect migration barriers and mobility, and extended defects such as dislocations, grain boundaries, interstitial aggregates, and voids.

Figure 1.5: 18 historical potentials for Fe. [15, 17, 22–33]



Density functional theory (DFT) is the primary method for producing datasets by which to fit or train IPs. [36, 37] Spin-polarized DFT calculations were collinear-only for many years. [38, 39] A multitude of empirical potentials have been fitted using datasets generated under this regime.

Non-collinear atomic magnetic moment DFT provides the capability for exploring the energy dependence of three atomic magnetic moment degrees of freedom instead of just one. [40, 41]

Constrained atomic magnetic moments calculations allow for explicit control of atomic magnetic moments—even when the specified spin configuration is *not* the ground state. This is an iterative procedure that appends a penalty term to the energy prediction, and requires progressively increasing the energy penalty coefficient λ until the the energy penalty becomes sufficiently small. The computational expense of this procedure is much higher than spin-polarized or ever non-collinear spin calculations. [19, 42]

More recent work has shown that properties prediction can depend on complex theoretical accounting of magnetism's influence. [43]

It is therefore clear that non-trivial systematic error arises in potentials for Fe due to the underlying methodologies used to generate the fitting data, and that improvements to those methodologies require at least as much attention as do IP fitting methodologies themselves.

1.1.5 Machine Learning (ML) and Atomistic Simulation

With ML it is postulated that the physics present in *ab initio* calculations can be learned and expressed by applications that give the end user access to near *ab initio* phenomenology at a fraction of the computational cost. This is expressed graphically in Fig. 1.6, where "EAM" refers to the Embedded-Atom Method which was used for many of the potentials shown in Fig. 1.5. For the purpose of this discussion, EAM is a fast approximator of atomic level behavior but not as accurate or general as DFT.

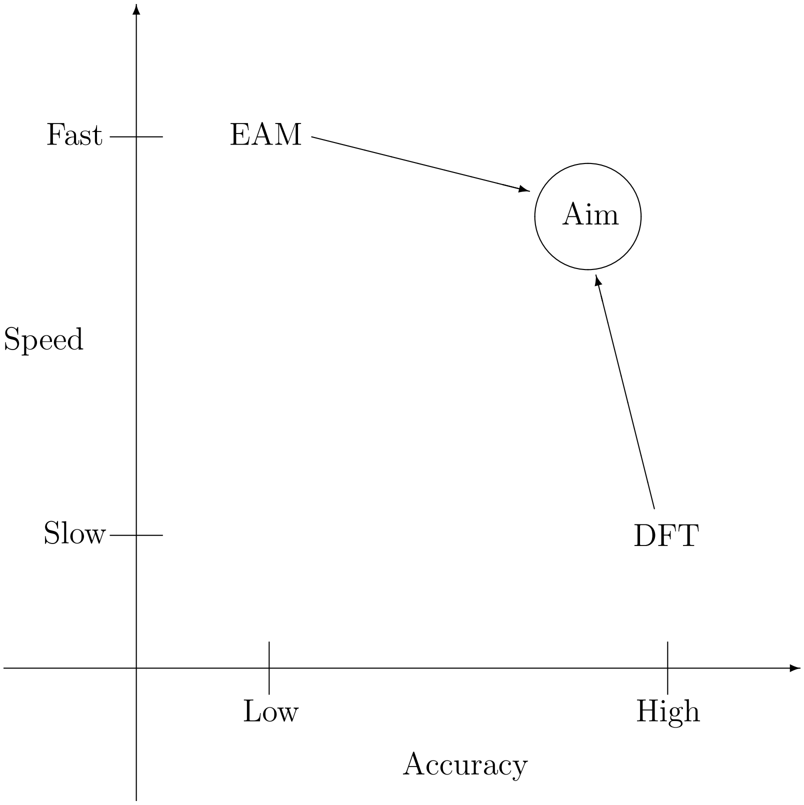


Figure 1.6: Aim of ML in atomistic simulation.

An ML IP paradigm is defined by its:

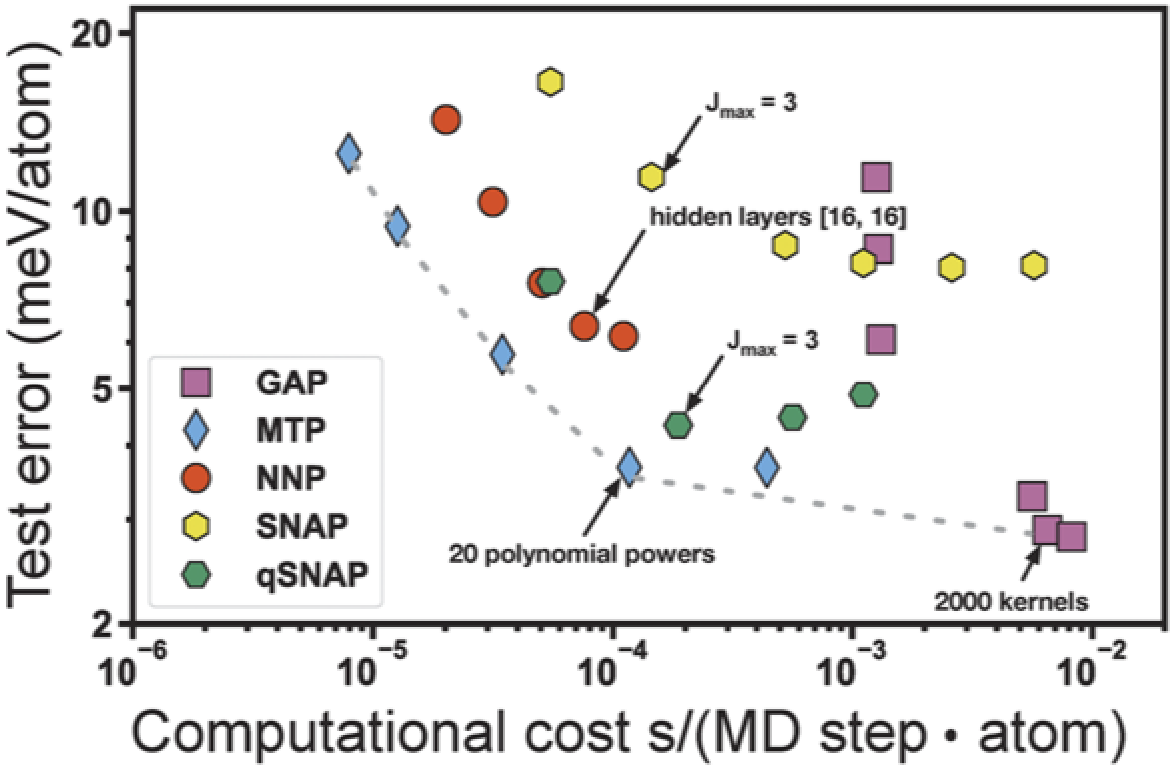
1. Featurization, which is the means by which raw data are processed into the quantitative inputs of the predictive model. Featurization is performed on a training dataset prior to training, and on new data prior to making predictions.
2. Regressor, which is a function that returns one or more continuously variable scalar quantities that are or map to a prediction. ML IPs tend to use one of three methods:
   - Linear regression, which finds a line of best fit. For this to work, the featurization must produce features that linearly correlate with the quantity to be predicted (system energy in the case of IPs).
   - Kernel regression, which uses a non-linearity and all elements of the training dataset to define a predictor.
   - Neural networks, which will be described in the next Chapter.

3. Training methodology, which is the means by which the trainable parameters of the regressor are adjusted using the training data.

An instance of a trained ML IP is also dependent on the dataset used to train it.

The following four sections briefly describe the four main ML IP methods in literature as of this writing. There is a recent cross-comparison of these four main methods looking at energy and force prediction error, and physical property predictions for six elements of various crystal and band structures: Li, Mo, Cu, Ni, Si, and Ge. [44] Iron is notably absent, although nickel is ferromagnetic. Fig. 1.7 shows a comparison of accuracy and cost comparison across the four major methods mentioned in the next subsections (qSNAP is an expansion of SNAP). "NNP" denotes the Behler-Parrinello method, which will be described shortly in brief and later at length. One key aspect of this comparison is that the training dataset available to each method was the same.

Figure 1.7: "Test error versus computational cost for the Mo system. . . . Timings were performed by LAMMPS calculations on a single CPU core of Intel i7-6850k 3.6 GHz. Black arrows denote the "optimal" configuration for each ML-IAP . . ." [44]



1.1.5.1 Behler-Parrinello (BP)

BP can succinctly be expressed as:

1. A set of basis functions used to convey atomic coordinates and species identity into,
2. Neural networks that calculate the potential energy of a system of atoms.

Relevant implementation details of BP are presented in the next Chapter. [45]

1.1.5.2 Gaussian Approximation Potential (GAP)

GAP uses “smoothed” atomic density function and bispectrum featurization to feed kernel regression. [46] Kernel regression defines a non-linear transform of features based on a kernel function and the complete list of features used to train the predictor. [47] It is remarkably accurate and precise in predicting, for example, vacancy formation energies, and the Peierl’s barrier in α-Fe. [48] While these are remarkable achievements, GAP’s practical utility for the typical end user of molecular dynamics simulations is limited by its extremely long calculation time. As seen in Fig. 1.7, GAP is about two orders of magnitude slower than BP in the presented application.

1.1.5.3 Spectral Neighbor Analysis Potential (SNAP)

SNAP uses the same featurization framework as GAP, but feeds a linear regressor (instead of a kernel regressor). [49] This sacrifices some of the accuracy and precision of GAP in favor of several orders of magnitude reduction in execution time. Users of SNAP can partially compensate for the less expressive regressor by using more features than a similarly purposed GAP potential, but of course this is done at the expense of computational efficiency.

qSNAP is an extension of SNAP that uses fits a regressor using quadratic terms in addition to linear terms. [50]

1.1.5.4 Moment Tensor Potential (MTP)

MTP computes tensors using Kronecker products of atomic coordinates and then condensing them down into scalars that feed a linear regression. [51]

1.1.6 Spin-Aware Interatomic Potentials

While most interatomic potentials are spin-agnostic, spin-aware potentials are not without precedent. The major options available for Spin-Lattice Dynamics (SLD) as of this writing are SPILADY and the LAMMPS SPIN package.

1.1.6.1 SPILADY

SPILADY is an SLD program which uses a geometric empirical potential and overlays spin Hamiltonian terms that allow changes to direction or both direction and magnitude of spins. It also uses a correction term that compensates for the implicit or explicit assumptions about magnetism built into the geometric potential. [52]

$$\mathscr{H} = \mathscr{H}_{latt} + \mathscr{H}_{spin} + \mathscr{H}_{corr} \tag{1.6}$$

$$\mathscr{H}_{latt} = \sum_i \frac{\vec{p}_i^2}{2m} + U(\vec{R}) \tag{1.7}$$

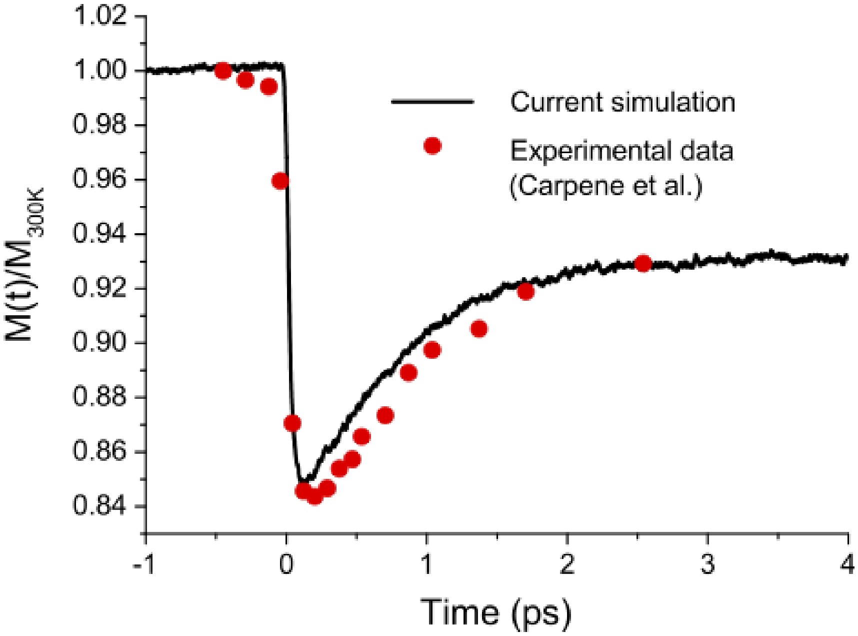
$$\mathscr{H}_{spin} = -\frac{1}{2}\sum_{i,j} J_{ij}(\vec{R})\vec{S}_i \cdot \vec{S}_j + \sum_i A_i(\vec{R})S_i^2 + B_i(\vec{R})S_i^4 + C_i(\vec{R})S_i^6 + D_i(\vec{R})S_i^8 \tag{1.8}$$

$$\mathscr{H}_{corr} = \frac{1}{2}\sum_{i,j} J_{ij}(\vec{R})|\vec{S}_i||\vec{S}_j| \tag{1.9}$$

$J_{ij}(\vec{R})$ is called an exchange function, which has an arbitrary functional form and is fitted using *ab initio* data. [53,54] When electrons spatial coordinates are exchanged, but the wavefunction remains the same, there is an energy penalty—because two electrons cannot occupy the same state and therefore relax to unique wavefunctions that are optimal for their specific spatial distribution. The "exchange interaction" is quantum mechanical in nature and lacks a classical analog. A_i, B_i, C_i and D_i are Landau coefficients used to predict the energy of an atoms spin to allow for spin magnitude to change in dynamics calculations. These fitted exchange forms are not general; a form appropriate to the lattice structure must be chosen. This leaves the door open for large errors when lattice structure is not unambiguously defined, such as for crack tips, amorphous regions, surfaces, clusters, nanowires, etc.

SPILADY has been used to simulate laser pulse demagnetization in bulk BCC iron and show good agreement with experiment. The laser pulse excites the electrons of the system which disorders the atomic spins causing magnetization to drop, then rebound and settle at a lower value due to the increased temperature .

Figure 1.8: Laser pulse demagnetization in Fe. [52]

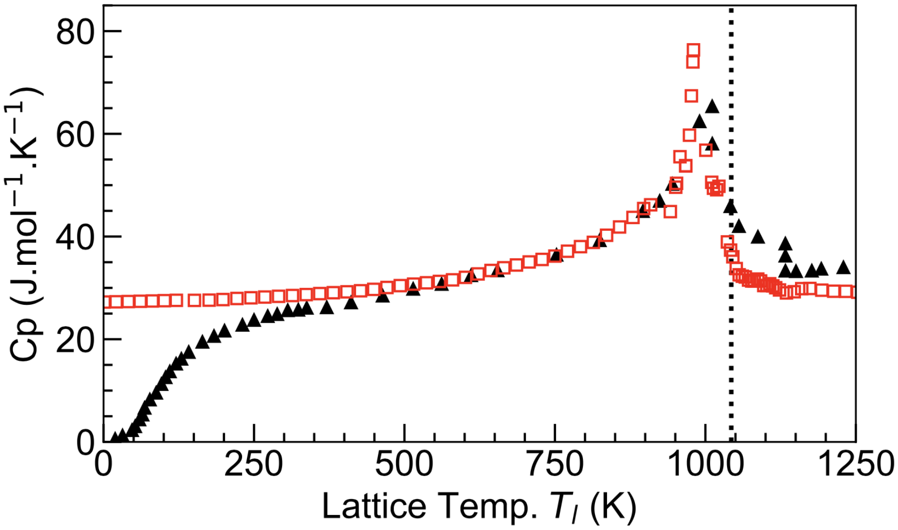


1.1.6.2 LAMMPS SPIN package

LAMMPS is a well-known MD program that has been a standard tool in academia and industry for many years. [55] LAMMPS implements SLD via the SPIN package that expands the definitions of atoms to include spins, and offers several different spin Hamiltonian terms with fitted coefficients that govern spin interactions. [21] The contributions of the spin-based potential terms are overlaid onto a geometric potential, commonly EAM, for the purposes of computing forces on atoms and derivative of energy with respect to spin. LAMMPS as of this writing offers only the ability to permute spin direction.

There is also a recent publication using SNAP in conjunction with the fitted exchange function of the LAMMPS SPIN package that examines the specific heat of iron, and observes an abrupt change near the Curie Temperature. [56] The authors do however mention that they forced the magnetization to match experimental value for each temperature sampled, which is different from magnetization being a dependent variable. This was done by using different temperatures for the lattice and spin thermostats (which are functions that randomly permute atom velocities and spins to simulate interaction of the system with a heat bath). [57]

Figure 1.9: Specific heat as a function of temperature in Fe as determined using SNAP in the LAMMPS SPIN package. [56]



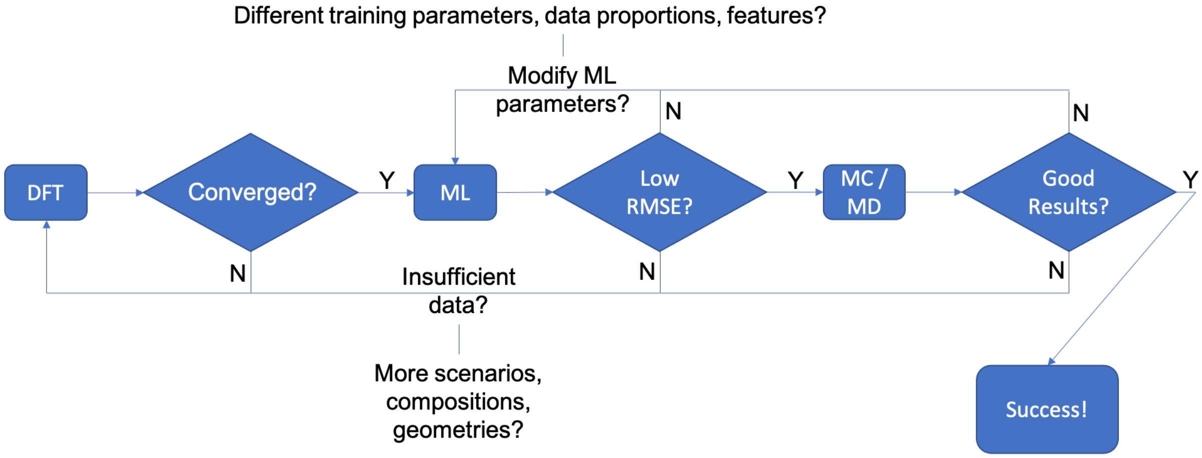
1.1.6.3 magnetic Moment Tensor Potentials (mMTP)

While there is not an open-source platform for researchers to investigate the performance of mMTP, its authors report using a small number of geometric configurations and for each of those a large number of spin configurations and obtain a respectably low energy RMSE. This is a generalization of the above mentioned MTP approach. This work is available in pre-print as of early 2020, when the bulk of the work of this thesis was already completed. [58]

1.1.7 The Potential Development Roadmap

The process of developing an ML IP is conceptually the same as for any other. It starts with data generation, then fitting (or ML), and finally executing simulations and making predictions.

Figure 1.10: Machine-learned interatomic potential development roadmap


Different training parameters, data proportions, features?
Modify ML parameters?
N
N
DFT
Converged?
Y
ML
Low RMSE?
Y
MC / MD
Good Results?
Y
N
Insufficient data?
N
N
More scenarios, compositions, geometries?
Success!

Any step along this process can be fraught with peril. Data generation can be a major hurdle, particularly if calculations in one's area of interest are not stable—or if the quantity of data required for an application is huge due to the degrees of freedom in the system. ML model training involves fine tuning of many hyperparameters that all materially affect numerical accuracy. Large molecular dynamics (MD) simulations can take days or even weeks, and failure to achieve predictive aims can reveal both new data requirements and re-evaluation of or evolution of ML methods. For example, the LAMMPS Benchmarks website reports 32k Cu atom simulations taking 1.86 μs per 5 fs timestep using one processor. [59] 32M atoms simulated for 5 μs of simulation time would take 1,860,000 s, which is ~517 hours or ~ 3 weeks, to execute using the hardware mentioned. Using multiple (or very many) processors reduces computation time, but not necessarily by a factor equal to the number of processors.

1.2 Scope of Thesis

The intent of this work is to innovate in the space of interatomic potential development methodology, and showcase the new methodology with proof-of-concept (POC) applications that *require* explicit treatment of spin.

The new work presented in this thesis is a comprehensive effort to:

1. Develop a new Spin-Aware, NN IP (SANNIP) development methodology since none exists as of this writing (and no spin-aware ML IP methodology existed at thesis outset),
2. Develop a spin-variation specific approach to generating a training dataset using Density Functional Theory (DFT),
3. Achieve significant numerical accuracy improvements using SANNIP as compared to the state-of-the-art spin-agnostic NN IP approach (BP) on a dataset for iron containing both high-geometric and spin variation,
4. Train and use the new IPs to perform Monte Carlo spin relaxations in iron using different spin initializations to show SANNIPs are capable of exhibiting phenomenology consistent with hysteresis,
5. Train and use the new IPs to perform Monte Carlo spin relaxations of ferromagnetic iron at different temperatures to resolve the Curie Temperature of iron, and
6. To quantify SANNIP's execution time in MC / MD environments in relation to other methods.

The work presented is the result of years of continuous iteration between these and similar objectives, and ends with conclusions and suggestions for future work.

1.2.1 New Interatomic Potential Development Methodology

As explained in Section 1.1.3, existing spin-agnostic methodologies cannot fully model the complex physical phenomenology of magnetic materials—particularly for α-Fe. If pure iron is to be modeled from low temperature ($\sim$0K) up to 5,000 or 10,000K, then the situation is even worse. This work offers a solution to that problem via the addition of spin basis functions to the Behler-Parrinello method. These spin basis functions translate information about each atoms' magnetic environment to its NN with the intent of improving accuracy and resolution of the PES.

The spin bases are fundamentally enabling with regard to the expansion of BP into strongly magnetic systems across a large domain of temperature and spin states.

This work also introduces novel neural network architectures that have an increased capacity to learn and express relationships between geometry, spin, and energy in atomistic environments. These developments are motivated by the insight that lead to development of convolutional neural networks applied to image analysis: "If you know something about the structure of your problem, for sure, you should try to design your neural network so that it takes advantage of that structure." [60] Here, basis functions can be grouped by their informational character and subsequently each group input into its own 1D convolutional layer. This is analogous to the grouping of pixels by proximity for input into 2D convolutional filters.

1.2.2 New Spin-Aware Dataset Generation Approach

DFT allows users to fix the positions of a group of atoms in a specified periodic structure, and then evaluate both the potential energy of that system and the forces on each atom. It is explicitly seeking the electronic ground state of the system, which includes the magnetic contribution due to unpaired electron spins. This causes DFT to gravitate toward a zero Kelvin spin temperature without regard to the energy state (and implied temperature) of a perturbed lattice—and that tends to be non-physical for typical finite-temperature systems.

In practice this means that paramagnetic (PM) data for Fe is hard to get with DFT because it will often just relax the spins to collinear without regard to the initialization.

Literature exists for a way around this, so-called "constrained magnetism" calculations which superimpose an energy penalty into the objective function (energy) based on how well the spins in DFT's solution match those specified by the user. This does allow for explicit exploration of perturbed magnetic states, however these calculations are exceptionally costly—rendering this method intractable for the purpose of generating a large training dataset in a "reasonable" amount of time.

This work therefore presents a DFT workflow that traps it in desired spin states while still converging to the energy minimum for that state. This executes at least one order of magnitude faster than the conventional "constrained magnetism" approach, and enabled the conducting of a broad survey of spin states essential to the proper functioning of NN methodologies, given their inherent sensitivity to gaps in the training domain.

1.2.3 Proof-of-Concept Regarding Numerical Accuracy

Herein is presented a multi-faceted cross-analysis comparing the new spin bases and NNs, separately and together, against BP using a large dataset and two subsets chosen for their high-spin or high-geometric variation character.

BP is chosen as the comparator arm because it is the state-of-the-art NN IP methodology as of this writing. It is by no means a "fault" of BP that it is spin-agnostic, because, as mentioned in Section 1.1.3, spin is not important in many contexts and would slow such work considerably. It is only for systems of significant magnetic character that SANNIP has an advantage. SLD using empirical approximations for spin interactions have not been considered because they rely on fixed functional forms that are tailored to particular lattice phases, and additionally were at thesis outset dependent on classical geometric potentials. There is now a published SLD approach using SNAP to treat geometry, however it still relies on fixed-form empiricism to treat spin. It is also not clear if empirical spin-aware potentials properly treat defects, which seems unlikely given the strong phase dependence of empirical spin relations.

This work shows an over 10x reduction in energy prediction RMSE for a high spin-variation subset compared to spin-agnostic BP. This explicitly shows that the novel spin bases allow for paramagnetic configurations to be clearly distinguished from ferromagnetic configurations in a way that is not possible using spin-agnostic potentials.

On a large dataset that varies highly in both the geometry and spin configuration spaces, the combined use of new spin bases and novel NN architecture achieves a 5x reduction in energy prediction RMSE compared to spin-agnostic BP.

Lastly, even when applied to a largely ferromagnetic data subset with high geometric variation, the new spin bases *still* yield a 40% reduction in energy prediction RMSE—conclusively proving that explicit treatment of spin is essential to accurately predicting energy in iron and that the extent of efficacy increase is proportional to the level of spin-variation in the training dataset.

1.2.4 Novel Application 1: Consistency with Magnetic Hysteresis

This work's first spin-specific physical exhibition is consistency with experimental observations of magnetic hysteresis. Monte Carlo was used to relax the spin configurations of a fixed lattice at 300K from three different initializations: ferromagnetic (FM), paramagnetic (PM), and anti-ferromagnetic (AFM). The resultant magnetization was found to be greater for the FM initialized case, and the PM and AFM cases relaxed to a lower magnetization. The PM/AFM magnetization termini were also the same. This trend is consistent with experimental observations of magnetic hysteresis in that magnetization is a function of the magnetic history of the material—that a relaxed spin state depends on its initialization.

1.2.5 Novel Application 2: Resolution of Curie Temperature of Iron

This work closes with presentation of the first ever resolution of the Curie Temperature (T_C) of iron via atomistic simulation using a NN IP. There are *ab initio* mean-field approaches predictions of T_C , and Spin-Lattice Dynamics T_C predictions using fitted exchange functions—however the present work uses neither of those approximations. [56, 61–63] SANNIP allows for SLD using a single, inseparable function of both coordinates and spins trained solely on *ab initio* data without any adjustment of parameters to better match experimental results. The spin state of fixed lattices was initialized to perfect FM alignment and then relaxed using MC at a distribution of temperatures. Largely FM character is retained in the relaxed configurations up to 1100K, and at 1150K the relaxed state is strongly PM. This first-in-class prediction conclusively proves the potency of our approach in mapping macroscopically observable magnetic states to the atomistic relationships between spin and energy.

1.2.6 Quantify Computational Cost Relative to Other Methods

Increased accuracy is not enough to justify the use of a new method. End users care about execution time, and how methods compare against each other. To that end, SANNIP has been compared against BP, both with and without the new NN architecture. Using a relatively simple Figure of Merit-based (FOM) approach, it is shown that the greatest benefit comes from using SANNIP with the conventional BP NN, followed by SANNIP with the new CNN. While the CNN does improve energy prediction RMSE relative to the SANNIP using the BP NN by about 20%, the execution time triples. This last bit of accuracy therefore comes at dramatically increased cost (although certainly not an order of magnitude increase in cost). While some users may benefit from that last bit of accuracy, SANNIP using the BP NN is likely to be an optimal configuration for most users.

Chapter 2

Neural Network Regressors and Behler-Parrinello

Behler-Parrinello neural network interatomic potentials were first introduced in 2007, in a now-seminal Physical Review Letter. [45] It ushered in a new era of ML and NN approaches to learning and expressing the potential energy surfaces (PESs) of pure materials. Many, *many* papers have been published that apply, expand, or compete with this method. Since the core of this thesis' innovations build upon BP methodology, it and one of its applications are here explained.

But first, let's take a closer look at the application of NNs to regression tasks in general, and then to the application of NNs to an example interatomic potential, Lennard-Jones. The reader is encouraged to reflect upon this seriously despite the seemingly obvious nature of the trade-offs discussed because deep understanding of these realities is key to any practitioner's success in diagnosing and remediating deficiencies in a NN IP. NN IP fitting is different from EAM or other classical approaches in part due to the specific nature of NNs as regressors. The reader is advised to consider, if low sampling density can cause trouble even in 1D as shall soon be shown, what then is the expected difficultly in learning a PES across both geometry and spin?

2.1 Neural Network Regression Essentials

The task of reviewing seminal texts on the fundamentals of neural networks is left to the reader, regarding: their construction, training through backward propagation, and so forth. There are many excellent resources in the tutorials of all major packages such as Tensorflow, PyTorch, and so on. [64, 65] This work does however offer a small, targeted tutorial on the relationship between dataset size, noise, quantity of trainable parameters, extrema, boundaries, overfitting, and functional forms in the context of neural networks as regressors—and ultimately for expressing potential energy surfaces (PESs).

Fitting a NN regressor can be a risky proposition, and one that depends on the shape of the underlying function, the number of samples in the domain of interest, and the noise in the range / dependent values / experimental observations.

This is because *a NN is a universal approximator*, and therefore *NNs lack any inherent functional form.* [66] This is in stark contrast to analytically-derived, closed-form functions which are often smooth and continuous in their values and at least first-order derivatives. Fixed functional forms themselves are a significant amount of information, thus enabling effective treatments of certain problems using only a relatively small amount of fitting data. To accomplish a practically useful fit using an NN requires a lot more data, but also allows for freedom from functional forms that are themselves approximations and non-universal.

For the base case demonstration, an underlying function of $f(x) = \sin(x)$ is to be learned. Gaussian noise has been superimposed.

The NN is a softplus-activated bilayer, 4 neurons per layer.

$$\text{Softplus:} \quad f(x) = \ln\left(1 + e^{x}\right) \tag{2.1}$$

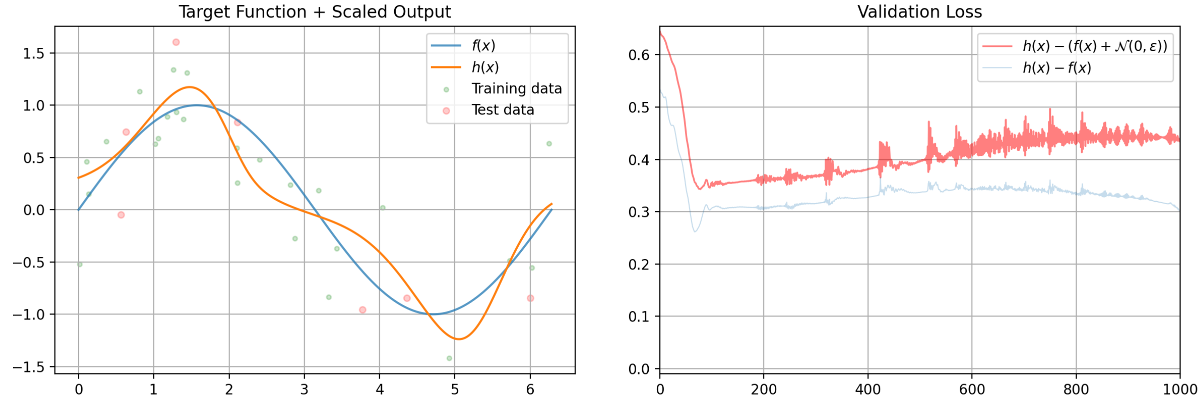
In the plots on the left of each Figure, the green dots are training data points, and the red dots are test or validation data points unseen during the training process. (Test and validation are interchangeable here, although this is not universally the case.) The losses reported in the lower plots are validation losses as a function of training epochs.

The blue line in the plots at right, denoted $h(x) - f(x)$, is the loss between the prediction and the underlying function evaluated at the domain of the validation points. If, for example, the prediction is fitting the underlying function better than the validation data, then the blue line will be lower than the red line. If, on the other hand, the prediction is closer to the validation points than the underlying function the blue line will be higher than the red line—which is characteristic of overfitting.

2.1.1 Few samples, high noise

Fig. 2.1 is obviously the most troublesome case. The density of the green dots is not constant across the domain, and in sparely populated areas the fit bows significantly away from the underlying function. The validation error reaches a minimum early on in the training process, and climbs in an oscillatory manner thereafter. The testing point near about (3.8, -0.95) is particularly affected by the NN fit pulling away from the underlying function to approach the training point near (4, 0), which it is able to do because of the ultra-sparse density in that local area.

Figure 2.1: NN fit of $\sin(x)$: few samples and high noise.

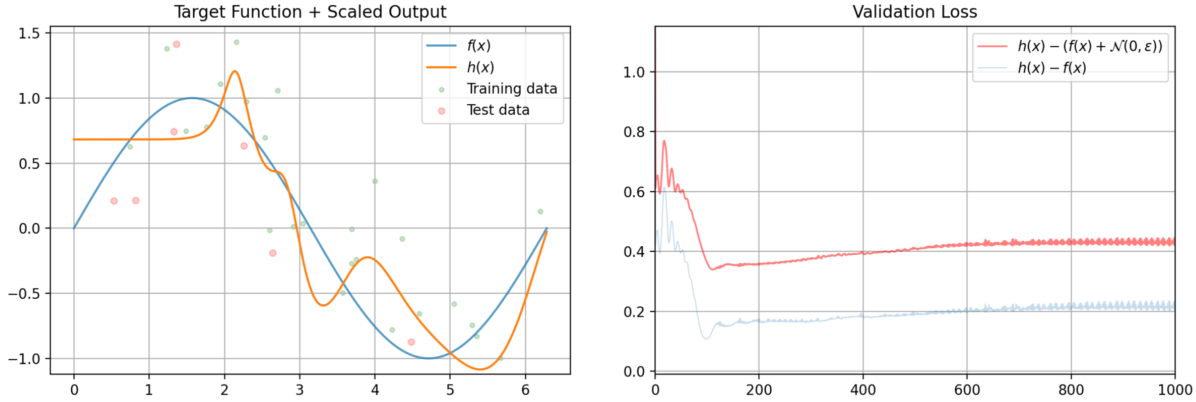


The interpretation of this pursuant to IP development is that sparsely sampled, noisy data regions are likely to be associated with high error and non-physical behavior. It is important to know what that looks like. If $\sin(x)$ were the form of a potential being fitted, then the local extrema (equilibrium, for example) positions and energies would be very wrong. The maximum force exerted by the potential would occur in the wrong place and be too strong. In fact, very high force would be present at the actual underlying minimum. If the body in the potential ever got near the boundaries of the domain the energy and forces would be wrong there too. It might reasonably wondered how this could get worse.

2.1.2 Overfitting few samples, high noise

If one were to try and obtain a better fit in this situation by increasing the number of nodes (neurons) per layer to 16, then one might obtain Fig. 2.2. This situation *has* gone from bad to worse, allowing the increased expressivity of the NN to form a uniquely odd shape that squiggles around haphazardly in order to reach as many green dots as possible.

Figure 2.2: NN fit of $\sin(x)$: overfitting with few samples and high noise.



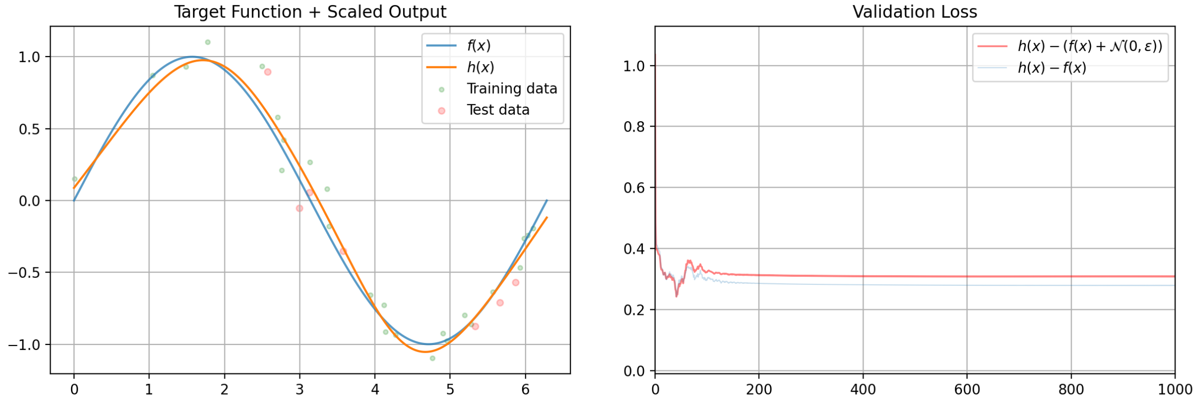
Now there are extra extrema that don't exist physically, very high forces, and a flat boundary condition at one end—totally wrong!

2.1.3 Few samples, low noise

If dense sampling cannot be achieved, then sometimes noise can be reduced. This is shown in Fig. 2.3, and the situation is much improved. The extrema are still off, but the overall shape is well-behaved and the boundaries are looking better.

This predictor actually fits the training points better than the underlying function, as indicated by the blue line in the bottom left quadrant being higher than the green line. It is another example of overfitting, but overfitting here is due to a paucity of data samples rather than noisy data or surplus untamed dynamism in the NN. It is worth noting that the validation error reaches a minimum and bounces off of it and then essentially flatlines, rather than increasing steadily.

Figure 2.3: NN fit of $\sin(x)$: few samples and low noise.

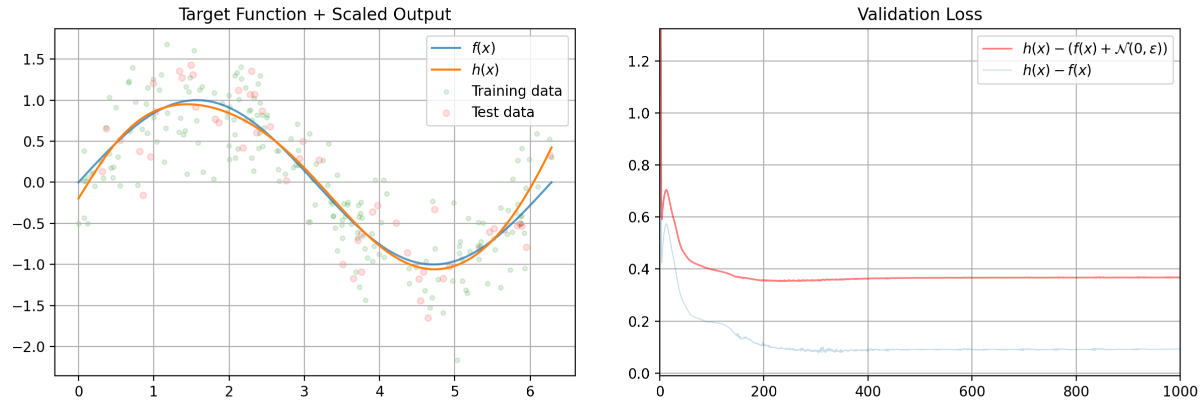


Physically, this is a much more useable potential than the above examples—although if used in the low temperature regime then it wouldn't be reasonable to expect perfect results. Zooming in on the region around the minimum reveals that the distortion of the potential's shape might be enough to foil delicate calculations.

2.1.4 Many samples, high noise

Fig. 2.4 shows that increased sampling improves the overall shape of the fit despite high noise, but extrema are still wrong and there remain high errors in value and derivative near domain boundaries. Once again there is a minimum validation error, however in this case there is no acute bounce and the difference between the stable fit and the minimum is slight.

Figure 2.4: NN fit of $\sin(x)$: many samples and high noise.

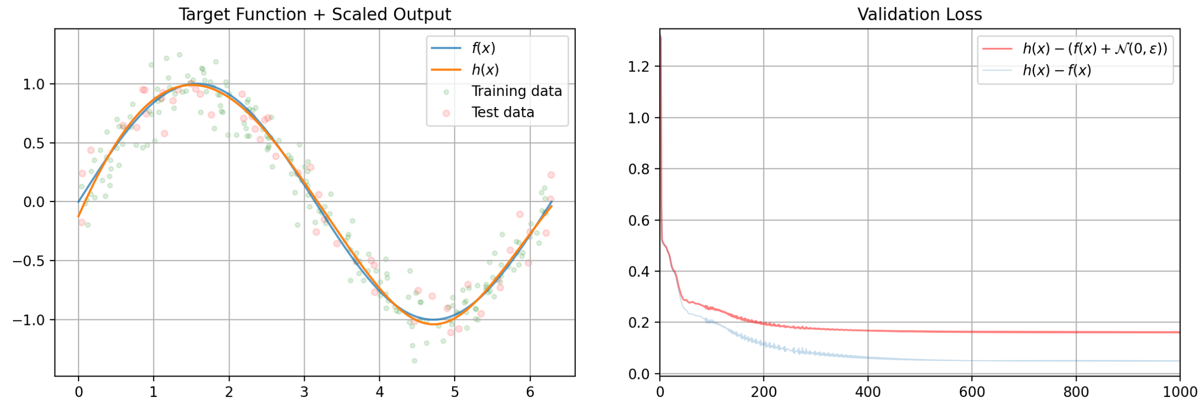


Practically this means that enough noisy data can more or less give okay phenomenology—but only well inside the training domain. Venture out near the outer extremities and noise takes over, obscuring the underlying reality one is trying to approximate.

2.1.5 Many samples, low noise

Fig. 2.5 shows ideal conditions, with the best fitting of shape, extrema, and boundary conditions. In this situation the validation loss reaches its minimum at training's end, and the underlying function is fit better than either the training or validation points.

Figure 2.5: NN fit of $\sin(x)$: many samples and low noise.



Physically this means that NN IPs can be excellent given low noise and sufficiently dense sampling in the domain of interest. If sampling is extended beyond the immediate domain of interest then the result is improved by shifting the ever-present poor boundary performance to regions outside the intended scope.

This may explain why sampling very high energy states, non-physical crystal structures (like FCC iron at low temperature), and a wide range of otherwise unusual coordination numbers can be important for fitting potentials—perhaps even critical for NN IPs.

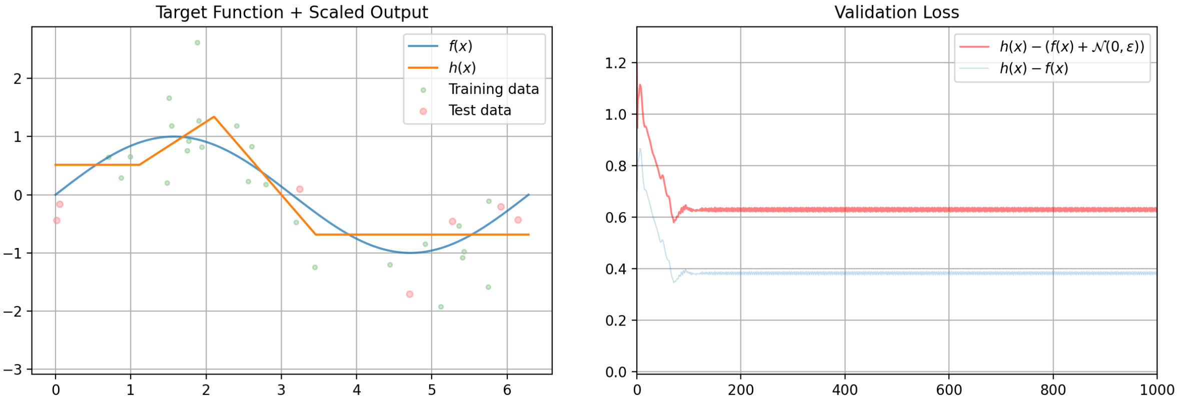
2.1.6 Rectified Linear Units

Rectified Linear Units (ReLU) draw their inspiration from rectifiers used in Electrical Engineering in AC to DC converters (and other places). [67] They are ubiquitous in NN packages such as Tensorflow, PyTorch and others. [64, 65] They compute quickly—however they do have a discontinuous first derivative at the origin.

$$\text{ReLU:} \quad f(x) = \begin{cases} 0 & x < 0 \\ x & \end{cases} \tag{2.2}$$

What happens if ReLU-activated NNs are used to fit $\sin(x)$? Fig. 2.6 shows the same 4-node-per-layer bilayer, just changing activation—training on the noisy, sparsely sampled space of our first example.

Figure 2.6: ReLU-activated NN fit of $\sin(x)$: few samples and high noise.



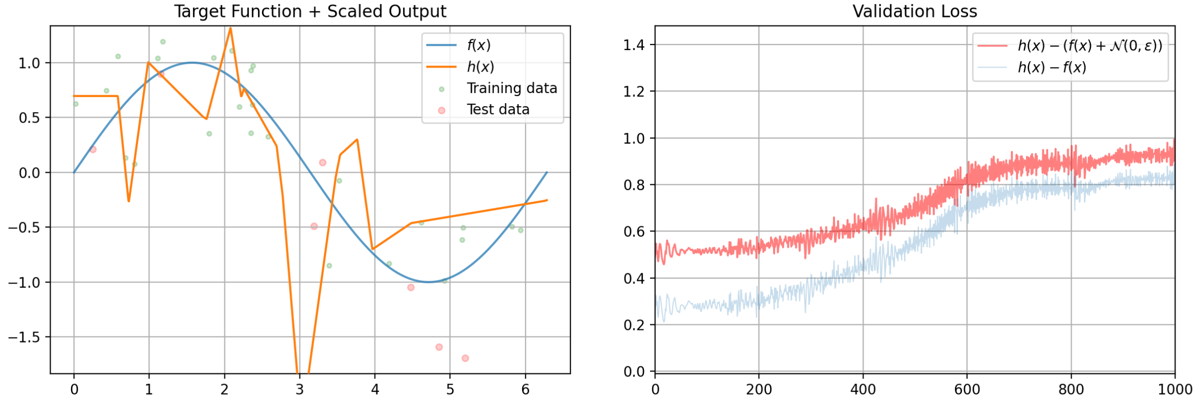
This result is quite poor, and not useable as a potential. The minima are entire regions, and the forces are constant for large contiguous subsets of the domain, and zero (or nearly zero) for more than half of the domain.

2.1.7 Overfitting using ReLU

Fig. 2.7 shows an attempt to improve performance vs the ReLU base case by using 16 neurons per layer. The functional form appears more inspired by the architecture of Frank Gehry than by $\sin(x)$, and once again the validation error reaches an early minimum and rises strongly and consistently.

In this scenario the prediction fits the training points better than the underlying function by a lot—overfitting at its worst. There are not many samples, the data is noisy—and the functional form of the activation function is poorly suited to expressing the underlying functional form of the data.

Figure 2.7: ReLU-activated NN fit of $\sin(x)$: overfitting few samples and high noise.

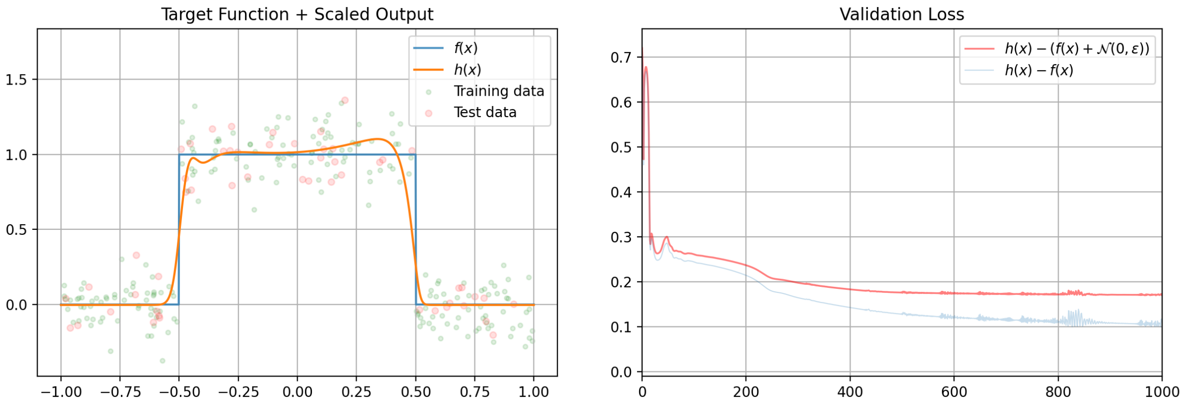


2.1.8 Square Wave

Trying to fit a square wave with a softplus-activated NN regressor reveals a lot actually. It shows that sharp changes in a function's derivative are hard to treat properly, even with high sample density and low noise. This is something to watch for anytime physical reality is likely to be represented by functional forms with sharp boundaries or regions with high second-order (or higher) derivatives.

The fit ends up skewed, and with artifacts that are not representative of the underlying function. This is similar in nature to the Gibb's phenomenon arising in Fourier Series approximations to the square wave.

Figure 2.8: NN fit of square wave: many samples and low noise.



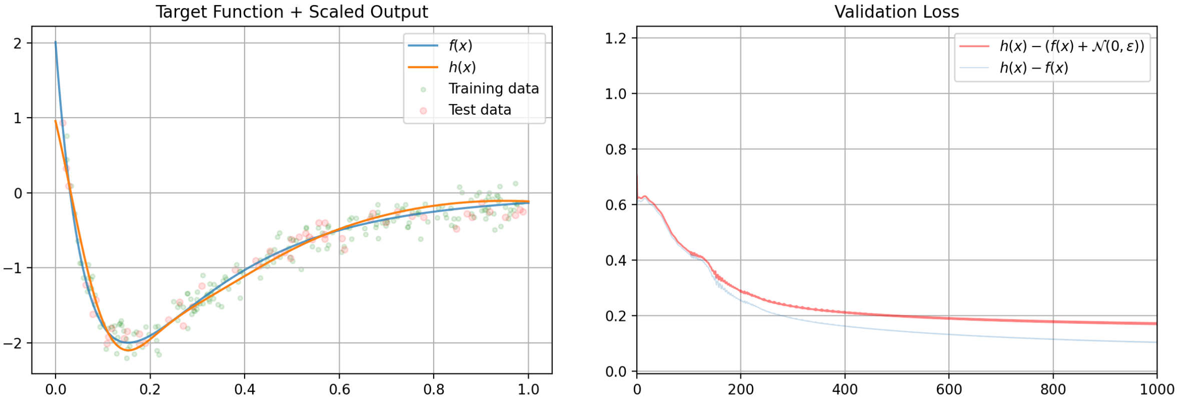
2.1.9 Lennard-Jones

The previous examples have focused on functions that are simple mathematically, but not representative of functions that express useful atomistic potential energy surfaces (PESs). The Lennard-Jones (LJ) model is relatively simple mathematically and has a physically-relevant functional form with: (1) a single energy minimum, (2) a steep asymptotic ascent on one side of the minimum, and (3) a gradual approach to zero on the other. [68, 69]

$$V_{LJ}(r) = \frac{A}{r^{12}} - \frac{B}{r^{6}} \tag{2.3}$$

The following fit uses same softplus-activated bi-layer as before:

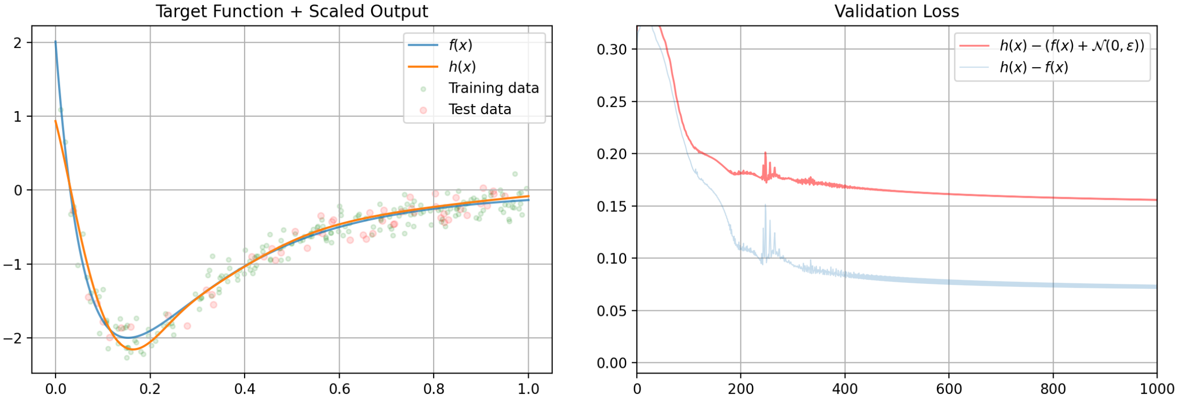
Figure 2.9: NN fit of Lennard-Jones: Poor fit for equilibrium.



This is perhaps okay at first glance, but not certainly not excellent in that the location and value of the minimum are wrong, and the potential is off by a lot at the lower bound. The forces under compression get too strong too fast, and not strong enough at increased compression. Above equilibrium interatomic distances, the forces oscillate between too strong and too weak. Note that statistically this is in the same highly-sampled/low-noise regime shown in previous examples.

How can this be fixed? What if the number of neurons in each layer are doubled?

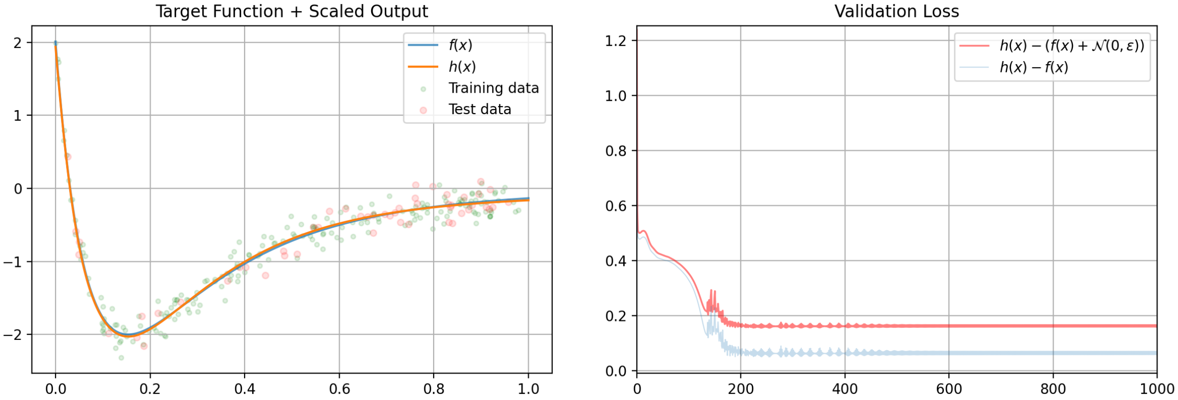
Figure 2.10: NN fit of Lennard-Jones: More neurons.



That didn't help. The fit of the above equilibrium interatomic distance region is better—but the minimum and form near it are worse!

It this therefore clearly seen that precisely fitting the equilibrium position with a NN can be quite difficult. What if the original network is kept and instead the distribution training frequency is changed to emphasize samples closer to the equilibrium point?

Figure 2.11: NN fit of Lennard-Jones: More frequently trained near equilibrium.



In order achieve an overall low error in the regions atoms are most likely to be found, and still trend correctly in outlier sub-domains, the frequency with which training samples have been presented to the NN was altered to over-emphasize samples near the equilibrium point. This is a critical insight to keep in mind as the reader ventures forward throughout the text of this thesis. The practical prescription is that sampling must be especially dense near equilibria and important local minima—otherwise the resulting potential will not be able to perform well near those points. When a potential is not concerned with equilibria (such as for high temperature solid phases, liquids, and gasses) then this is much less of an issue.

2.1.10 Data Subsets with different Means and Dispersions

An additional relevant scenario is the combining of data subsets that have the same underlying function, *but have different noise distributions*—non-zero and different means, and different dispersions.

This situation involves many degrees of freedom and thus its detailed exploration is left to the reader, however it is worthwhile to note that degree of subset domain non-overlap, sample count from each dataset, and the magnitude of the noise averages and dispersions are all extremely significant.

Historically, nearly all interatomic potential fitting occurs in this domain. Different DFT scenarios inherently span non-overlapping subdomains, and unavoidable *de facto* changes in k-spacing can affect what is effectively the mean value of the noise.

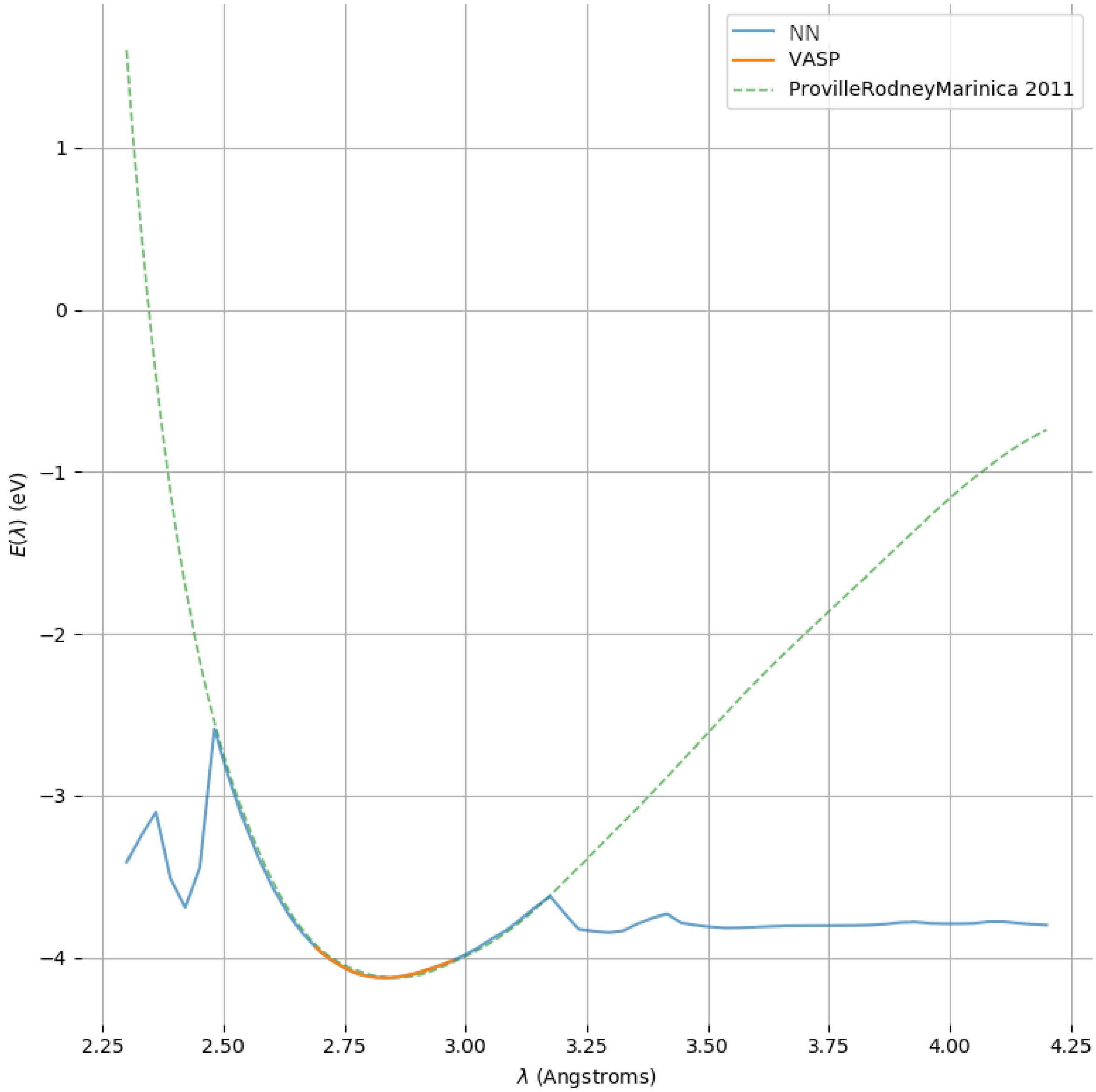
Specifically, if one executes 1x1x1 BCC unit-cell calculations and vary its size over a broad range from $0.75\lambda_{\mathrm{eq}}$ to $1.5\lambda_{\mathrm{eq}}$, and does not change the number of k-points used for each calculation, then the k-spacing changes from beginning to end by a factor of two.

If that reality is compared with the results of typical k-points convergence testing, it is seen that doubling the number of k-points for nearly any simulation box and set of atoms will change the energy result and the atomic magnetic moment predictions (no matter how narrowly the energy convergence tolerance is set).

2.1.11 Predictions Outside the Training Domain / Poor NN Extrapolation

NNs are exceptionally poor at extrapolating. When one presents an NN with inputs that are outside its training domain, the results are unpredictable. This is, as stated above, due to NNs' lack of an inherent functional form that would otherwise contain information to support extrapolation. Fig. 2.12 shows a rough account of predictions made inside and outside the training domain. Note that the error outside of the training domain is huge. (The orange "VASP" line is just one subset of training data, which extended to where the greed-dashed and blue lines diverge.)

Figure 2.12: NN fit of square wave: few samples and high noise.



In the case that NN IPs, the energy prediction was extremely low for lattice parameters outside the training domain—which is completely non-physical. In MD simulations this can lead to all of the atoms of a system collapsing or even imploding into a very small volume, and failure mode requires specific countermeasures to prevent.

One such countermeasure is to pair an NNIP with a ZBL overlay that smoothly takes over for interactions below a certain distance. DFT data for interactions in those regions are particularly costly anyway, given the *huge* perturbation from equilibrium that they represent. Then one needs to consider low density cases, where a relatively small amount are likely to improve the overall PES considerably.

2.1.12 Gradient Descent Sub-types

There are typically three types of “gradient descent” that depend on how much of the dataset is presented at each calculation of model weight updates.

1. Gradient descent (GD) or batch gradient descent (BGD) generally refer to using the entire training dataset at each weight update calculation.
2. Stochastic gradient descent (SGD) generally means randomizing the order of the samples in the training dataset and then calculating weight updates using each sample individually.
3. Mini-batch gradient descent (MBGD) generally means randomizing sample order and partitioning the random-ordered training dataset into multiple “mini-batches” or just batches. When batch size is specified and $\neq 1$, this is the algorithm being used.

There may be differences of opinion with regard to how these processes are named, but these are the essential three.

Each one has benefits and weaknesses that arise from the quantity of samples used at each weight update calculation, that are accentuated by the lack of inherent functional form in neural networks.

2.1.12.1 Effect of Batch Size on Training

Let’s say the Boston Symphony Orchestra is going to play The Four Seasons by Vivaldi, and a patron has sponsored the recording effort. Microphones are placed throughout the hall, each one receiving a different pressure-vs-time curve. They are like the trainable parameters of a neural network. The shape of the hall, the placement of the instruments, temperature, humidity, ambient pressure all affect how pressure waves travel from the instruments to the microphones. This is like the neural network. The instruments are the source of the *information* that is to be captured using the microphones. They are like data samples.

How many instruments play at once is the batch size.

Let's say each instrument plays its part in isolation. The microphones glean the best possible notion of each part. Some microphones might be in destructive interference zones (dead spots), and this can happen to NNs, but there really isn't any better way to capture the full detail of all instruments. This is SGD, and it obviously takes the longest time of any approach to recording Vivaldi. Maybe the Beatles recorded the White album this way, but that's a small number of parts. An additional problem with this approach is that the sound engineer would have an inordinate amount of work to do before the recording is ready for release. Even the best sound engineer will find it exceedingly difficult to replicate the essence of live experience for discerning ears.

Taken to the other extreme, when all instruments play at once (as before the live audience) then the microphones get a much better sense of the aggregate and how loud each instrument is relative to each other. The sound engineer's job is simplified by orders of magnitude. This is GD.

Mini-batch gradient descent is like having different groups of instruments play at the same time. The relative loudness of instruments can be pieced together more easily this way. Constructive and destructive interference will vary with different groupings, so that dead spots shouldn't be as much of an issue. The sound engineer has more work than in the GD case, but far less than SGD.

The overall time and effort required to satisfy the patron varies considerably with the number of instruments that play at once—and the specific choices of which instruments play at once.

Said differently, batch size is a materially significant parameter that affects the training time and efficacy of predictive models.

2.1.13 NN Fitting Tutorial Summary

The above is here summarized into phenomenological observations:

- Insufficient sampling is a cause of overfitting, and cripples NN regressors. Noise must be very low when few samples can be obtained in order for resultant fits to have even limited predictive capacity. For IPs fitted in this regime: local minima are probably wrong in both input and energy spaces, and predictions worsen significantly as the input domain boundary is approached.
- Overfitting can also be the result of a mismatch between data quantity and regressor complexity.
- High noise dramatically increases data requirements, and IP extrema and domain limit-approach error are present even with high sample density.
- Activation functions are essentially basis functions for expressing data's underlying functional form.
- Accurately fitting extrema with NN IPs requires an extra-high sampling density in the immediate vicinity of each extremum.
- Results outside the training domain are unpredictable and may have errors many orders of magnitude higher than the intra-domain RMSE.
- Large batch size tends to emphasize learning of general trends whereas smaller batch sizes or SGD tends to emphasize fine structure.

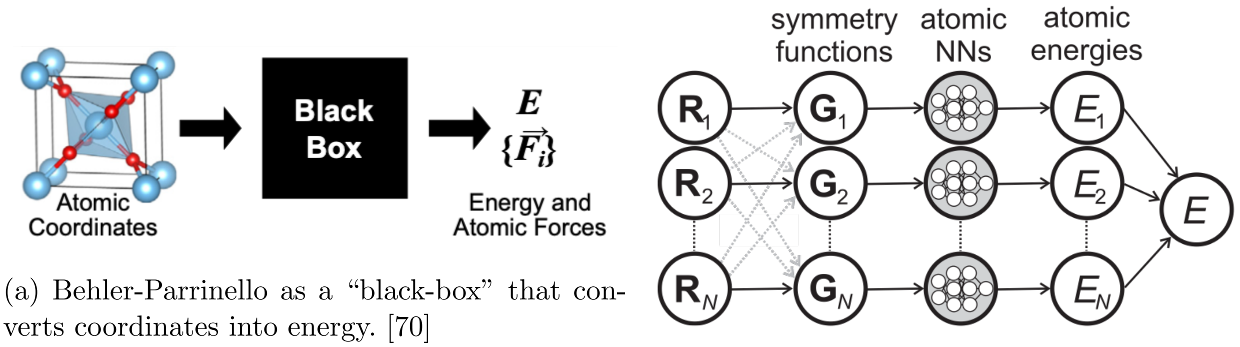
Despite the promise, press, and ubiquity of NN methods in recent years—they should not be delved into with reckless abandon. The following practical prescriptions are proposed:

- With insufficient sampling *and* high noise, efforts must concentrate on improving data generation workflows.
- Widen the training domain beyond the desired application space, and constrain the application space away from both extrema and boundaries.
- If fitting of extrema is required, sample intensely around each one.

- Regressor complexity should generally start with the minimum possible to achieve a baseline result, and tentatively explore from there.
- Choose an activation function appropriate to functional form. (The caveat to this is that features that are already basis expansions of raw data may work well with ReLU anyway.)
- Batch size must be optimized in a way so as to quickly adjust to averages and aggregate behavior while also learning smaller-scale details of the target functional form.

2.2 Behler-Parrinello Overview

BP is succinctly expressed as a "black box" that calculates the potential energy of a system of atoms, given their coordinates, as shown in Fig. 2.13a. Fig. 2.13b, shows the internals of the BP black box.



(a) Behler-Parrinello as a "black-box" that converts coordinates into energy. [70]

(b) Behler-Parrinello "black box" internals. [71]

Figure 2.13: Behler-Parrinello overview.

First, all atoms' coordinates (R_i) are operated on by the "symmetry functions" to produce G_i. Those G_i are "features" or "descriptors" that are input into a NN which returns the energy contribution of that atom to the system's energy. All individual atoms' contributions are summed to provide a single scalar quantity which can physically interpreted as the potential energy of the system. For N atoms, D "features" (which will be defined later), and a neural network function f_{NN}:

$$\text{Coordinates:} \quad \mathbb{X} \in \mathbb{R}^{3N} \tag{2.4}$$

$$\text{Featurization:} \quad \mathbb{G}(\mathbb{X}) \in \mathbb{R}^{N,D} \tag{2.5}$$

$$E = \sum_{i=1}^{N} f_{NN}(G_i) \tag{2.6}$$

$$\vec{F} = -\nabla E \tag{2.7}$$

Once the system's potential energy is computed, the gradient of the potential gives is the force on each atom. This is computed via the chain rule as shown in Equation 2.45 by first using backward propagation to obtain the partial derivative of energy with respect to the NN inputs, and multiplying that by the derivative of the inputs with respect to the atomic coordinates.

For extremely large systems that are important in MD simulations, this scheme scales linearly with the number of atoms in the system—in contrast to the scaling of DFT regimes, which at worst scale as the cube of the number of electrons. [72]

2.3 Featurization and "Symmetry Functions"

Behlerian "symmetry functions" are really a basis set decomposition of atomistic environments that construct what in machine-learning parlance are often referred to as "features." Features are just the quantitative information input into an ML process.

"Featurization" broadly refers to transforming, converting, or otherwise preparing "information" such that it can be presented in an exclusively quantitative form to a machine learning process (in the case of BP, a NN regressor). Information can be collections of pixels that form an image, the words of a phrase or sentence, the atomic coordinates of a periodic system, the presence or absence of a condition, or anything else that can be expressed quantitatively or qualitatively. Featurization can be accomplished according to any arbitrary set of rules (including conditional logic), and generally “raw” features are further processed to improve the efficacy and efficiency of the training process.

This section highlights the significant aspects of BP featurization that are relevant to this work.

2.3.1 Translational, Rotational, and Ordinal Invariance

Since the potential energy of a system of atoms is independent of coordinate translation, axis rotation, and labeling of atoms of the same element (or isotope if need be), it is necessary to ensure that any potential energy calculation scheme has the same properties:

$$E(\vec{r}_0, \vec{r}_1, \ldots \vec{r}_N) = E(\vec{r}_0 + \vec{o}, \vec{r}_1 + \vec{o}, \ldots, \vec{r}_N + \vec{o}) \quad \forall\, \vec{o} \in \mathbb{R}^3 \tag{2.8}$$

$$E(\vec{r}_0, \vec{r}_1, \ldots \vec{r}_N) = E(M\vec{r}_0, M\vec{r}_1, \ldots, M\vec{r}_N) \quad \forall \text{ rotation matrices } M \tag{2.9}$$

$$E(\vec{r}_0, \vec{r}_1, \ldots \vec{r}_N) = E(\vec{r}_1, \vec{r}_0, \ldots, \vec{r}_N) \quad \text{for atoms of the same element/isotope} \tag{2.10}$$

This can be implemented in various ways, and BP implements translational and rotational invariance as a property of the featurization using the distance function and dot products, and ordinal invariance using the commutative property of sums via summing atomic contributions to system energy (regardless of the form of atomic energy contribution regressors).

$$d(\vec{r}_1, \vec{r}_2) = d(\vec{r}_1 + \vec{o}, \vec{r}_2 + \vec{o}) \tag{2.11}$$

$$d(\vec{r}_1, \vec{r}_2) = d(M\vec{r}_1, M\vec{r}_2) \tag{2.12}$$

$$\theta(\vec{r}_1, \vec{r}_2, \vec{r}_3) = \theta(\vec{r}_1 + \vec{o}, \vec{r}_2 + \vec{o}, \vec{r}_3 + \vec{o}) \tag{2.13}$$

$$\theta(\vec{r}_1, \vec{r}_2, \vec{r}_3) = \theta(M\vec{r}_1, M\vec{r}_2, M\vec{r}_3) \tag{2.14}$$

$$E = \sum_i E_i \tag{2.15}$$

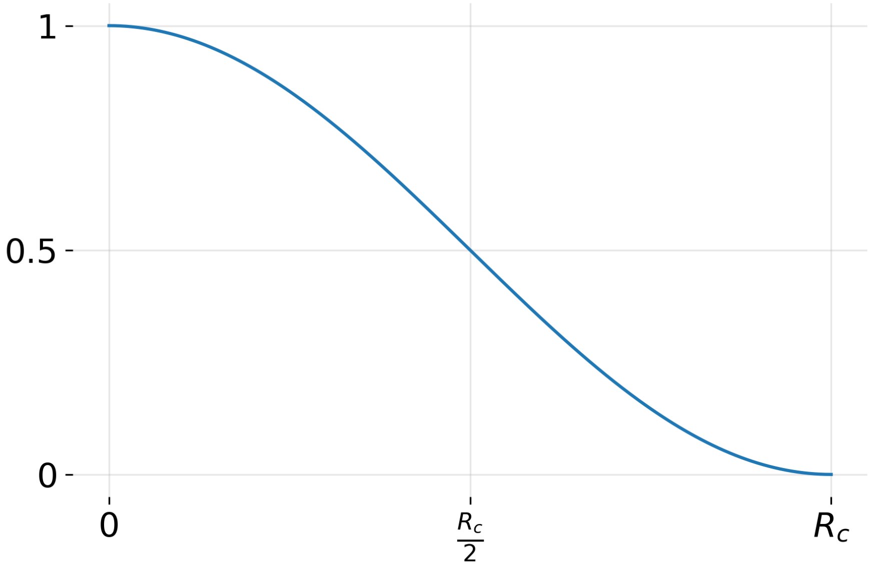
2.3.2 Cutoff

Fig. 2.14 and Eq. 2.16 show the cutoff function proposed by Behler and Parrinello in 2007. [45] It smoothly varies from one to zero over the domain bounded by the cutoff radius. Its purpose is to prevent step changes in the symmetry function values as atoms cross the cutoff radius.

$$f_c\Big(R_{ij}\Big) = \begin{cases} 0.5 \cdot \left[\cos\left(\frac{\pi R_{ij}}{R_c}\right) + 1\right] & \text{for} \quad R_{ij} \leq R_c \\ 0 & \text{for} \quad R_{ij} > R_c \end{cases} \tag{2.16}$$

$$\frac{\partial f_c\Big(R_{ij}\Big)}{\partial R_{ij}} = f_c'(R_{ij}) = \begin{cases} -\frac{\pi}{2R_c} \cdot \sin\left(\frac{\pi R_{ij}}{R_c}\right) & \text{for} \quad R_{ij} \leq R_c \\ 0 & \text{for} \quad R_{ij} > R_c \end{cases} \tag{2.17}$$

Figure 2.14: Behler-Parrinello cutoff function. [45]



The BP cutoff function possesses many critical mathematical properties required of well-behaved cutoff functions. It is:

1. smooth and everywhere first-order differentiable,
2. zero-valued at the cutoff radius (R_c), and

3. of zero-valued first-derivative at the origin and R_c.

2.3.3 Radial Basis

Figs. 2.15 and 2.16 and Eq. 2.18 show the radial "symmetry" function proposed by Behler and Parrinello in 2007. [45] The BP radial basis function is a Gaussian radial basis function (GRBF) combined with the BP cutoff. [73] It has three tunable parameters: the offset R_s which defines its maximum value, the decay constant η which defines how quickly it attenuates with distance from the maximum, and the cutoff radius R_c for the cutoff function. The BP radial basis also has a smooth first-order derivative.

$$G_i^2 = \sum_{j\neq i}^{N} e^{-\eta(R_{ij}-R_s)^2} f_c\Big(R_{ij}\Big) \tag{2.18}$$

$$\frac{\partial G_i^2}{\partial R_{ij}} = -2\eta(R_{ij}-R_s)e^{-\eta(R_{ij}-R_s)^2} f_c\Big(R_{ij}\Big) + e^{-\eta(R_{ij}-R_s)^2} f_c'\Big(R_{ij}\Big) \tag{2.19}$$

$$= \Bigg(-2\eta(R_{ij}-R_s) f_c\Big(R_{ij}\Big) + f_c'\Big(R_{ij}\Big)\Bigg) e^{-\eta(R_{ij}-R_s)^2} \tag{2.20}$$

Several potential configurations for the BP radial basis function are shown in Fig. 2.15. It is worth while noting that the maximum value of the BP Radial Basis is *not* the offset value, due to the action of the cutoff. Fig. 2.16, shows the basis functions from Fig. 2.15 (and one additional) normalized to the same value, and superimposed on them the underlying Gaussian radial basis function centered at the actual peak of the corresponding BP radial basis. Note the increasing skewness as R_s approaches the cutoff radius. The R_s values and corresponding maxima are listed below in Table 2.1. This is very likely not to be a severe complication, and indeed BP has been shown to be effective many times in many contexts. Nevertheless, this observation motivated the development of new basis functions that control symmetry or asymmetry explicitly to improve efficacy and suit specific purposes. This concept is further developed in this work and presented in Appendix C.

Figure 2.15: (l) The Behler-Parrinello radial basis function, shown here with $R_s = 0$ and several values of decay constant η. (r) BP radial basis with varied R_s and constant η.

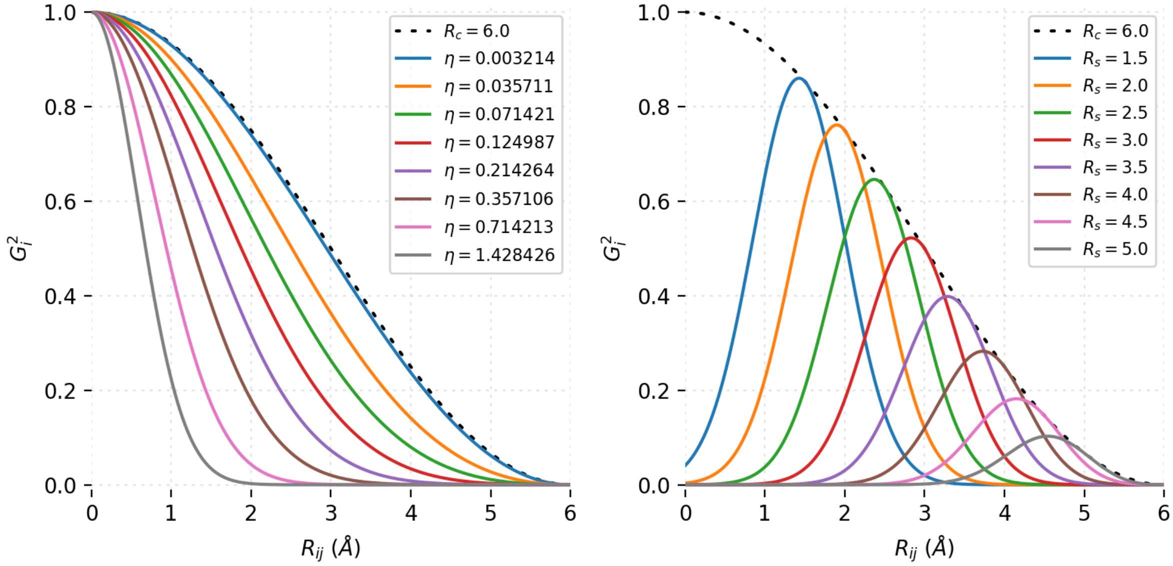
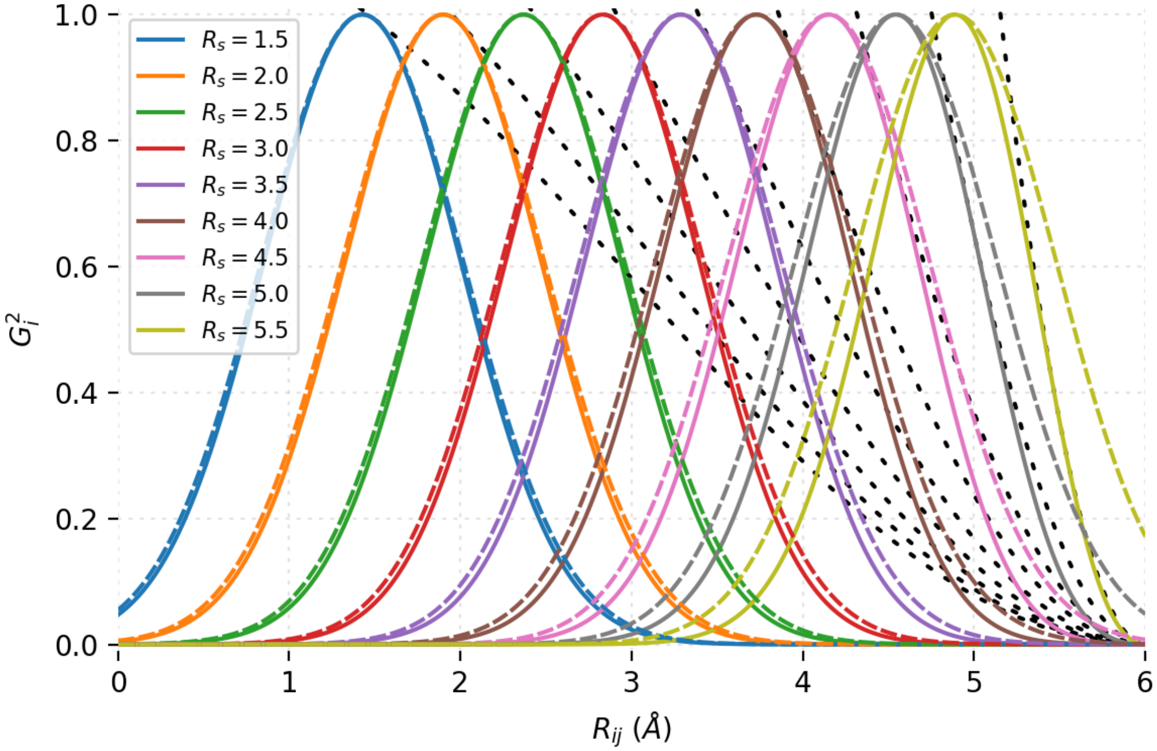


Figure 2.16: Normalized radial bases and GRBF overlaid to show increasing skewness as R_s approaches cutoff radius R_c.



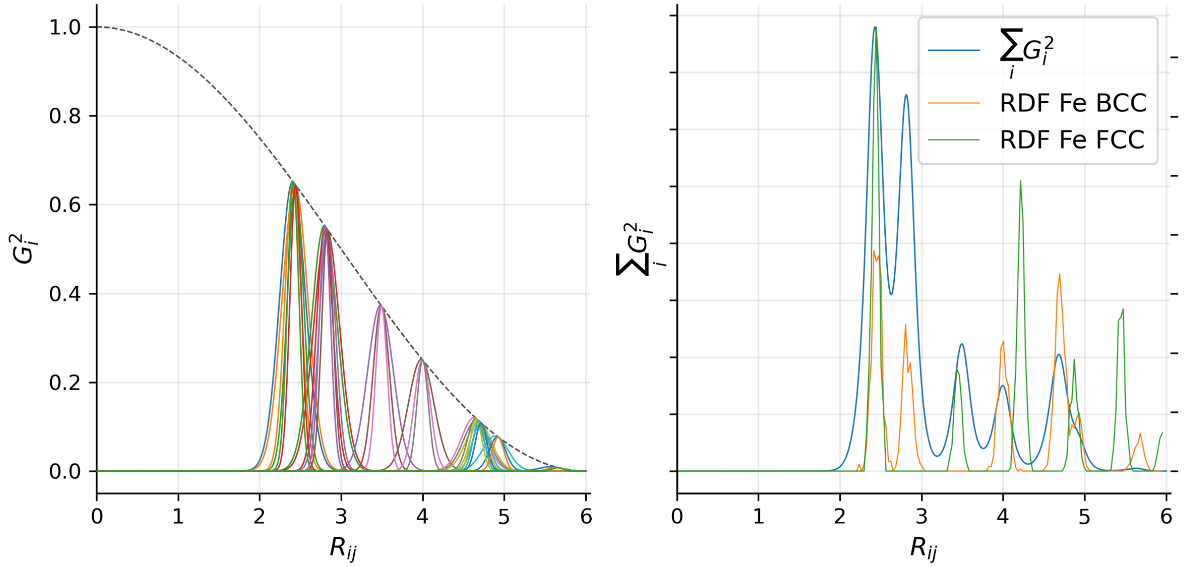
R_s	$\max(G_i^2)$
1.5	1.42745
2.0	1.89936
2.5	2.36834
3.0	2.83146
3.5	3.28578
4.0	3.72838
4.5	4.15046
5.0	4.54323
5.5	4.88911

Table 2.1: Table of G_i^2 maxima as a function of R_s.

2.3.4 Physical Interpretation of Radial Bases: Radial Distribution Functions and Bonner Spheres

Basis functions used in BP are like detectors for specific conditions. Each one asks, "are there atoms present with a particular relation to the central atom?" The η values can be adjusted to make a detector very narrow and precise, or very general. Fig. 2.17 shows a set of radial basis functions that is tuned to be maximally sensitive at the peaks of the radial distribution function (RDF) of BCC and FCC iron, essentially comprising basis set decomposition of those RDFs. This is the interpretation of the author, that the radial basis functions of BP and similar methods are detectors for specific relationships, and comprise basis set decompositions of radial, angular and spin aspects of atomic environments. This interpretation was used to inform the design of the spin basis functions presented in the Chapter on Methods.

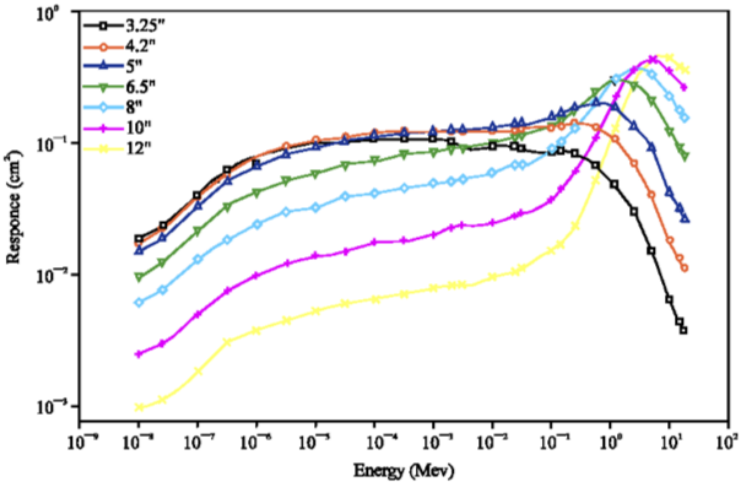
Figure 2.17: BP basis set and summation compared with BCC/FCC Fe RDF.



An analogous process to BP basis functions capturing information about atomic environments is the use of Bonner Spheres to measure neutron spectra. [74]

Measuring neutron spectra is a difficult process, requiring relatively sophisticated methods and thinking to get it all to work. Bonner spheres have varied moderator thicknesses and detectors at equal distance to the radiation source. This changes the shape of the neutron spectrum incident at each detector. This is critical because the detectors' readings are really integrated in energy to a scalar. By calibrating each Bonner Sphere using sources of known spectra and materials of known cross-section, it is possible to then use the spheres to decompose other spectra. As one adds more spheres of different thickness, the precision and resolution of the spectrum under test improves. Fig. 2.18 shows detector response as a function of incident neutron energy for a set of identical detectors inside different thickness Bonner Spheres.

Figure 2.18: Detector response to a set of Bonner Spheres as a function of incident neutron energy. [75]

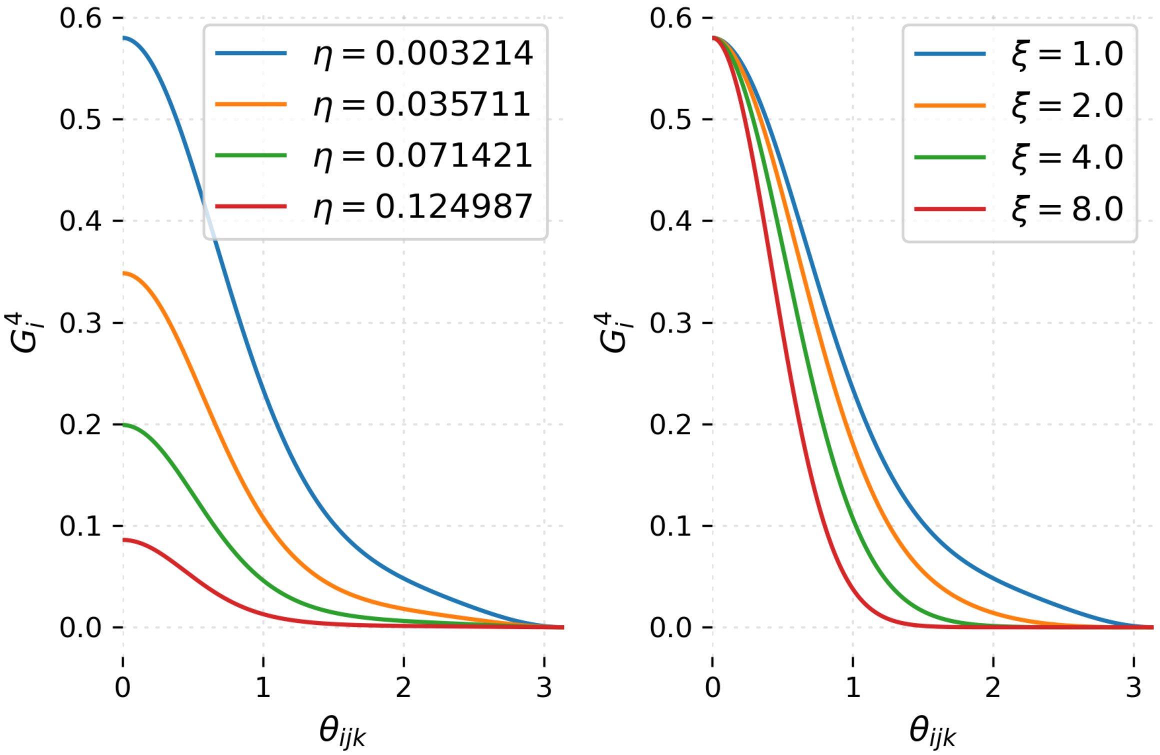


2.3.5 Angular Basis

Fig. 2.19 and Eq. 2.21 show the radial "symmetry" function proposed by Behler and Parrinello in 2007. [45] The BP angular bases are perhaps inspired in part by Stillinger-Weber potentials, which use a similar $\cos\theta_{ijk}$ term [76]:

$$G_i^4 = 2^{1-\xi} \sum_{j\neq i}^{N} \sum_{k\neq i,j}^{N} \Big[1 + \lambda \cos\theta_{ijk}\Big]^{\xi} e^{-\eta(R_{ij}^2+R_{ik}^2+R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \tag{2.21}$$

Figure 2.19: (l) The Behler-Parrinello angular basis function, shown here with all things equal except the decay constant η. (r) BP angular basis with varied ξ.



The BP angular basis also has a smooth first-order derivative. Since this basis has five terms, we first find the derivative of the angular term.

$$\alpha = \Big[1 + \lambda \cos\theta_{ijk}\Big]^{\xi} \qquad \frac{\partial\alpha}{\partial\cos\theta_{ijk}} = \lambda\xi\Big[1 + \lambda\cos\theta_{ijk}\Big]^{\xi-1} \tag{2.22}$$

$$\frac{\partial\alpha}{\partial\theta_{ijk}} = -\lambda\xi\sin\theta_{ijk}\Big[1 + \lambda\cos\theta_{ijk}\Big]^{\xi-1} \tag{2.23}$$

The radial term here is a little different than the G^2 radial term, however it is still straightforward.

$$\beta = e^{-\eta(R_{ij}^2+R_{ik}^2+R_{jk}^2)} \qquad \frac{\partial\beta}{\partial R_{ij}} = -2\eta R_{ij}\beta \tag{2.24}$$

$$\frac{\partial\beta}{\partial x_i} = \frac{\partial\beta}{\partial R_{ij}}\frac{\partial R_{ij}}{\partial x_i} + \frac{\partial\beta}{\partial R_{ik}}\frac{\partial R_{ik}}{\partial x_i} \tag{2.25}$$

$$\frac{\partial\beta}{\partial x_j} = \frac{\partial\beta}{\partial R_{ij}}\frac{\partial R_{ij}}{\partial x_j} + \frac{\partial\beta}{\partial R_{jk}}\frac{\partial R_{jk}}{\partial x_j} \tag{2.26}$$

$$\frac{\partial\beta}{\partial x_k} = \frac{\partial\beta}{\partial R_{ik}}\frac{\partial R_{ik}}{\partial x_k} + \frac{\partial\beta}{\partial R_{jk}}\frac{\partial R_{jk}}{\partial x_k} \tag{2.27}$$

The cutoff functions are differentiated as before.

$$\gamma = f_c(R_{ij}) \qquad \psi = f_c(R_{ik}) \qquad \chi = f_c(R_{jk}) \tag{2.28}$$

Having found derivatives of all of the individual terms, we now assemble the complete picture.

$$\frac{\partial G_i^4}{\partial x_i} = 2^{1-\xi}\sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_i}\beta\gamma\psi\chi + \alpha\frac{\partial\beta}{\partial x_i}\gamma\psi\chi + \alpha\beta\frac{\partial\gamma}{\partial x_i}\psi\chi + \alpha\beta\gamma\frac{\partial\psi}{\partial x_i}\chi + \cancelto{0}{\alpha\beta\gamma\psi\frac{\partial\chi}{\partial x_i}}\right] \tag{2.29}$$

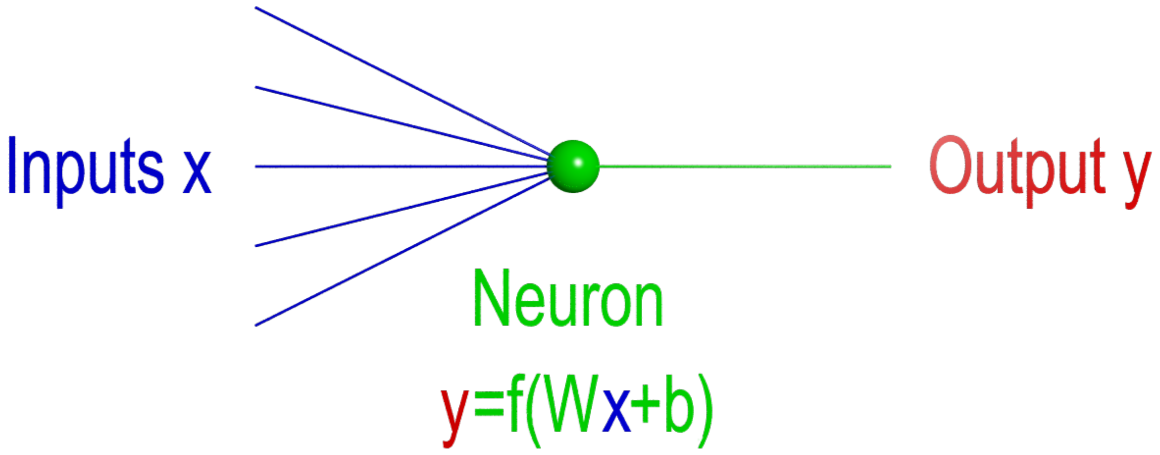
$$\frac{\partial G_i^4}{\partial x_j} = 2^{1-\xi}\sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_j}\beta\gamma\psi\chi + \alpha\frac{\partial\beta}{\partial x_j}\gamma\psi\chi + \alpha\beta\frac{\partial\gamma}{\partial x_j}\psi\chi + \cancelto{0}{\alpha\beta\gamma\frac{\partial\psi}{\partial x_j}}\chi + \alpha\beta\gamma\psi\frac{\partial\chi}{\partial x_j}\right] \tag{2.30}$$

$$\frac{\partial G_i^4}{\partial x_k} = 2^{1-\xi}\sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_k}\beta\gamma\psi\chi + \alpha\frac{\partial\beta}{\partial x_k}\gamma\psi\chi + \cancelto{0}{\alpha\beta\frac{\partial\gamma}{\partial x_k}}\psi\chi + \alpha\beta\gamma\frac{\partial\psi}{\partial x_k}\chi + \alpha\beta\gamma\psi\frac{\partial\chi}{\partial x_k}\right] \tag{2.31}$$

2.4 “Atomic Neural Networks”

A neural network is just a function; it accepts numbers and outputs numbers. The fundamental unit of a neural network is the “neuron.” As shown in Fig. 2.20, a neuron accepts a group of scalars, computes a weighted sum of those scalars, adds a bias, applies an “activation function,” and outputs the result. While it is conceivable to construct neurons with non-linear terms, the weighted sum neuron is general practice.

Figure 2.20: The “neuron.”



Activation functions can be any function. If multiple neurons are placed in series without non-linear activation, then the successive linear transforms could be expressed as one linear transform. Even an infinite series of linear transformations is the same as just one. Non-linear activation functions change that, making any series of neurons with non-linear activation a unique mathematical object. That said, NN regressors often use linearly-activated outputs, although some problems may benefit from non-linear activation. There are many commonly used activation functions:

$$\text{Linear:} \quad f(x) = x \tag{2.32}$$

$$\text{ReLU:} \quad f(x) = \begin{cases} 0 & x < 0 \\ x & \end{cases} \tag{2.33}$$

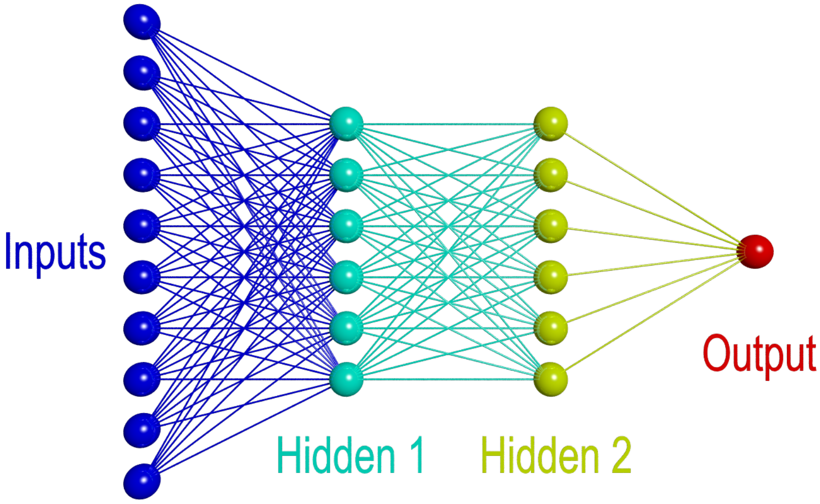
$$\text{Softplus:} \quad f(x) = \ln\left(1 + e^{x}\right) \tag{2.34}$$

$$f(x) = \tanh(x) \tag{2.35}$$

A “layer” refers to a set of neurons that all accept the same inputs, and a “hidden-layer” is one that the end user typically doesn’t see because it is neither input nor output. A "fully-connected" layer is one that accepts all of the inputs from the previous layer.

So, conventional BP uses two-hidden-layer, fully-connected NNs with single outputs, as shown in Fig. 2.21. In the world of ML, this is relatively simple—and it works.

Figure 2.21: Conventional Behler-Parrinello fully-connected, single-task bilayer.



The blue nodes are the inputs, cyan the first hidden layer, yellow the second hidden layer, and the red node is the output. The new work of this thesis departs from the "fully-connected" paradigm as shown in Chapter 3.

This fully connected bilayer architecture is used in nearly all BP and BP-based publications to date. [44, 45, 70, 71, 77–80]

2.5 Pre- and Post-processing

Pre- and post-processing of ML inputs and outputs condition data so that the subsequent ML process is more effective in terms of the accuracy or performance of the trained model and more efficient in terms of the training time to get there. They are generally regarded as essential, particularly for randomly initialized and iterative approaches such as linear regression using gradient descent and NNs. BP accomplishes this using the rescaling method according to their literature, however the methodology for pre- and post-processing is arbitrary and in theory any method is acceptable so long as it is reversible.

Rescaling shifts the maximum and minimum of a feature's distribution to arbitrary values and linearly interpolates for all data points between the extremes. The weakness of this approach is that the mean of the distribution for an input/output need not be zero, which tends to complicate NN learning as described above.

2.5.1 Feature Rescaling from -1 to 1

Since features are calculated on a per atom basis, the mean and standard deviation can be calculated in a straightforward manner. This methodology of rescaling is mentioned in BP literature, and implemented from -1 to 1 in `aenet`. [70, 71]

$$G_{\text{NN}} \in [-1, 1] \tag{2.36}$$

$$G_{\min} = \min(G_i) \qquad E_{\max} = \max(G_i) \tag{2.37}$$

$$G_{\text{NN}} = \frac{2}{G_{\max} - G_{\min}} \Bigg(G_i - G_{\min} \Bigg) - 1 \tag{2.38}$$

$$G_i = \frac{G_{\max} - G_{\min}}{2} \Bigg(G_{\text{NN}} + 1 \Bigg) + G_{\min} \tag{2.39}$$

2.5.2 Energy Rescaling from -1 to 1

$$\frac{E_{\text{NN}}}{n} \in [-1, 1] \tag{2.40}$$

$$E_{\text{min}} = \min\left(\frac{E_i}{n_i}\right) \qquad E_{\text{max}} = \max\left(\frac{E_i}{n_i}\right) \tag{2.41}$$

$$E_{\text{NN}} = n_i\left[\frac{2}{E_{\text{max}} - E_{\text{min}}}\left(\frac{E_i}{n_i} - E_{\text{min}}\right) - 1\right] \tag{2.42}$$

$$E_i = n_i\left[\frac{E_{\text{max}} - E_{\text{min}}}{2}\left(\frac{E_{\text{NN}}}{n_i} + 1\right) + E_{\text{min}}\right] \tag{2.43}$$

2.6 Energy Prediction and Force Calculation

The contribution of individual “atomic neural networks” are summed to produce the system’s potential energy:

$$E = \sum_i^N E_i \tag{2.44}$$

The gradient of energy with respect to spatial coordinates then is calculated using the chain rule.

$$\frac{\partial E}{\partial x_i} = \underbrace{\frac{\partial G}{\partial x_i}}_{\text{above}}\underbrace{\frac{\partial G_{\text{NN}}}{\partial G}}_{\text{below}}\underbrace{\frac{\partial E_{\text{NN}}}{\partial G_{\text{NN}}}}_{\text{NN code}}\underbrace{\frac{\partial E}{\partial E_{\text{NN}}}}_{\text{below}} \tag{2.45}$$

When using rescaling:

$$\frac{\partial G_{\text{NN}}}{\partial G} = \frac{2}{G_{\text{max}} - G_{\text{min}}} \tag{2.46}$$

$$\frac{\partial E}{\partial E_{\text{NN}}} = \frac{E_{\text{max}} - E_{\text{min}}}{2} \tag{2.47}$$

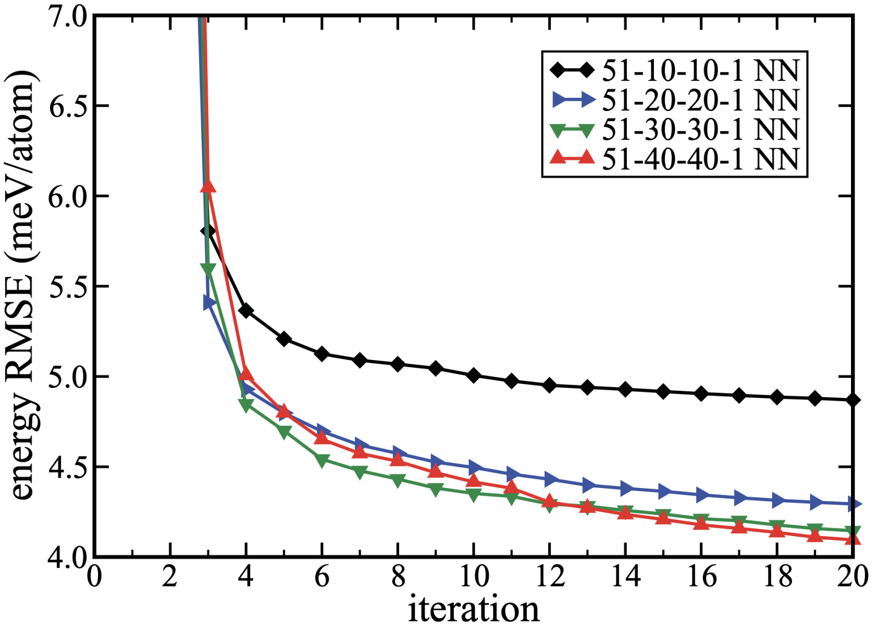
2.7 Application on Copper

BP is an excellent methodology for learning and expressing relationships between geometry and energy for non-magnetic materials. This is abundantly shown in the 2012 paper on Cu by Artrith and Behler. [78]

Approximately 38k data samples were calculated via DFT, including manifold configurations: bulk (FCC, BCC, SC, HCP), surfaces (FCC, BCC, HCP), clusters, bulk vacancies (FCC, BCC, SC) and surface vacancies (FCC, BCC). For each configuration type, many different box geometries and atom quantities were used.

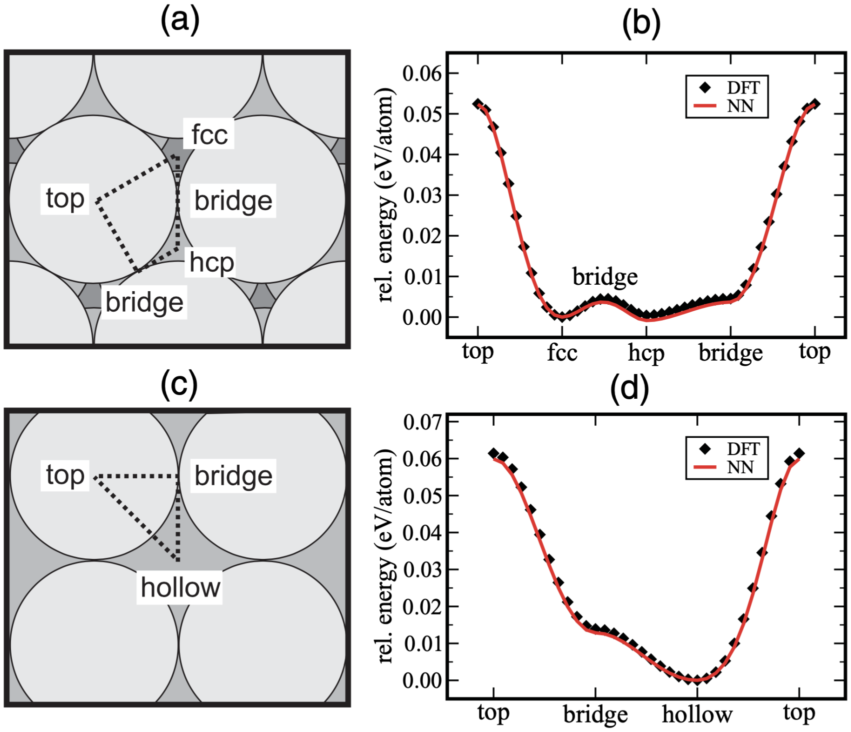
Their showcase featurization used 8 radial bases and 43 angular bases. This featurization was then used to train fully-connected bi-layers of various layer widths, results for 10, 20, 30, and 40 as seen in Fig. 2.22. The biggest improvement presented came from increasing each layer's width from 10 to 20 neurons, and their stated optimal fit came from using 30 neurons per layer. They report training and testing energy RMSEs of 3.6 and 3.9 meV/atom, respectively. They also trained on force data and offer force RMSEs of 42.8 and 42.0 meV/Bohr. Energy training and test MAEs were reported as 2.09 and 2.22 meV/atom, and force training and testing MAEs were reported as 29.3 and 29.4 meV/Bohr, respectively.

Figure 2.22: Training RMSE (meV/atom) vs training iteration for NNs of various sizes. [78]



Results for cohesive energy, lattice parameter, bulk modulus, unrelaxed and relaxed bulk vacancy formation energy, unrelaxed and relaxed surface energies, and unrelaxed and relaxed surface vacancy formation energies are presented. Bulk properties and surface energies tended to be within 1% of DFT, and vacancy formation energies tended to be within 5% (with outliers up to 10%). Energy over the course of an MD run and along several atomic surface migration paths were presented, showing good agreement with DFT as seen in Fig. 2.23. Forces in clusters and for surface defects were also shown to be in good agreement with DFT.

Figure 2.23: BP showing good agreement with DFT for atoms on various paths across Cu surfaces. [78]



These relevant, impactful results for non-magnetic materials show that BP is a valid starting point to build upon using spin bases that translate atomic spin environments into NN features.

Chapter 3

Methods

3.1 Novel Spin Bases

It remains an open question as to how best capture spin information using basis functions so as to fully articulate the physically relevant features of an atom's environment to a NN. A more realistic assessment is that the featurization of an atomic environment is like a "lossy compression" in which some information is lost, but hopefully not enough to harm the application. The first attempt in this work was to implement a basic framework quickly and assess it for effectiveness. Following that, the framework was expanded to use multiple bases of each type with varied powers of spin terms and cutoff radii. Then, the spin bases were wrapped in Gaussian radial basis functions.

3.1.1 Spin Bases 000

The initial approach to formulating spin bases was loosely based on physics relations for energetics of and interactions between magnetic fields. Only one spin base of each type was used in this initial, exploratory approach. The intent was to compare using these spin bases against not using them and see the difference in terms of RMSE of system energy prediction.

The spin bases are as follows:

$$S_i^1 = \vec{S}_i \cdot \vec{S}_i \tag{3.1}$$

$$S_i^2 = \sum_{j \neq i}^{N} \left[\vec{S}_i \cdot \vec{S}_j \right] f_c(R_{ij}) \tag{3.2}$$

$$S_i^3 = \sum_{j \neq i}^{N} \left[(\vec{S}_i \times \vec{S}_j) \cdot \mathbf{\hat{r}_{ij}} \right] f_c(R_{ij}) \tag{3.3}$$

$$S_i^4 = \sum_{j \neq i}^{N} \left[(\vec{S}_i \cdot \mathbf{\hat{r}_{ij}})(\vec{S}_j \cdot \mathbf{\hat{r}_{ij}}) \right] f_c(R_{ij}) \tag{3.4}$$

Information about the radial proximity of spins to the central atom will be incorporated via the cutoff function for all cases except self-spin.

The first spin basis, S_i^1, is inspired the relations for the energy stored in a magnetic field:

$$E = \frac{1}{2\mu_0} B^2 \tag{3.5}$$

The second spin basis, S_i^2, is inspired by two related phenomena: (1) the energy of interaction between a magnetic moment and a magnetic field, and (2) the Heisenberg spin Hamiltonian used by the SPILADY and LAMMPS SPIN packages as referred to in Eq. 1.8:

$$E = -m \cdot B \tag{3.6}$$

$$\mathscr{H}_{spin} = -\frac{1}{2} \sum_{i,j} J_{ij}(\vec{R}) \vec{S}_i \cdot \vec{S}_j \tag{3.7}$$

The spin basis S_i^2 causes the NN to subsume the exchange function, eliminating the need to assign a fixed form to it thus bypassing a major weakness of fitted exchange SLD approaches. This makes the NN SLD approach flexible and adaptable to any context, for example for point-defects, surfaces, clusters, and ambiguously defined atomistic scenarios.

S_i^2 can also be expressed as follows by factoring $\vec{S}_i$ out of the sum:

$$S_i^2 = \vec{S}_i \cdot \sum_{j \neq i}^{N} \vec{S}_j f_c(R_{ij}) \tag{3.8}$$

The third spin basis, S_i^3, is based on the antisymmetric exchange spin Hamiltonian term (Dzyaloshinskii-Moriya interaction) shown in Eq. 3.9. [81–83] This is relevant in spin-spiral systems and magnetic skyrmions. [84–86]

$$\sum_{i,j,i \neq j}^{N} \vec{D}_{ij} \cdot (\vec{S}_i \times \vec{S}_j) \tag{3.9}$$

Lastly, S_i^4 is inspired by magnetic dipole interactions. [87]

$$\mathscr{H}_{dip} = \sum_{i,j,i \neq j} \frac{g_i g_j}{r_{ij}^3} \Big((\vec{S}_i \cdot \hat{R}_{ij})(\vec{S}_j \cdot \hat{R}_{ij}) - \frac{1}{3}(\vec{S}_i \cdot \vec{S}_j) \Big) \tag{3.10}$$

3.1.2 Spin Bases 000a, Varied Cutoffs and Self-Spin

Following the initial success of Spin Bases 000, the next logical step was to scale up and expand the framework by using multiple spatial cutoffs, and multiple powers of self-spin drawing inspiration from the Heisenberg-Landau spin Hamiltonian used by Ma and Dudarev in SPILADY to relate energy to atomic magnetic moments thus allowing for spin magnitude to be a degree of freedom in their simulations shown in Eq. 3.11. Again, the Spin Bases 000 setup was using exactly one basis of each type as a *prima facie* test of the spin bases.

$$\mathscr{H}_{H-L} = A_i(\vec{R})S_i^2 + B_i(\vec{R})S_i^4 + C_i(\vec{R})S_i^6 + D_i(\vec{R})S_i^8 \tag{3.11}$$

Results for this setup are not explicitly presented in this work, however the purpose of the Spin Bases 000a was to further explore the space spanned by those basis forms and their efficacy in reducing energy prediction error compared to BP.

3.1.3 Spin Bases 000b, significantly more expressive

Spin Bases 000 also used spin-squared terms only, and no other powers—which is physically questionable. There are other powered relationships between spin and energy and the following were postulated to more fully explore that space. An additional spin basis, S_i^5, was also conceived of and experimented with

$$S_i^{1a} = \left[\vec{S}_i \cdot \vec{S}_i\right]^{\xi} \tag{3.12}$$

$$S_i^2 = \sum_{j \neq i}^{N} \left[\vec{S}_i \cdot \vec{S}_j\right]^{\xi} f_c(R_{ij}) \tag{3.13}$$

$$S_i^3 = \sum_{j \neq i}^{N} \left[(\vec{S}_i \times \vec{S}_j) \cdot \mathbf{\hat{r}_{ij}}\right]^{\xi} f_c(R_{ij}) \tag{3.14}$$

$$S_i^4 = \sum_{j \neq i}^{N} \left[(\vec{S}_i \cdot \mathbf{\hat{r}_{ij}})(\vec{S}_j \cdot \mathbf{\hat{r}_{ij}})\right]^{\xi} f_c(R_{ij}) \tag{3.15}$$

$$S_i^5 = \sum_{j \neq i}^{N} \left[(\vec{S}_i \times \mathbf{\hat{r}_{ij}}) \cdot (\vec{S}_j \times \mathbf{\hat{r}_{ij}})\right]^{\xi} f_c(R_{ij}) \tag{3.16}$$

3.1.4 Spin Bases 001

One common practice in ML featurization is to transform features using Gaussian radial basis functions. [88] This approach is used here to transform the above defined bases in a manner hopefully more suitable in mathematical form to expressing relationships between geometry, spin and energy using NN regressors. Interpreted as detectors, these Gaussian radial basis function spin bases are highly tunable and can be sensitive to only specific cases or be used to cast a broad net with much overlap between bases. An additional intent is to be absolved of any responsibility for setting powers of spins in the manner described for

Spin Bases 000b, since the decay constant η can achieve a similar effect.

$$S_i^1 = e^{-\eta_1((\vec{S}_i \cdot \vec{S}_i)^{\frac{\xi}{2}} - \alpha_1)^2} \tag{3.17}$$

$$S_i^2 = \sum_{j \neq i}^{N} e^{-\eta_2(\vec{S}_i \cdot \vec{S}_j - \alpha_2)^2} f_c(R_{ij}) \tag{3.18}$$

$$S_i^3 = \sum_{j \neq i}^{N} e^{-\eta_3((\vec{S}_i \times \vec{S}_j) \cdot \mathbf{\hat{r}_{ij}} - \alpha_3)^2} f_c(R_{ij}) \tag{3.19}$$

$$S_i^4 = \sum_{j \neq i}^{N} e^{-\eta_4((\vec{S}_i \cdot \mathbf{\hat{r}_{ij}})(\vec{S}_j \cdot \mathbf{\hat{r}_{ij}}) - \alpha_4)^2} f_c(R_{ij}) \tag{3.20}$$

These are the bases used to generate what is presented in the Chapter on Results.

3.1.5 Modified Rescaling

This work presents a new data pre- and post-precessing concept called simply "modified rescaling." This is different from regularization, which is an approach that allows or improves predictions for underdetermined data (more features than samples), and limits overfitting in cases where the fitting function is too complex relative to the underlying functional form. In modified rescaling, feature distributions are adjusted to have a zero mean, and a unity 1D distance limit from the mean. That means dividing the zero shifted data by a scaling factor set to the maximum absolute value difference between a data value and the data mean.

Note, while the current work has implemented this to pre- and post-process NN inputs (features), it has not been implemented for NN outputs (energy).

$$G_{\text{NN}} \in [-1, 1] \tag{3.21}$$

$$\mu_G = \frac{1}{\sum_{i=1}^{M} N_i} \sum_{i=1}^{M} \sum_{j=1}^{N_i} G_{i,j} \tag{3.22}$$

$$G_d = \max(|G_i - \mu_G|) \tag{3.23}$$

$$G_{\text{NN}} = \frac{G_i - \mu_G}{G_d} \quad G_i = G_d G_{\text{NN}} + \mu_G \tag{3.24}$$

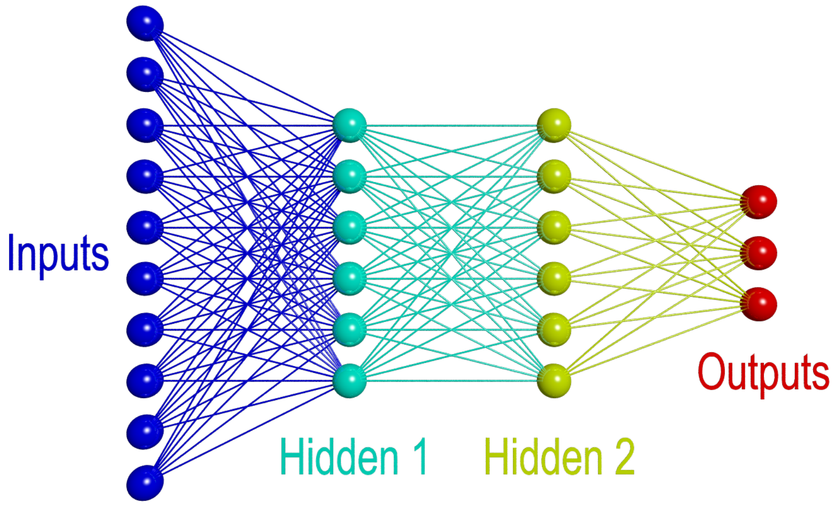
3.2 Novel Neural Network Architectures

Neural Networks have been catapulted to prominence by the paper on ImageNet in 2012 using deep CNNs. [89] This outstanding success ushered in a new era in science and brought the phrase "Big Data" into common parlance.

3.2.1 The Multi-Task Network vs Multi-Network Task

The original BP methodology was generalized to multiple elements by expanding featurizations in various ways—and using a separate NN for each element. This arrangement allows for element-specific featurizations, but does not allow for any cross-talk or transfer learning between the isolated NNs. NN literature from principally CS/ML focused investigators does however offer what they call a "multi-task (MT) neural network" (MTNN). As shown in Fig. 3.1, MT learners use multiple output nodes, one for each task, with a common set of inputs and main layers.

Figure 3.1: Fully-connected, multi-task bilayer.



Application of MT architectures was the very first modification the work of this thesis made to BP. Multiple MT BP setups were conceived of and implemented by the author in 2019, as documented by commits to the author's github. This architecture was arrived at independently and published by another group in 2020. [90]

The key to using an MTNN with BP features to treat multi-component systems is twofold:

1. The output used for a given atom is determined by its elemental identity,
2. The featurization must be the same for all elements. (Element is used broadly and could mean isotope if necessary.)

Radial and angular feature sets can be expanded in many ways, although this is not required. The only requirement is that the featurization for each atom must be the same. There is ambiguity allowed in that the central atom is only identified by selecting an output. So, one could construct radial terms that are independent of the neighbor atom identity, *and* ones for each specific neighbor identity. This is similar to using a total RDF and partial RDFs in materials characterization. An analogous process can treat angular terms.

Early work pursuant to this thesis deployed MTNNs for two ternary potentials: HON, and FeNiCr. In the case of HON, the earliest version used neighbor identity-agnostic basis functions and a later one used different neighbor-ID specific "partial bases." Those efforts were discontinued in favor or pursuing better basis functions in the context of pure iron.

Another potential use of MTNNs is to predict per atom quantities other than energetic contributions. For example, one early BP work used two separate NNs to predict atomic net charge and "short-range" energy contributions to isolate electrostatic interactions. [77] Early work of this thesis used a simliar approach implemented with an MTNN for HON systems, where one output predictied partial charge and the other predicted "short-range" energy. This approach was ultimately deemed to be unnecessarily complex since the conventional BP approach performed just as well for that target case.

3.2.2 Principal Component Weight Initialization

What if one were to initialize the weights of several nodes of a NN to the first few principal components (PCs) describing the vast majority of the variance of the dataset? Would that save time or achieve a better result?

Several different approaches were tried:

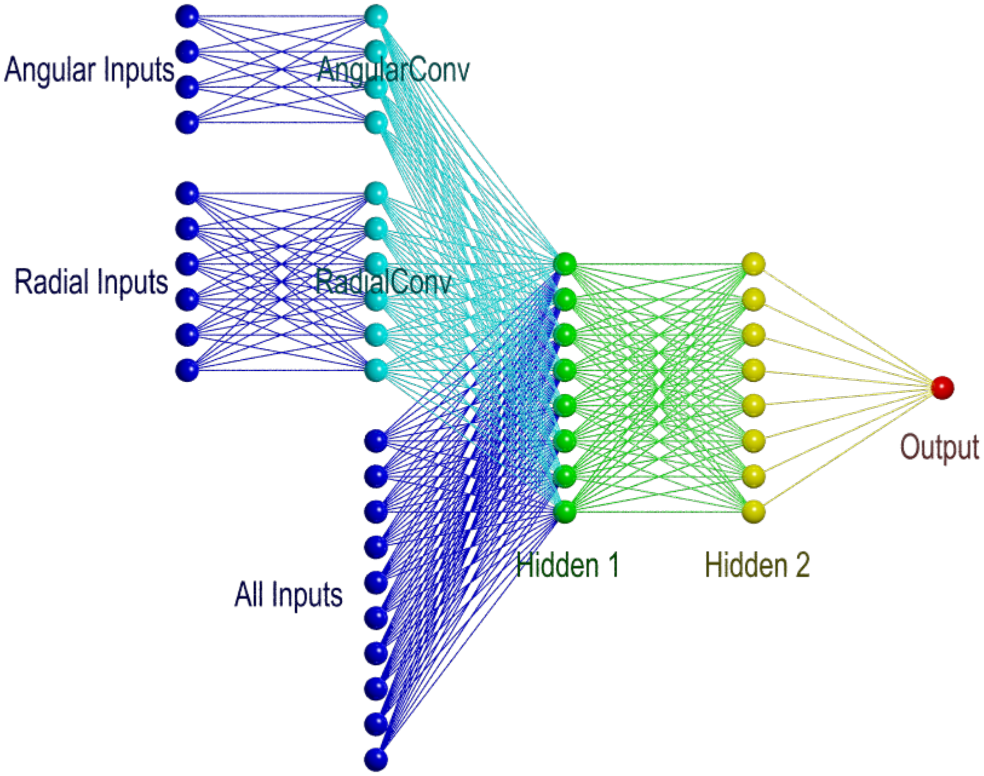
- Initialize a set of first-hidden layer weights only.
- Initialize a set of first-hidden layer weights, and the convolutional layers separately with PCs over their bases. (Convolutional later definitions presented later in this Section.)

Initial efforts used all available PCs, and it is now suspected that the low-variance-explaining PCs were probably not meaningful. Randomly initialized neurons for all but the high-variance explaining PCs is the procedure recommended for future work.

3.2.3 Convolutions by Basis Class

In BP, the angular and radial information are input all at once; every neuron of the first hidden-layer receives each basis function regardless of class (radial or angular). The new “convolutional” NN (CNN) of this work uses class-specific “convolutional” layers that accept all of the inputs of a given class (radial, angular, or spin). The outputs of the class-specific convolutional layers *and* the complete set of basis function values are then fed into the first fully-connected hidden layer, and downstream of that proceeds in the same manner as the BP NN. Fig. 3.2 is a diagram of the new CNN. The top set of blue and cyan nodes represent a first class, for angular bases in this example. The angular basis function inputs go to the angular convolution layer, and the outputs of that layer then tie into the fully-connected main setup—which still receives the complete set of bases as before, although this is now in addition to the convolutional outputs.

Figure 3.2: Basis Class-Grouped Convolutional NN.



The guiding concept is that knowledge of a dataset's structure can be exploited by controlling the connectivity between neuron groups within the NN, as is done with convolutional "filters" in image processing. [88] It is postulated that information from proximal pixels is more important than from distal ones. A filter that only sees a group of pixels near a pixel under evaluation can do things like detect edges or gradients in particular directions, and then subsequent layers can construct higher order objects until ultimately the algorithm can distinguish between cats, dogs, and airplanes. This outperforms a fully-connected setup that simply gets all pixel values dumped into it at once. Instead of pixels, the current application is concerned with basis functions, and grouping them by type is the first connectivity-limited architecture of this work.

3.2.4 Convolutions by Basis Sub-Class

The next innovation beyond the class-wise 1D convolutions is convolution within sub-classes. In this scheme, the radial, angular, and spin classes are broken down further. The radial class is broken down into groups of common decays across offsets and groups of common offset across decays. Similarly, the angular class is broken down into common decays across exponents and common exponents across decays. The spin class is broken down into the spin sub-classes, S_i^1 through S_i^4.

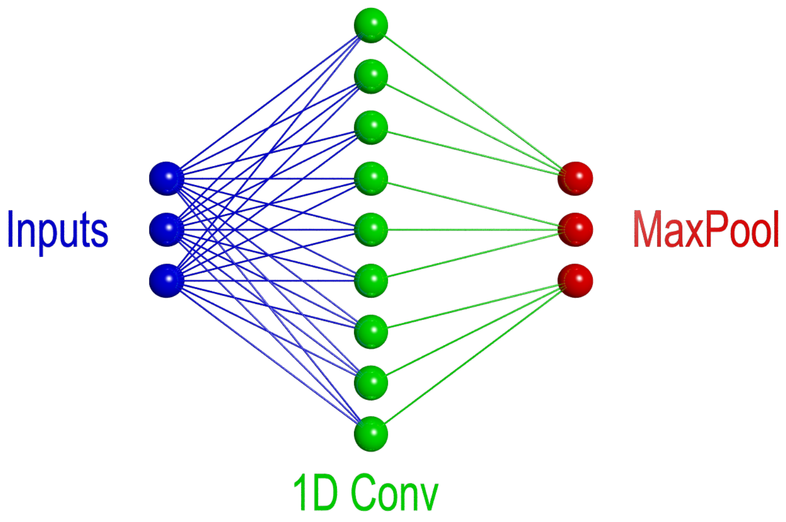
This NN architecture significantly increased computation time, yet did not yield meaningful reduction in energy prediction errors.

3.2.5 Expansions of Convolutions

The CNN architectures proposed above are only the simplest possible applications of their design concept. In ML literature, it is common to see “max pooling” or other pooling layers following CNN filter layers. Deep CNNs generally apply many iterations of: CNN, max pool, and activation.

Pooling functions could offer gains beyond those of the basic forms described above. Fig. 3.3 shows the basic setup for pooling sections implemented in this work. In this diagram, inputs are fed into a 1D convolutional layer and each of the subsequent pooling neurons accepts inputs from three of the 1D convolutional neurons. There is no overlap; each 1D convolutional neuron’s output goes to exactly one of the max pooling neurons. While the label used is “MaxPool,” and that was the one experimented with, other pooling functions could be used.

Figure 3.3: Pooling Network Architecture Subsection



Many different architectures are possible by combining layers in different orders. For example, conventional ML CNNs often iterate with a CNN → max pool → activation pattern. As to what operates best in any given case, we do not generally know. We only have knowledge of what has worked best in the past.

In this work many combinations were tried: CNN → activation → CNN → activation, CNN → max pool → activation, CNN → activation → CNN → max pool → activation, CNN → max pool → activation → CNN → activation, and so on. It was found that they all dramatically increase training time, but provided limited or no benefit. That result is context dependent, so other applications may find benefits that outweigh costs. These new evolved CNN approaches also all significantly increase the complexity of their C/C++ implementation required for MD.

3.2.6 Trainable Parameters Per Network Input (TPI)

Using equivalent geometric bases, the spin-aware NN IP approach of SANNIP has more trainable parameters than BP, because of the spin bases. However, if the downstream architecture remains the same then SANNIP actually has fewer trainable parameters per input (TPI) than BP. This relation for the conventional bilayer is shown below:

$$\mathrm{TPI}(i) = \frac{(i+1)h_1 + (h_1+1)h_2 + (h_2+1)}{i} \tag{3.25}$$

$$\lim_{i\to 0} \mathrm{TPI}(i) = \infty \quad \text{and} \quad \lim_{i\to\infty} \mathrm{TPI}(i) = 1 \tag{3.26}$$

That means that if SANNIP achieves better performance: (1) on the same data, (2) using the same geometric bases, and (3) using a conventional bilayer network, then it is gleaning more information per basis than BP.

For the new CNN, as implemented, the number of trainable parameters in the spin convolution layer increases as the square of the number of spin inputs:

$$\mathrm{TPI}_{\mathrm{BP}}(i) = \frac{(i_r+1)i_r + (i_a+1)i_a + (i+1)h_1 + (h_1+1)h_2 + (h_2+1)}{i} \tag{3.27}$$

$$\mathrm{TPI}_{\mathrm{SANNIP}}(i) = \frac{(i_s+1)i_s + (i_r+1)i_r + (i_a+1)i_a + (i+1)h_1 + (h_1+1)h_2 + (h_2+1)}{i} \tag{3.28}$$

The Table 3.1 shows the TPI for all cases presented in this work.

TPI	BP	SANNIP
Bilayer	41.7	36.3
CNN	64.1	58.3

Table 3.1: Trainable parameters per input for bilayer NN and CNN for BP and SANNIP.

3.3 Spin-Aware Dataset Generation

The ability of Behler-Parrinello (BP) NNIP methods to learn and express $\mathscr{H}(\vec{r})$ is well established, in terms of prediction of: system energy and atomic forces, lattice constants, bulk modulus, bulk vacancy formation energies, surface energies, surface vacancy formation energies, etc. It is therefore herein taken for granted that the geometric degrees of freedom in metallic systems can be well treated.

When generating a dataset intended to be used to fit or train an IP, it is important to consider the specific aims of that potential. The intent of this work is POC of a methodology for a magnetic moment / spin-aware, neural network-based interatomic potential (SANNIP) that can directly learn and express relationships between geometry, spin, and energy. Given the vastness of the sample space of $\mathscr{H}(\vec{r}, \vec{s})$, the scope of this work is constrained to methodology validation—and leaves high-fidelity, dense sampling of that space for other purposes to future work. The principal concern here is to span a meaningful space of geometric configurations and spin configurations to lend validity to evaluations of numerical accuracy, and support impactful modeling applications that explicitly require a spin-aware approach.

As mentioned in the Background Chapter, NN IPs require extra sampling in the vicinity of local minima, therefore the ferromagnetic states for each geometric configuration must be sampled. Additionally, sampling of spin states must include a gradual and smooth increase in the magnitude of perturbations from the FM ground state. It is also important to consider higher-energy spin configurations for iron such as paramagnetic and anti-ferromagnetic states because without explicit knowledge of those states the NN might under-predict the energy of such configurations. The situation regarding derivatives (forces and updates to spin) in the domain of interest is also improved by explicit knowledge of higher energy spin states.

3.3.1 Methodology Overview

Successful POC therefore depends on broad sampling in both $\vec{r}$ and $\vec{s}$. $\vec{r}$ must be sufficiently covered to show that SANNIP compares favorably to BP across a broad range of geometries. Sampling just $\vec{r}$ implies some standardized condition for $\vec{s}$, which should generally be the global spin ground state. For Fe at 0K this is perfect FM alignment.

Varied initial magnetic moment inputs to DFT codes have been used to obtain a distribution of $\vec{s}$ for each $\vec{r}$. This reduced calculation time by at least one order of magnitude in comparison to "constrained magnetism" DFT calculations which append a penalty term to the objective function under minimization (total system energy). The constrained magnetism penalty term can often be non-trivial, adding significant difficulty to taming that approach.

Having obtained a broad dataset, training can be executed on subsets with differing character to see how SANNIP does or does not improve in various situations compared to BP. For example, training can be executed on a data-subset comprised of the FM subset for each $\vec{r}$. Similarly, a data-subset can be constructed from all $\vec{s}$ for a given $\vec{r}$.

3.3.2 Geometry Specification

Each sample in the dataset of this thesis began with specification of each atom's Cartesian coordinates. While there are many methods for producing such coordinates, here, conventional EAM potentials and MD ensembles as implemented in LAMMPS have been used. Using NPT and NVT ensembles mitigates bias introduced by sampling at only a handful of lattice parameters (or system sizes). These ensembles also provide a random, physically-relevant distribution of atomic position perturbations from equilibrium lattice sites. Geometric stochasticity of this form is critical for two key reasons: (1) to avoid overfitting by including lots of small-scale variations alongside larger-scale variations, (2) so that the error in DFT calculations themselves can be normally distributed allowing invocation of the Central Limit Theorem, and (3) error caused by variation in k-spacing may be more easily averaged resulting in a smoother potential.

3.3.3 Spin Specification

Ferromagnetic alignment is the ground state for BCC iron below its Curie Temperature of 1043 K, and this is the first spin configuration calculated for each geometry. A primary interest here is establishing the FM ground state for each geometry, and then evaluating higher-energy spin states. Again, this is critical for both correct prediction of ground states by NNs *and* reasonable derivatives (forces).

Random variables in polar angle and azimuth are generated and then converted to cartesian coordinates to sample spin-space as perturbations from FM. Polar angle is sampled as the inverse cosine the absolute value of a random draw from a normal distribution $\mathcal{N}(0, \sigma)$. Azimuthal angle is sampled from a uniform distribution $\mathcal{U}[0, 2\pi)$.

$$\Theta = \arccos(|\mathcal{N}(0, \sigma)|) \tag{3.29}$$

$$\Phi = \mathcal{U}[0, 2\pi) \tag{3.30}$$

$$\sigma \in \{0, 0.015, 0.05, 0.1, 0.2\} \tag{3.31}$$

To sample paramagnetic (PM) states, uniform random sampling of azimuth is again used, however sampling of polar angle is changed to the following:

$$\Theta = \arccos(1 - \mathcal{U}[0, 2)) \tag{3.32}$$

Finally, a variety of anti-ferromagnetic (AFM) spin configurations are sampled by using different orders of opposite spin in each geometric configuration. Since LAMMPS was used to generate geometries and LAMMPS can make arbitrary changes to the ordinal labels of atoms, it turned out not to be the case that spin configurations uniformly involved adjacent atoms having opposite spin inside each unit cell, or between adjacent unit cells—however it is the case that each system's magnetic moment initialization has a net value of zero.

3.3.4 DFT Workflow

DFT uses a charge distribution and a set of wavefunctions to calculate energy, and iteratively updates both the charge distribution and the wavefunctions until the system energy value change is less than a threshold value (the convergence critireon). (It is possible to fix the charge distribution in DFT, but this is not useful in the present context.). The spin configuration of atoms is dependent on the electronic structure, specifically which electrons are spin up vs spin down in what orbital and how those orbitals are shaped and overlapped.

The VASP workflow for generating data for training this work's spin-aware potential is as follows:

1. Execute a non-spin polarized calculation with a 24 electronic step limit, and save the charge distribution.
2. Load the prior charge distribution, initialize atomic magnetic moments to the desired state, execute a spin-polarized calculation with a 24 electronic step limit, and save the charge distribution.
3. Load the prior charge distribution, initialize atomic magnetic moments to the desired state, execute a spin-polarized calculation with a 300 electronic step limit, and save the charge distribution.
4. Load the prior charge distribution, initialize atomic magnetic moments to the desired state, execute a spin-polarized calculation with a 300 electronic step limit. Done.

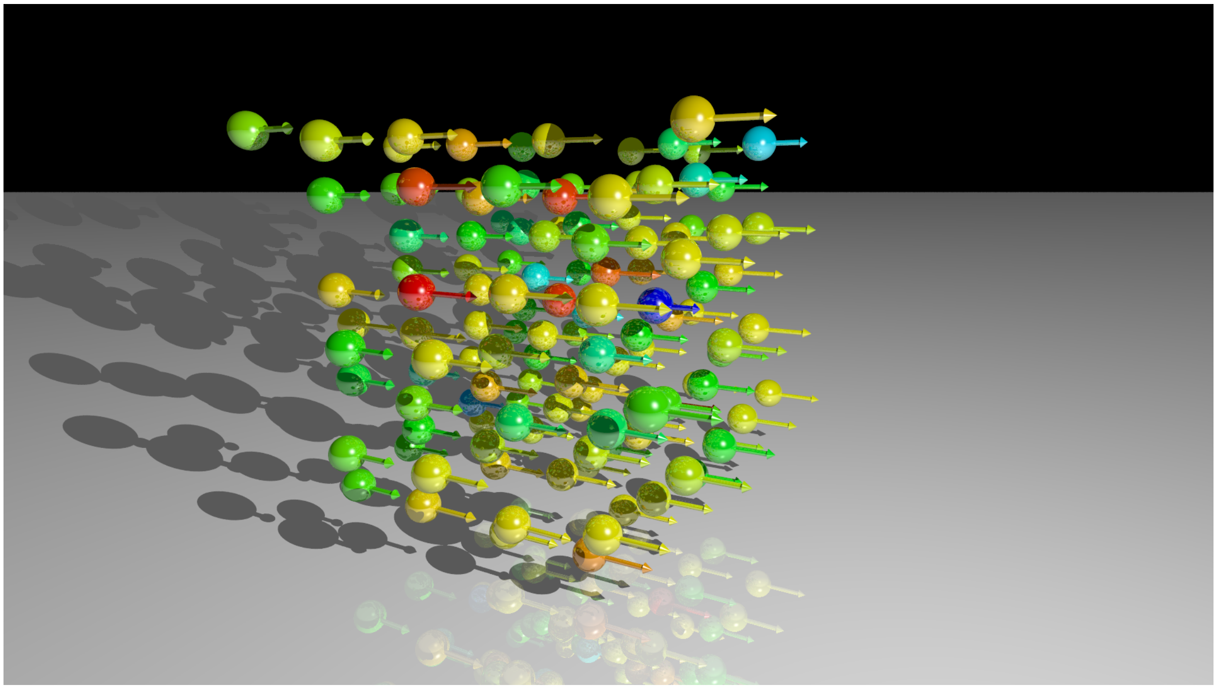
This multi-run workflow was necessary to both: (1) improve the quantity of calculations that complete (vs fail to reach the energy convergence criterion prior to exhausting all allowed electronic steps), and (2) improve the similarity between the desired spin configuration and the final result.

At each DFT execution, the magnetic moments are initialized to the desired values again, while wavefunction parameters are randomly initialized.

3.3.5 “Tour de Fer”

Perhaps the most significant subset of this dataset is dubbed the ”Tour de Fer.” It contains are seven geometric configurations, and each geometric configuration is itself subjected to seven different MD command sequences. The purpose is to quickly survey a broad range of geometric configurations and present a very significant challenge to any fitting or learning algorithm because it includes many defect types, phase changes, and creation of damage by fracture, compression, and extreme shear. Fig. 3.4 shows an example of a 4x4x4 BCC, FM-aligned di-vacancy configuration. The colors of each atom are representative of their spin magnitude, with blue being low magnitude and red being high.

Figure 3.4: “Tour de Fer” configuration example.



3.3.5.1 “Tour de Fer,” Geometric Configurations

It is important to include a wide range of bulk and point-defect configurations to support the claim that the POC evaluation of numerical accuracy is on hard-to-fit data, robust and conclusive. Full lattice γ-Fe is included since many of the other herein presented data subsets pass through its temperature range (1185-1667 K).

1. 3x3x3 BCC, full lattice
2. 4x4x4 BCC, mono-vancancy
3. 4x4x4 BCC, di-vancancy
4. 4x4x4 BCC, SIA 110
5. 4x4x4 BCC, SIA 111
6. 4x4x4 BCC, SIA 100
7. 3x3x3 FCC, full lattice

3.3.5.2 "Tour de Fer," MD Evolutions

The seven MD evolutions below were chosen to quickly span a wide range of extreme distortions. The intent of doing this was to encompass phenomenology that could significantly challenge any IP fitting methodology, again supporting the validity of our POC numerical accuracy evaluation. Phase change, fracture, slip, and damage accumulation are exhibited in the systems under test.

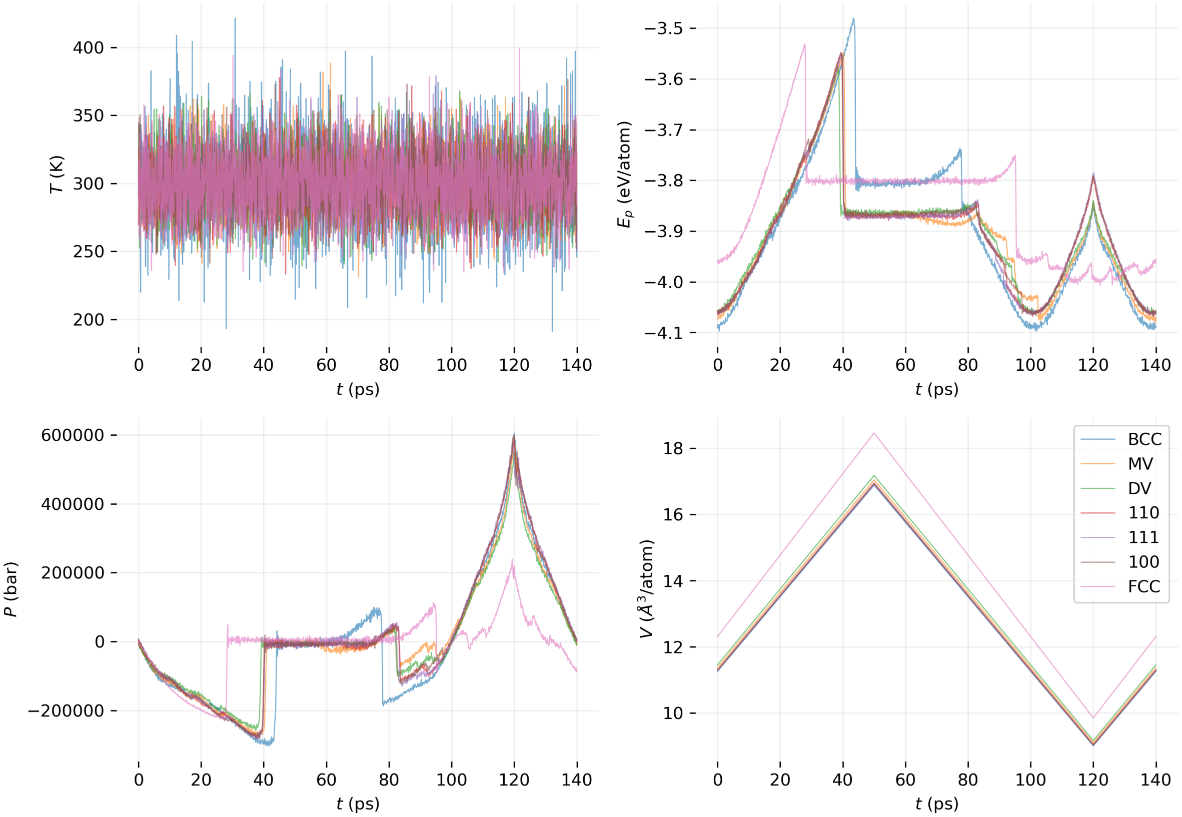
1. NPT, 300K → 2500K → 300K
2. NPT, 300K → 3500K → 300K
3. NVT, 300K, eq. → tension to 1.5 → compression to 0.8, [100] crystallographic direction
4. NVT, 300K, eq. → tension to 1.5 → compression to 0.8, [210] crystallographic direction
5. NVT, 300K, eq. → shear to 0.3 → eq., [100] crystallographic direction
6. NVT, 300K, eq. → shear to 0.3 → eq., [110] crystallographic direction
7. NVT, 300K, eq. → shear to 0.3 → eq., [210] crystallographic direction

Future work can build upon this by including deformations in more crystallographic directions.

3.3.5.3 [100] Compression / Tension

Fig. 3.5 shows a detailed picture of tension to fracture for each of the geometric configurations. The potential used was the Proville / Rodney / Marinica 2011 potential. Energy increased while pressure decreased until fracture occurred. The strain at which fracture occured was highest for full-lattice BCC, followed by mono-vacancy, self-interstitial atom configurations, then di-vacancy, and last full-lattice FCC. After fracture pressure was zero and energy was stable until the surfaces became proximal again at which point they repulsed each other before recombining. Damage was evident after recombination. Compression resulted in a large increase in pressure with a higher increase in energy for a strain of 0.8 than for 1.2 as seen by comparing time index 120,000 with time index 20,000.

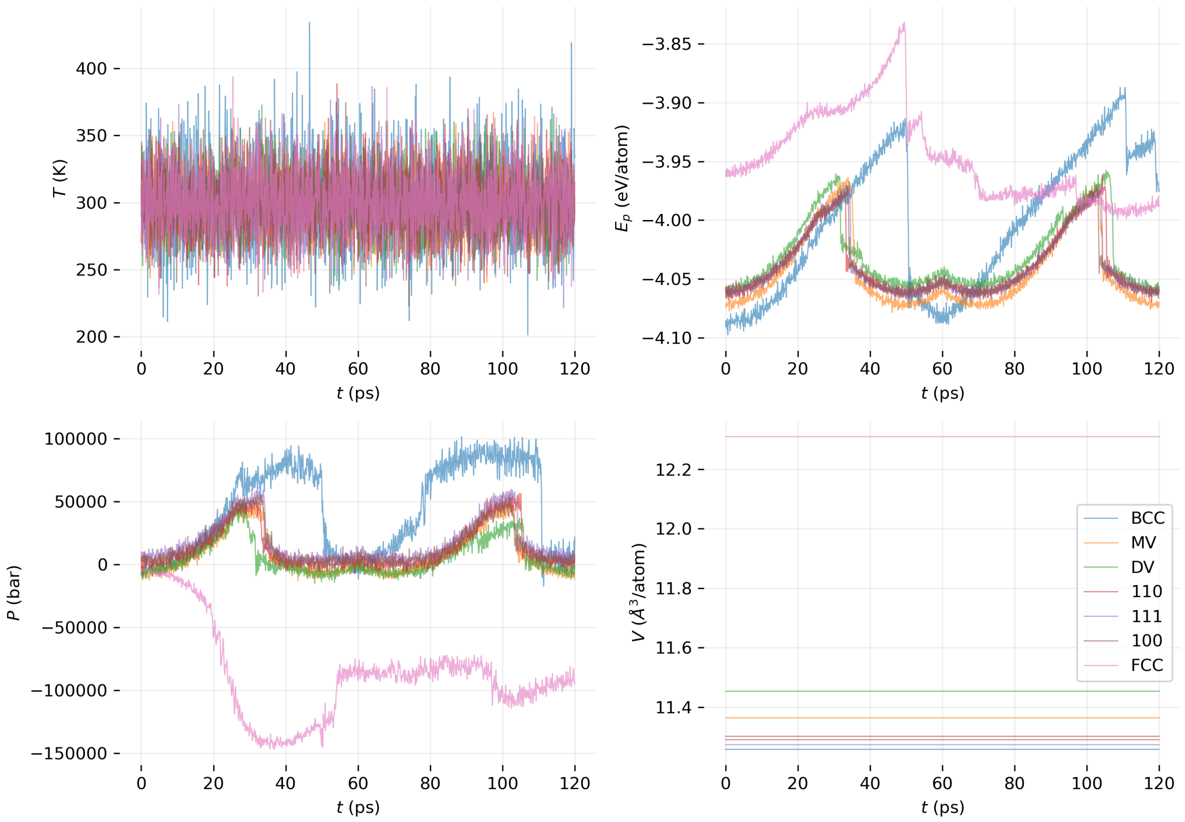
Figure 3.5: “Tour de Fer” 100 compression and tension MD.



3.3.5.4 [100] Shear

In Fig. 3.6 shear strain increases to 0.3 at time index 60,000, then decreases back to zero. Again, results are with the Proville / Rodney / Marinica 2011 potential. Different geometric configurations are plotted together as before. Slip occurs for different configurations at different times, with the same rank ordering as before except the full lattices exhibit slip at the same points. Damage accumulation is significant for the full-lattice systems, however those with defects tolerated slip well.

Figure 3.6: "Tour de Fer" 100 shear MD.



Interestingly, the FCC configuration exhibited significant negative pressure under shear until slip occurred.

3.3.6 Low T Ramp

The "Low T Ramp" uses 2x2x2 BCC and FCC configurations generated in LAMMPS under NPT conditions going from 300K to 1500K. The spin configuration specifications for this dataset are slightly different from the overall dataset.

$$\vec{s}_{\text{FM}} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad \vec{s}_{\text{111r}} = \begin{bmatrix} \mathcal{U}[0,1) \\ \mathcal{U}[0,1) \\ \mathcal{U}[0,1) \end{bmatrix} \quad \vec{s}_{\text{PM}} = \begin{bmatrix} \mathcal{U}[0,1) - 0.5 \\ \mathcal{U}[0,1) - 0.5 \\ \mathcal{U}[0,1) - 0.5 \end{bmatrix} \tag{3.33}$$

These directions would be normalized as necessary and then multiplied by the spin magnitude.

3.3.7 High T Ramp

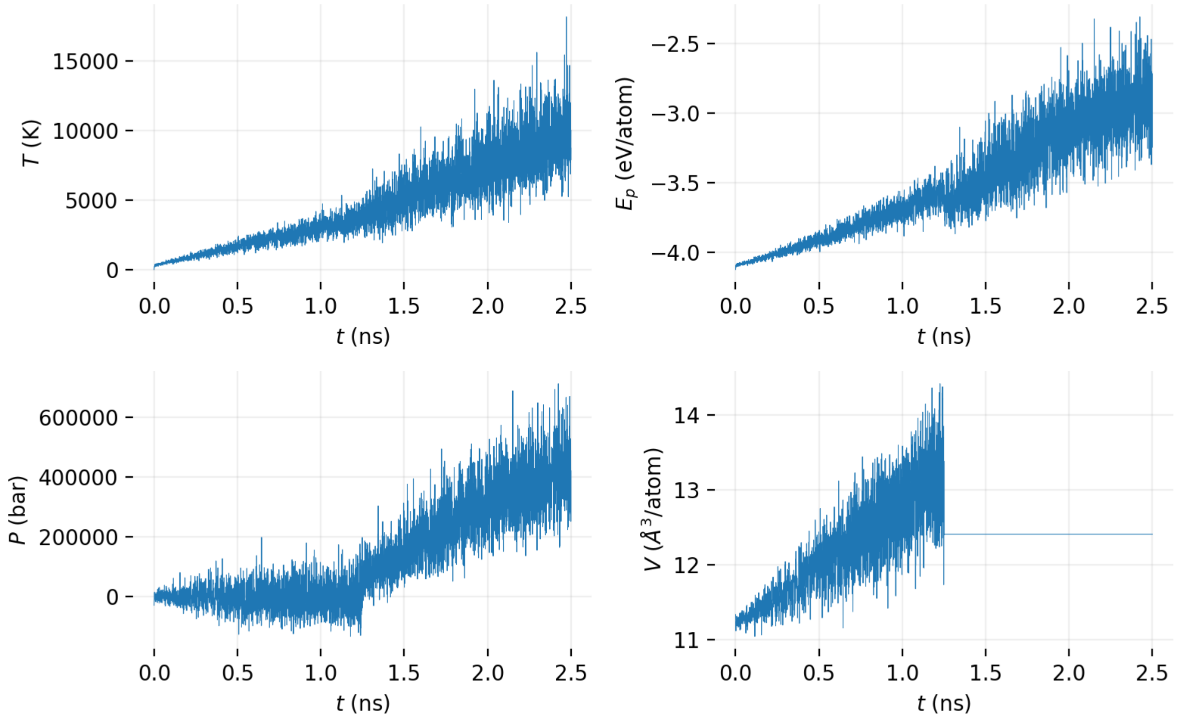
The “High T Ramp” uses a 2x2x2 BCC configuration generated in LAMMPS under two regimes:

1. NPT: 300K → 3800K
2. NVT: 3800K → 10000K

Nine spin configurations were then generated as described above: FM, four perturbed FM, PM, and three AFM.

The intent of the “High T Ramp” is to explore extreme geometric perturbations so as to provide further meaningful challenge to BP and SANNIP.

Figure 3.7: High T Ramp MD.



3.3.8 Nanowire

Nanowire geometric configurations were generated in LAMMPS again using the Marinica 2011 potential. A 12 Å^3 box was populated with a thin wire of 20 atoms that was continuous in one dimension. An NVT ramp from 300K to 5000K and then back down to 300K was used to evaporate and then condense the wire. After condensation, the wire changed the axis over which it was continuous several times, finally settling on an orientation different from that at outset.

Nanowire spin states were sampled in the same manner as the High T Ramp.

Chapter 4

Results

4.1 Spin Distribution of DFT Data

The first important result of this thesis is the spin distribution achieved in the dataset that was generated as described in the previous Chapter. There are several important questions to address:

- What quantitative measures best characterize magnetic phase and elucidate transitions?
- Are there FM states across a broad distribution of energies?
- Is there a smooth transition from FM to PM and AFM across a broad distribution of energies?

A reasonable measure of magnetization and FM character is the ratio of net system spin magnitude to the sum of atomic spin magnitudes:

$$\frac{||\sum \vec{s}||}{\sum ||\vec{s}||} \tag{4.1}$$

When this quantity has a value of 1, the system is perfectly FM. When it is 0, the system can be perfectly PM or AFM, or many other possible configurations. A second quantitative measure is therefore needed, and found in the variance of the polar angle from the system net spin:

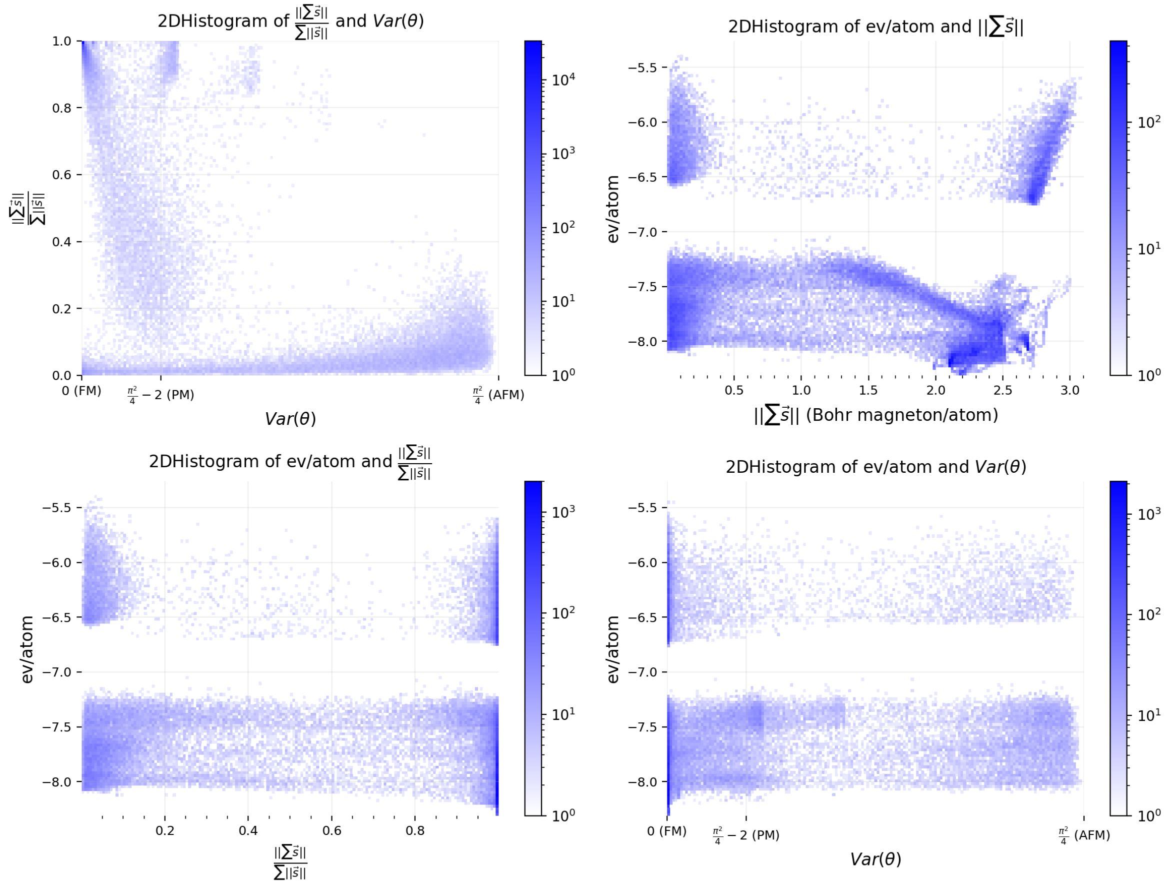
$$\text{FM:} \quad \sigma_\theta^2 = 0 \tag{4.2}$$

$$\text{PM:} \quad \sigma_\theta^2 = \frac{\pi^2}{4} - 2 \tag{4.3}$$

$$\text{AFM:} \quad \sigma_\theta^2 = \frac{\pi^2}{4} \tag{4.4}$$

Fig. 4.1 shows 2D histograms of spin state distributions in E, $||\sum \vec{s}||$, $\frac{||\sum \vec{s}||}{\sum ||\vec{s}||}$, σ_θ^2. The above described data generation scheme indeed covers a broad range of energy and spin configurations, including PM and AFM states. In particular, low/zero-magnetization states are spread across the full range of $Var(\theta)$ from FM through PM to AFM. The clearest avenue for improvement is to add more PM states, particularly for the high energy states which correspond to the nanowire configurations.

Figure 4.1: 2D histograms of spin state distributions in E, $||\sum \vec{s}||$, $\frac{||\sum \vec{s}||}{\sum ||\vec{s}||}$, and σ^2_θ.



4.1.1 Long T Ramp Analysis

5000 geometric configurations of 16 atoms were assigned nine different sets of initial magnetic moments, resulting in 45k samples. Fig. 4.2 shows the range of configurations from FM to PM (omitting AFM).

The top left subplot is a histogram of atomic magnetic moment magnitudes across all samples in this series. This distribution is important because various *ab initio* studies tend to report average or preferred magnetic moments, however there is far less literature on detailed distributions of spin magnitudes (as determined by DFT) in perturbed bulk lattices of common materials. This plot can be broken down or expanded separately to show spin magnitude distributions for different geometric configurations or as a function of energy.

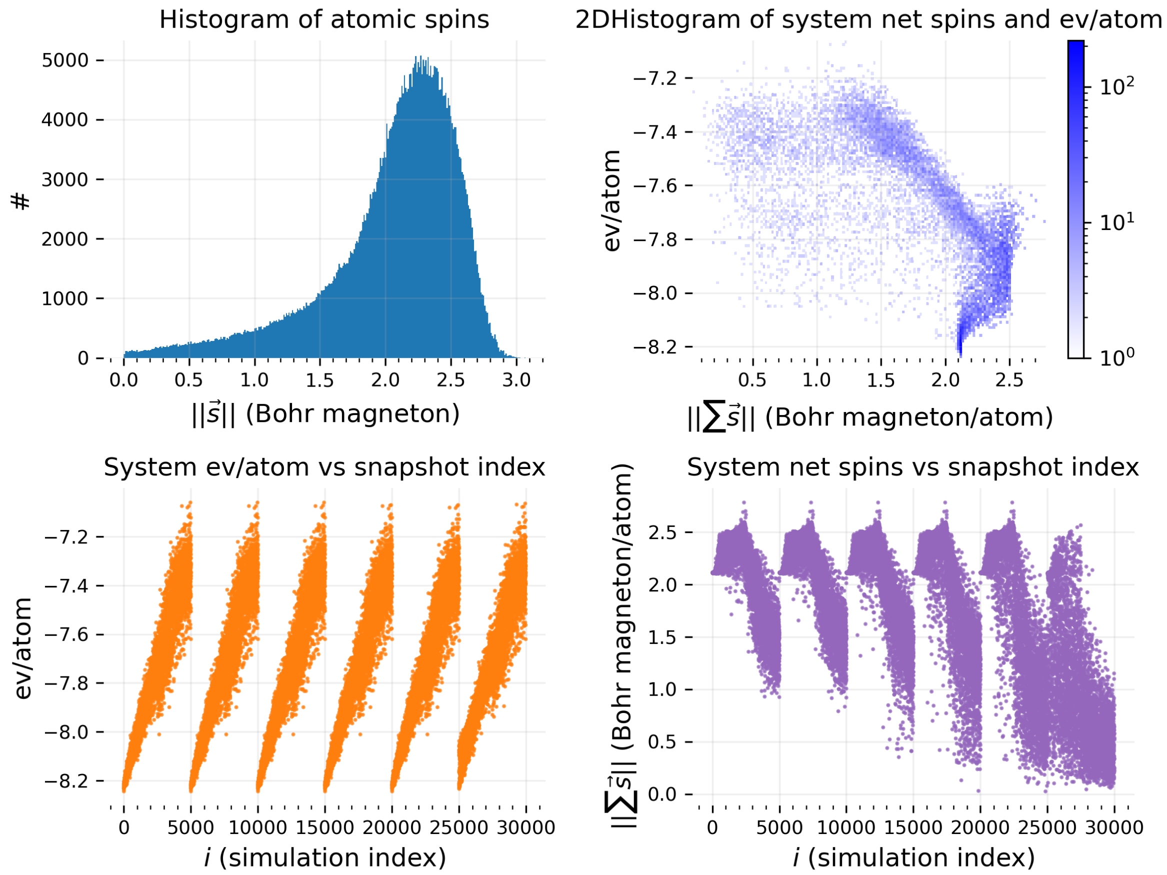
Further analysis along these lines could be interesting because there may be low or high spin magnitude atoms even in common thermal geometries and that could be mechanistically important in thinking about defect migration / aggregation or hydrogen adsorption in future work.

The top right plot is a 2D histogram of eV/atom vs net system magnetic moment magnitude. This shows that a broad range of system net spins are possible, although this distribution collapses to a single average at low energy. Paramagnetic character is observable at energy less than 200 meV/atom above the energetic ground state.

The bottom left shows a scatter plot of the energy values for each sample and is organized in six sets of 5000 samples (corresponding to geometric configurations), with each series of 5000 evaluated with an increased variance in polar angle, until the final series which is initialized to be fully paramagnetic. It is apparent that the final series has a significantly wider band of energies after index 25,000 than the other series do at their starts, with these areas of lower energy corresponding to the 300K start of the NPT ramp.

At bottom right system net spin magnitude is displayed for each simulation index. The strategy of gradually increasing polar angle dispersion has resulted in increasing tendency toward PM spin configurations.

Figure 4.2: Long T ramp spin state synopsis.



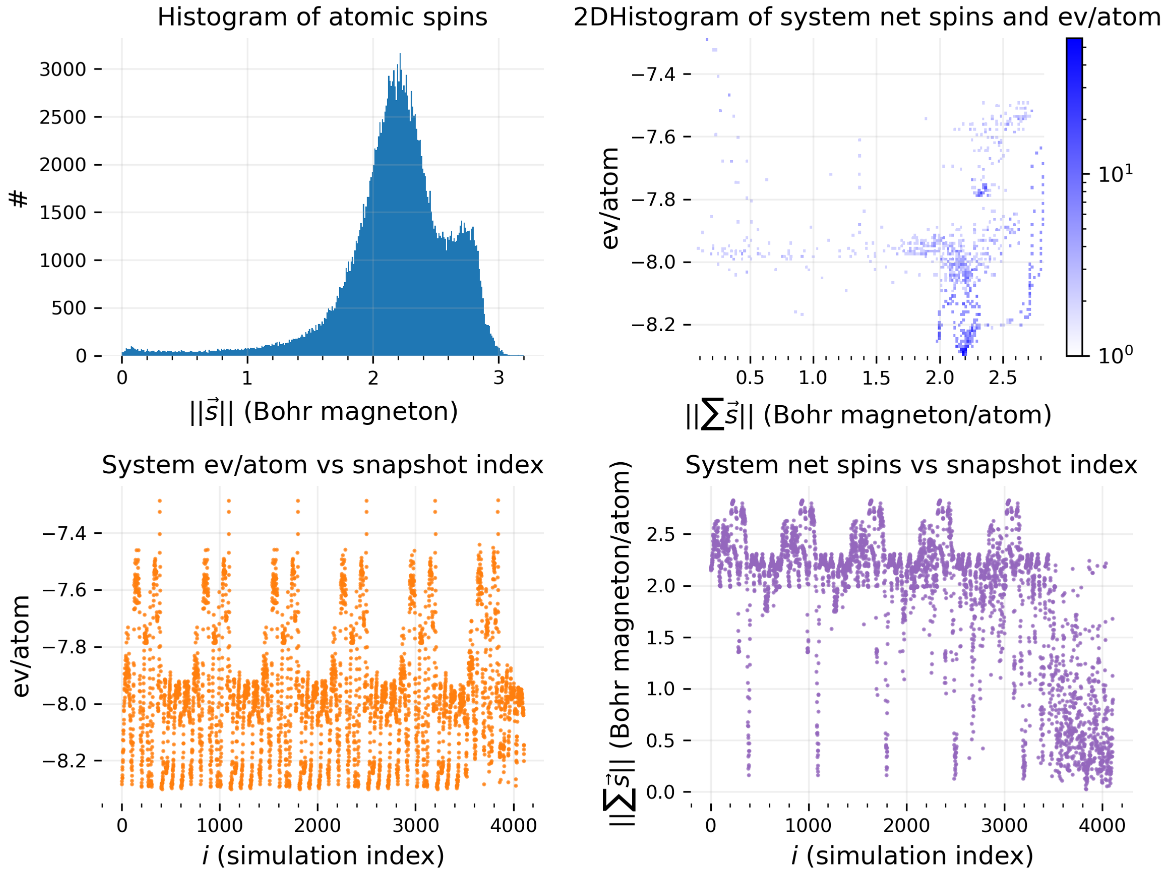
4.1.2 3x3x3 BCC full lattice TDF w/ varied spin

Fig. 4.3 are uses the same axes as Fig. 4.2 on all subplots.

This time, in the top left there is a bi-modal distribution in spin magnitude over these series, suggesting that different deformations are associated with significantly different spin distributions.

In the top right it is worth noticing that there are again low net system spin states at relatively low energy, and at bottom right there is the desired increasing tendency toward PM states as polar angle spin dispersion increased.

Figure 4.3: 3x3x3 BCC full-lattice "Tour de Fer" spin state synopsis.

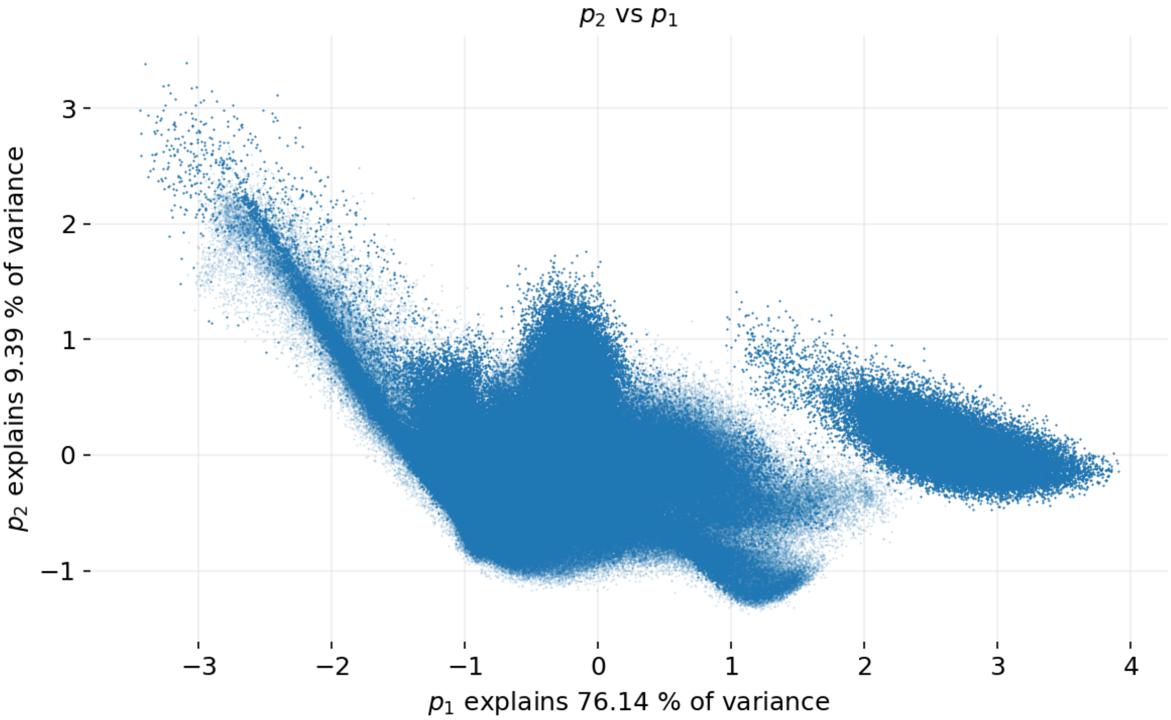


4.1.3 PCA of Spin-Aware Fe Dataset

Fig. 4.4 shows the PCA-space transform of the input features for the full dataset using this work's reference BP featurization. The data samples form a complex shape that in no way resembles a bi-variate normal distribution, and there are many paths between densely sampled regions that have sparsely sampled (or unsampled) regions between them. This means that the geometries present in the dataset (the Spin Bases are not utilized here) have some non-smooth transitions between forms. For example, the data to the far right is the nanowire data, which has a small number of atoms in a vacuum, whereas the large group of samples to the left is more or less bulk material (with a single point defects or di-vacancy). Digging deeper, the highly compressed "Tour de Fer" states are in the far-left group that extends linearly up and to the left from (-1, -1) to as far as (-3, 3). Geometric states nearest ground are near (-0.75, -0.75). The tensile fracture configurations are related to the tuft at (1, -1). The lump peaking at about (-0.3, 1.5) is associated with intensely sheared configurations. It is interesting therefore to see that specific types of geometric perturbations from ground state move along very different paths, and also that the extremes of those deformations tend to only connect through the ground state.

A major point about all that structure is that densely sampled yet isolated data regions are harder to fit, especially using NNs.

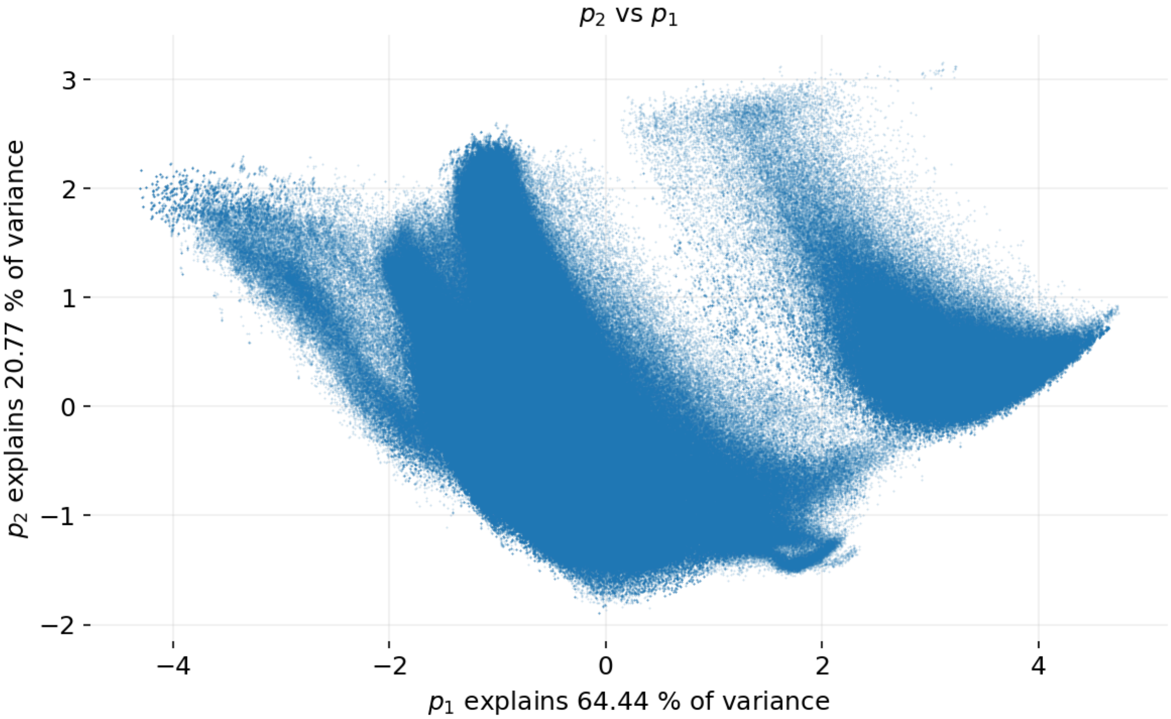
Figure 4.4: PCA transformed spin-agnostic features.



Other authors examine PCA on "feature vectors" from the GAP/SNAP methodology and then analyze clusters within to characterize microstructure. [91,92] Such a process is entirely conceivable to perform using BP/SANNIP featurizations, and may even be preferable given the simpler form which can be fast to compute and easy to tune when looking for specific things. This is left to future work.

Fig. 4.5 shows the PCA-space transform of the input features for the full dataset using a SANNIP basis set. The SANNIP features contain both geometry and spin information, which has the effect of broadly smearing the data in p_1, p_2-space. While the sum of variance explained by p_1 and p_2 remains roughly the same at $\sim 85\%$, the variance explained by p_2 more than doubled. The bounding curve that surrounds the dataset is shorter relative to the bounded area than with BP. It makes sense then that SANNIP could perform better than BP numerically, because there are fewer isolated data points far from the bulk.

Figure 4.5: PCA transformed spin-aware features.



The BP-featurized data in p_1, p_2-space is more concentrated than the SANNIP-featurized data. However, as seen in the next section, the SANNIP energy RMSE is more concentrated about zero compared to the spin-agnostic BP energy RMSE. By smearing the input in a physically relevant way, the output error distribution has been condensed.

4.2 Energy Prediction Error

The first step in proving the efficacy of our method is through its advantage over conventional, spin-agnostic BP in terms of energy prediction error.

In our case the most obvious and ubiquitous metric of performance for regressors is Root Mean Square Error (RMSE). Mean Absolute Error (MAE) is also a metric of performance, although RMSE is more sensitive to outliers and therefore more difficult to reduce than MAE.

NN hyper-parameters were kept constant across the wide range of scenarios presented, for both practical and conceptual reasons. Starting with the practical, optimal hyper-parameters cannot be known in advance and require extensive trial-and-error to investigate. Even perfunctory hyper-parameter optimization includes many different layer sizes and training durations, and an exhaustive search goes into six dimensions. Since the basis functions and NN architectures are the substantive scientific contributions of this thesis, computational resources were focused on exploring that space. Optimizing hyper-parameters for each prospective basis set and NN configuration would have increased computational needs by more than one order of magnitude per experiment. Keeping them the same throughout also makes the comparison between ML methods fair and accurate in terms of the potential developer's level of effort.

The hyper-parameters used are:

- Learning rate: 0.0001
- Training duration: 16000 epochs (one "epoch" here means "using every training sample exactly once")
- Batch size: 100 samples
- Fully connected hidden layer quantity: 2 layers
- Fully connected hidden layer size: 25 neurons/layer
- Activation function: ReLU

- Validation fraction: 0.2

There are however some unavoidable differences that may obscure ML process comparisons:

- Dataset composition: Some datasets contain more configurational variation which expose weaknesses of a system under test.
- Trainable parameters because of inputs: The Spin Bases add inputs to either NN scheme, thus increasing the number of trainable parameters in models with the same number of hidden-layer nodes.
- Trainable parameters because of architecture: The CNN architecture uses more trainable parameters than conventional BP NNs.

This work presents comparisons of four cases on three data subsets using two evaluation procedures. The BP basis set and the Spin Basis sets are constant. The four cases are:

- BP Bases using BP NN
- BP Bases using this work's CNN
- BP + Spin Bases (SANNIP) using BP NN
- BP + Spin Bases (SANNIP) using this work's CNN

The three data subsets are labelled:

- High-Spin Variation
- High-Geometry and High-Spin Variation
- High-Geometry and Low-Spin Variation

The two evaluation metric definitions are discussed in the following two subsections.

4.2.1 Aggregate Prediction of a Bag of Models

The trainable parameters of NNs are generally initialized to random values, which means that an identical training process using the same training dataset most likely will produce different results. This is not unique to NNs; many process in the world of ML are non-deterministic in this way. The training procedure itself can also infuse significant stochasticity, due to validation sample selection, batch composition, and sample/batch presentation order.

Therefore, if an ML process is executed to produce trained models, and every model is different, a set of trained models must be obtained in order to draw statistically significant conclusions about the effectiveness of the ML process that produced the models.

Since the splitting of the overall dataset into the training dataset and validation dataset can create a bias, the validation dataset is chosen randomly for each model. No two models have the same validation dataset, so points that are likely to be particularly instructive for training or penalizing in validation get shuffled from model to model, providing the broadest and most general picture with which to compare ML processes.

When all models have different and uniformly random validation datasets, then for a large enough "bag of models" each data sample will on average be included in a validation set a number of times equal to the validation set fraction. When that is the case, every model in the "bag" can be used to make a prediction for each data sample (instead of just its validation samples), and those predictions can be averaged to produce an aggregate result for each data point that is representative of the ML method—and not necessarily any particular instance of a trained model. For the aggregate prediction given sample i and M models:

$$E_{\text{agg},i} = \frac{1}{M}\sum_{m=1}^{M} E_{m,i} \tag{4.5}$$

The MAE and RMSE of the aggregate predictions of a bag of models on the entire available dataset is therefore the first metric used in this work for comparing ML methods that produce instances of trained models.

4.2.2 Validation Error of a Bag of Models

Our second metric is MAE and RMSE of the validation error on the concatenation of the set of validation sample predictions from each model in above mentioned "bag."

This value is generally not the same as the above, and can be either higher or lower since validation points chosen at random can be in hard-to-predict, high-error regions of the dataset or easy-to-predict, low-error regions.

This is another reason why multiple trained models from each method are needed to evaluate overall ML process performance.

$$E_{\text{val}} = E_{m_0} \cup E_{m_0} \ldots E_{m_M} \tag{4.6}$$

4.2.3 Runtime & Overall Figure-of-Merit (FOM)

Runtime for SANNIP and BP in MD environments depends on: (1) the time to calculate the basis function values for each atom, and (2) the time to calculate the system's energy and forces using the NN. Since angular BP bases are three-body terms, while radial and spin bases are two-body, the impact of spin basis value calculation on MD execution time is very small. A typical number of atoms inside a cutoff radius of 6 Å is about 80 for iron, meaning that adding 80 radial or spin bases increases the featurization time by the same amount as adding just one additional angular basis. The increase in featurization time from adding 20 spin bases can be neglected, while the same cannot be said for the NN computation time where the effect of one more basis is independent of type. Because training requires the NN calculations proportional in number to those needed for MD, training time differences reasonably approximate differences in overall execution time in MD environments using the different methods.

If users of MD and MC value a balance between execution time and RMSE, then one reasonable Figure of Merit (FOM) for IPs is the inverse of product of the two:

$$\mathrm{FOM} = \frac{1}{\text{exec. time} \times \mathrm{RMSE}} \tag{4.7}$$

For the results of this work, the cross-model concatenated validation error RMSE and training time for one complete use of the training dataset will be used to calculate FOM.

Training times were computed on the Ju Li Group's computational cluster using nodes with two Intel® Xeon® CPU E5-2650 2.00GHz processors each. For each time trial, a single node was fully utilized, including its memory. GPU computations may achieve different results.

4.2.4 High Geometry- and Spin-Variation Dataset

Ultimately, the highest impact comes from being able to make accurate predictions across configurations of *high variation in both geometry and spin*. Fig. 4.6 shows energy prediction error per atom ($E_{pred} - E_{DFT}$) vs average system energy, both in meV/atom using BP and the BP NN. First, there is clear separation between the different spin states: FM states are overpredicted, PM states vary considerably, and AFM states are hugely underpredicted. Note also how the PM and AFM states for the same geometries are significantly higher in energy and systematically under-predicted in energy, evident by the downward sloping bound to the right of the lowest energy samples for those spin states. Next, as seen at far left, BP is overpredicting the ground state energy by approximately 100 meV/atom, which is a very high amount of error. As mentioned in the tutorial, extrema can be hard to precisely capture with NN methods—and unseen spin makes the situation far worse. Also, the difference between energy states in samples < -7meV/atom drops as energy increases, forming an overall triangular shape in this error vs average system energy per atom plot. The difference between the ground state error of about +100 meV/atom and nearly -400 meV/atom amounts to a range of almost 0.5 eV/atom in energy prediction error due to spin—in what are the low energy lattices. This means that the energy penalty of spin perturbations from FM are greatest near equilibrium lattices and decrease with increasing system energy, and therefore temperature. Looking to the higher energy group which are nanowire geometries however there is a magnetic phase energy band about 200 meV/atom tall. In this area, the error due to spin perturbations seem relatively constant in energy, suggesting that there are different regimes beyond BCC and FCC for which empirical exchange relations would have to be defined by practitioners of that approach.

Fig. 4.7 shows error vs. energy using SANNIP and the BP NN on the same dataset. The systematic bias evidenced by clear separation of errors according to spin configuration is gone, and RMSE is dramatically reduced. However, the ground state in this case is still over-predicted.

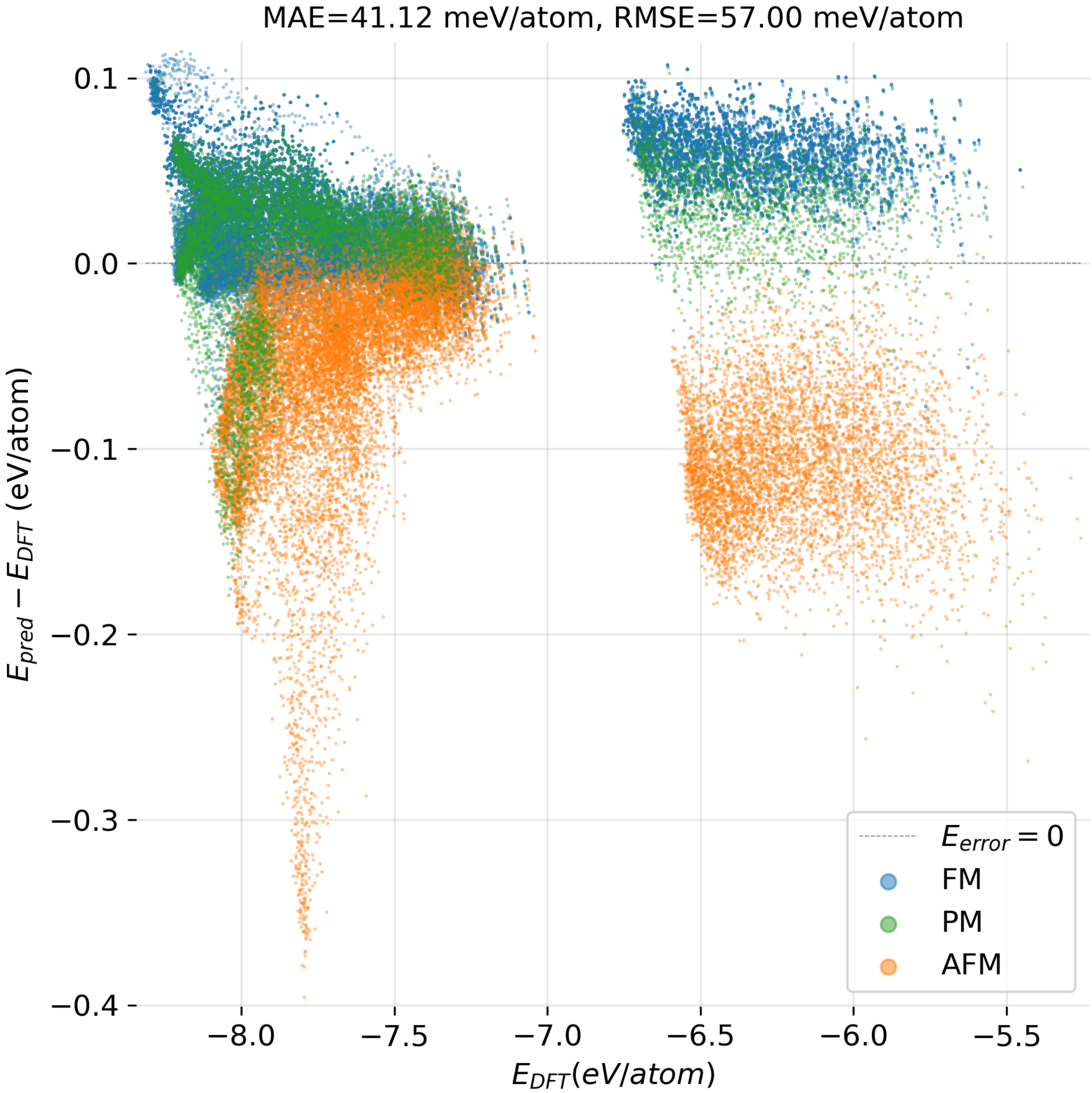


Figure 4.6: High geometry and high spin variation aggregate prediction using BP bases and BP NN.

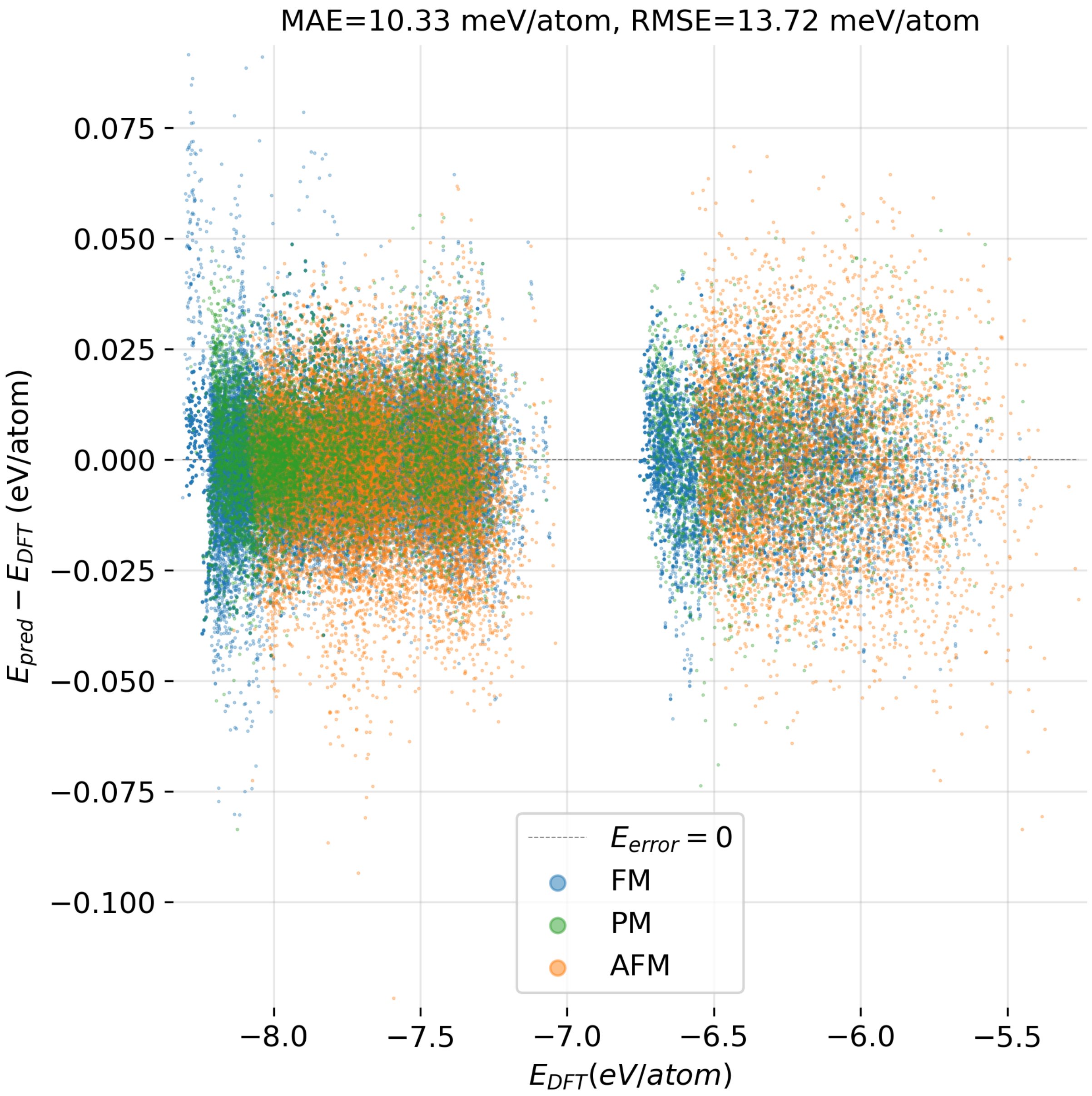


Figure 4.7: High geometry and high spin variation aggregate prediction using SANNIP and BP NN.

Fig. 4.8 shows BP bases using the new CNN of this work. While bounds in error are reduced slightly, the situation is not meaningfully different than BP using the BP NN. There are still large errors and groupings according to spin. Fig. 4.9 shows SANNIP using the new CNN, yielding further reduction in energy prediction RMSE and also reducing ground state error compared to SANNIP using the BP NN. This last qualitative observation may be significant for some users.

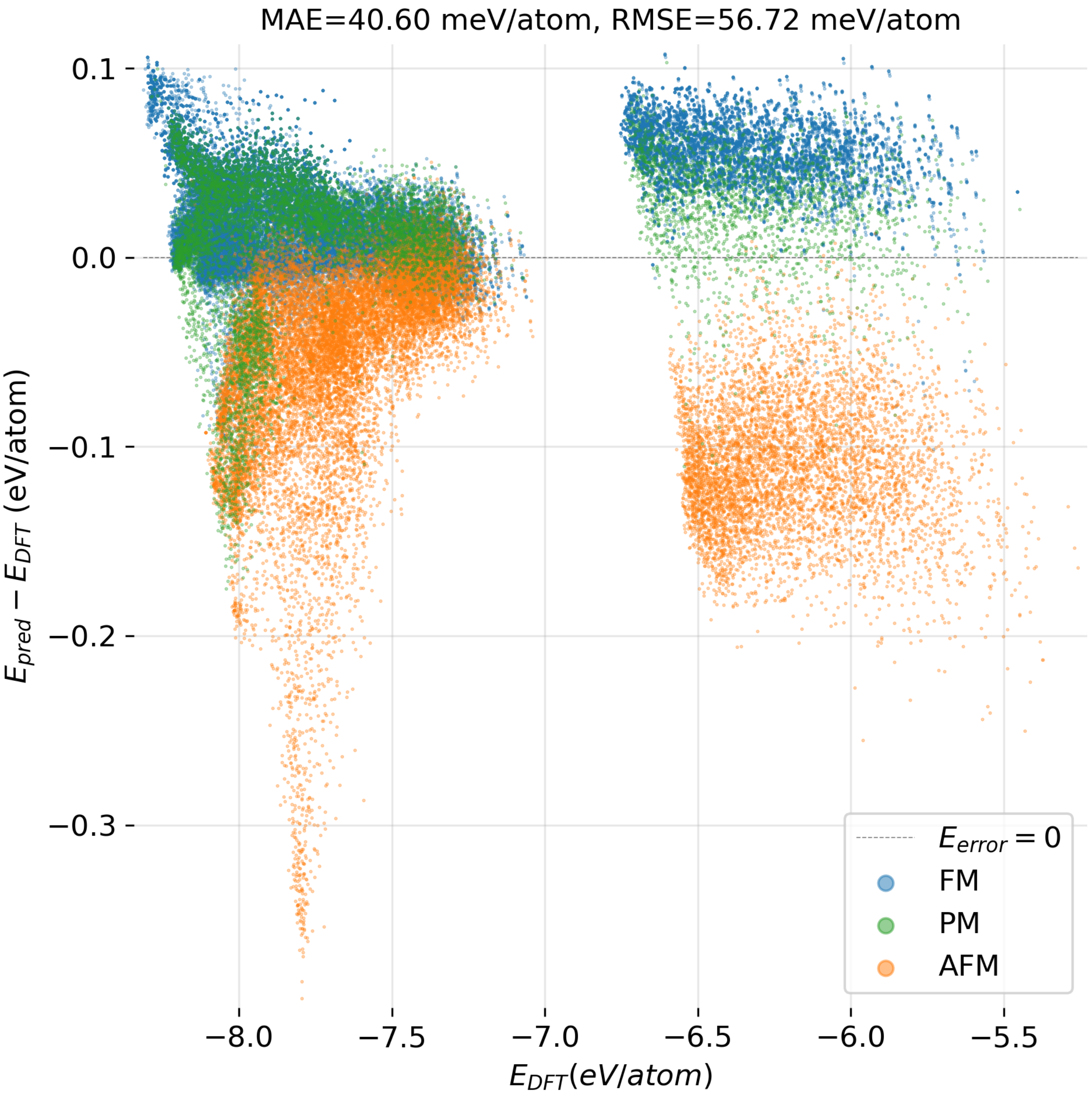


Figure 4.8: High geometry and high spin variation aggregate prediction using BP bases and new CNN.

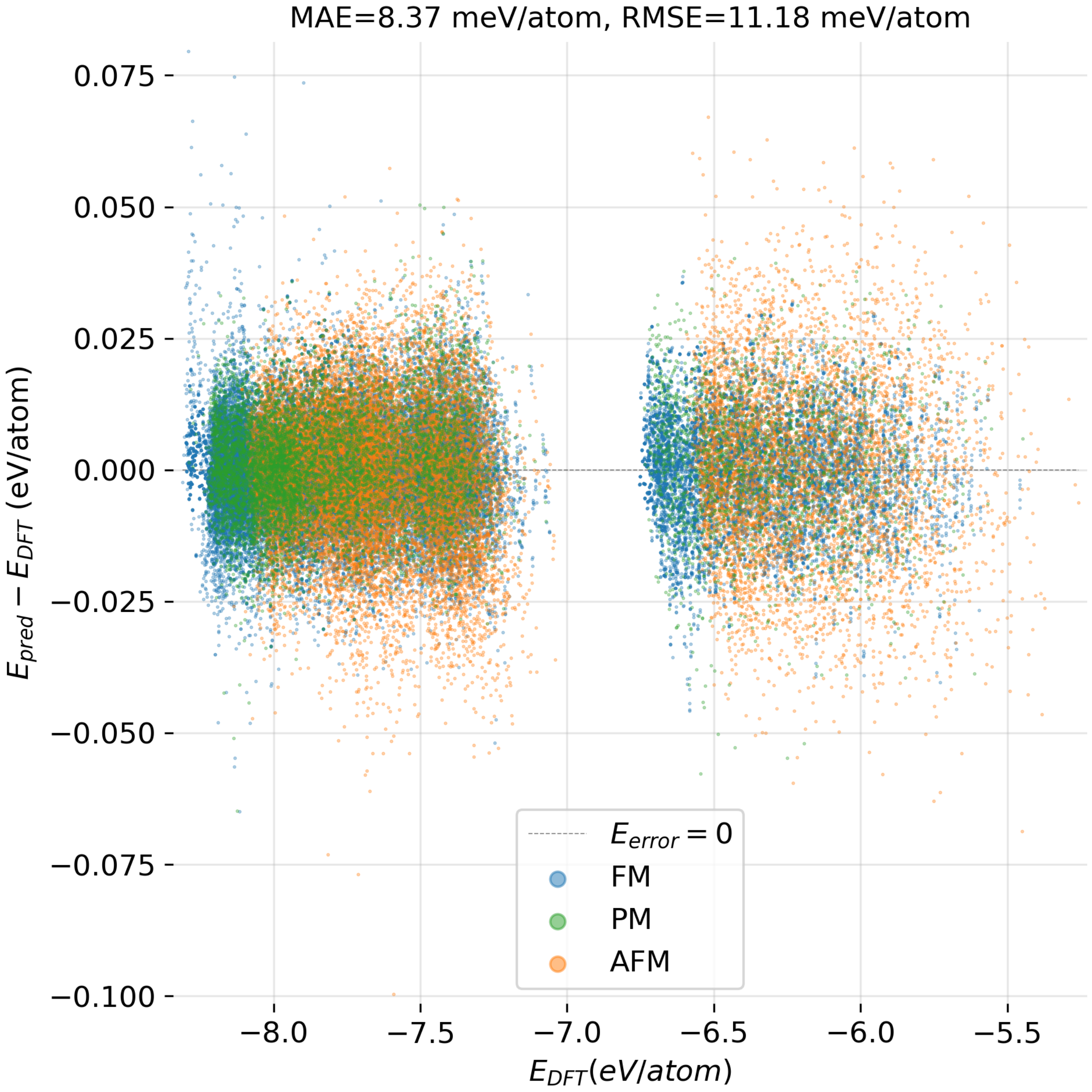


Figure 4.9: High geometry and high spin variation aggregate prediction using SANNIP bases and CNN.

Fig. 4.10 shows the validation error histograms for all four above mentioned cases. Spin-agnostic BP cannot achieve a normal distribution in error, which is suggestive of the systematic bias seen above in the clearly differentiated error groups labelled by spin. Mathematically, the BP error distributions do not have a mode of zero, and are highly skewed toward positive errors. SANNIP practically eliminates this skewness and produces more normally distributed errors. The CNN is only marginally useful in the systematically biased spin-agnostic BP cases, however it does narrow the error distribution for SANNIP.

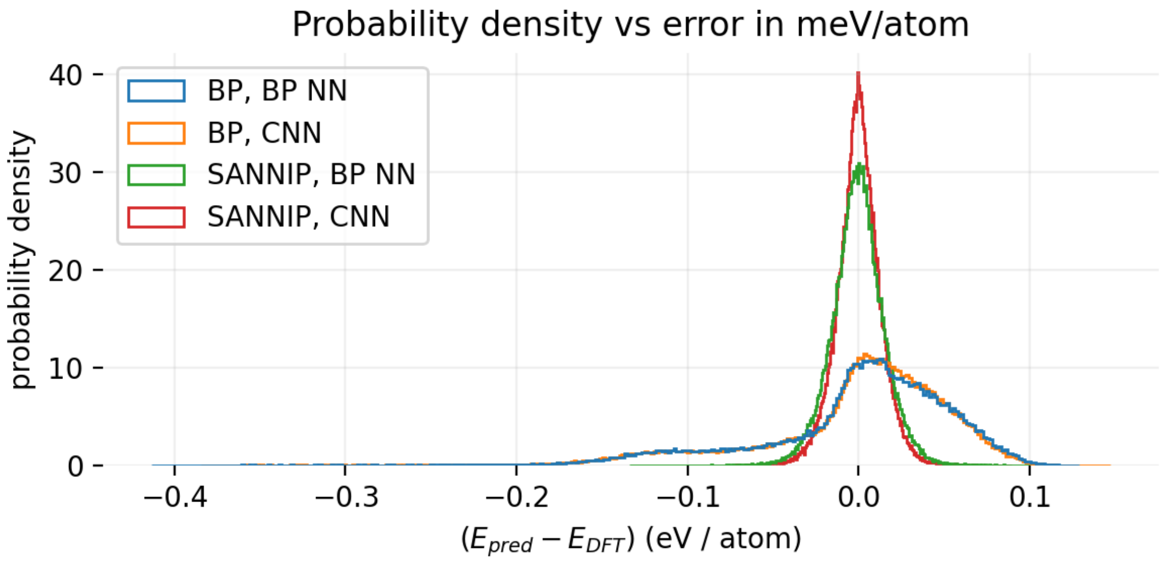


Figure 4.10: SANNIP vs BP, validation error histogram on high geometry and high spin variation dataset.

Table 4.1 summarizes these energy prediction error results for the complete set of combinations of bases and NNs for this dataset. It also states: the training time per use of the entire dataset (or epoch), the Figure-of-Merit (FOM) for each case, and the ratio of a case's FOM to that of BP using the BP NN. Use of the SANNIP bases increases execution time only slightly, whereas the new CNN nearly triples it. The highest FOM is therefore achieved by SANNIP using the BP NN, followed by SANNIP using the new CNN. BP using the new CNN is the worst performer by this metric. Fig. 4.11 shows execution time per epoch (s) vs. validation RMSE (meV/atom) for the four methods, and contours of equal FOM intersect the values of each.

Bases	BP	BP	SANNIP	SANNIP
NN Type	BP NN	CNN	BP NN	CNN
Agg. RMSE (meV/atom)	57.00	56.72	13.72	11.18
Val. RMSE (meV/atom)	57.76	57.50	15.56	12.83
$\frac{\text{s}}{\text{epoch}}$	10.2	27.0	10.4	30.5
FOM×10^3	1.70	0.644	6.18	2.56
FOM/FOM$_{\text{BP, BP NN}}$	1.00	0.378	3.63	1.50

Table 4.1: High geometriy- and spin-variation dataset RMSE comparison, SANNIP vs BP.

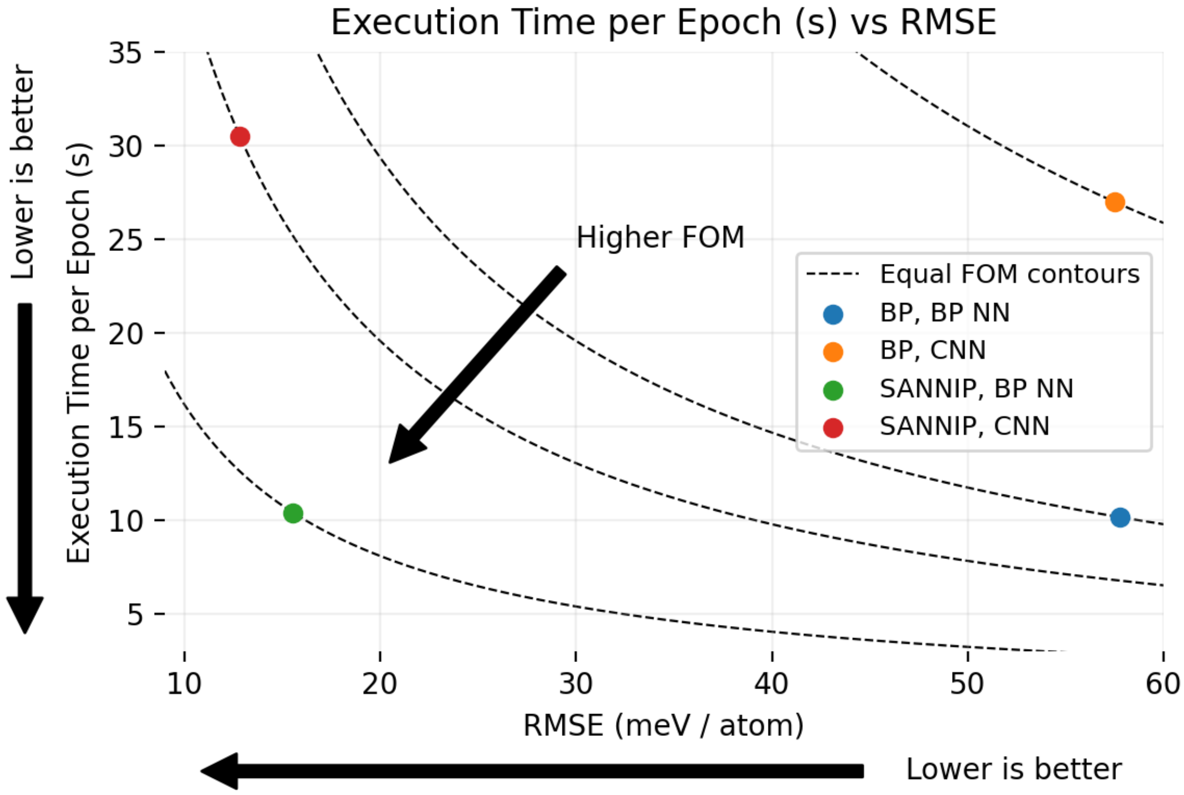


Figure 4.11: Execution time per epoch vs. RMSE across all methods applied to the high-geometry and high-spin variation dataset.

To summarize, SANNIP dramatically reduces or eliminates the systematic bias due to variation in spin configuration as seen in directly in Figs. 4.6, 4.7, 4.8, and 4.9. The error distributions shown in Fig. 4.10 further support this interpretation. Using a FOM that balances energy prediction RMSE with a reasonably proxy of execution time in MD environments, SANNIP using the BP NN is the best option, followed by SANNIP with the new CNN. Lastly, SANNIP with the new CNN best resolves the ground state.

4.2.5 High Spin-Variation Dataset

What happens when considering a dataset with low variation in geometry but high variation in spin? Fig. 4.12 shows error vs. energy using BP and the BP NN on a subset of the overall dataset. This dataset's geometric configurations are both BCC and FCC, produced by MD simulations as temperature was varied from 300K to 1500K. Each geometry is used with three spin states: FM, perturbed FM, and PM. In this case, the FM states are all similar and positive in error, while the PM state errors vary considerably and include the highest errors in this application.

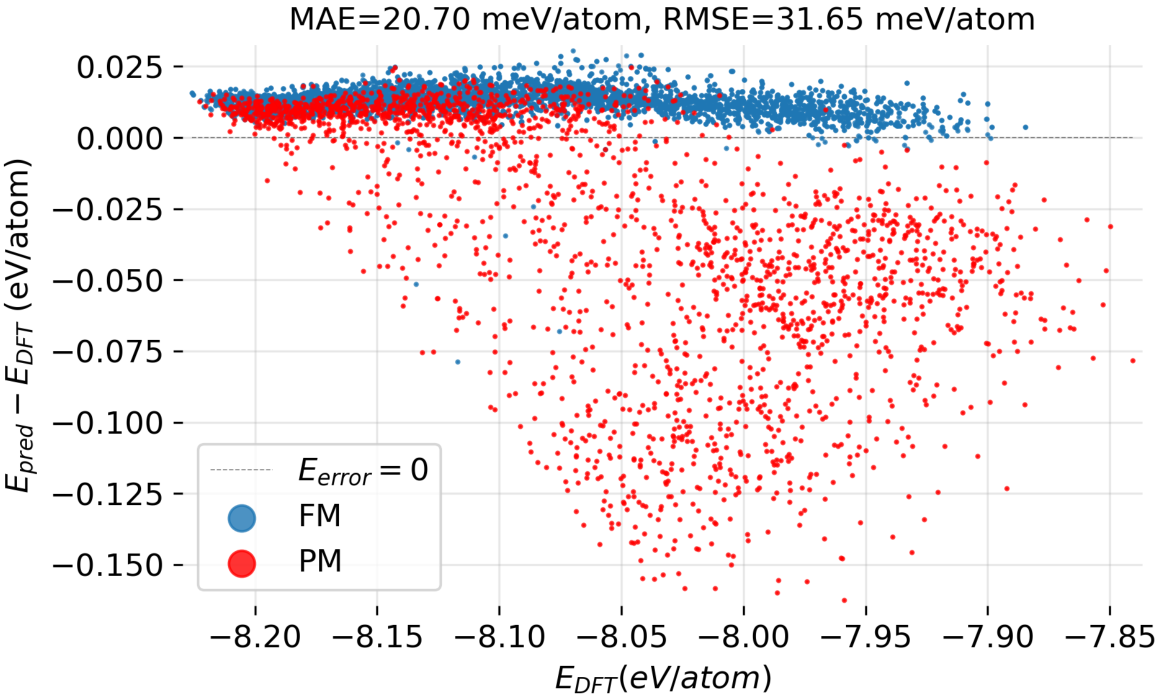


Figure 4.12: High spin variation aggregate prediction using BP bases and NN.

Fig. 4.13 shows error vs. energy using SANNIP and the BP NN on this high-spin-variation subset of the overall dataset. The grouping of errors by spin is gone and this time *RMSE is reduced by greater than 10x.*

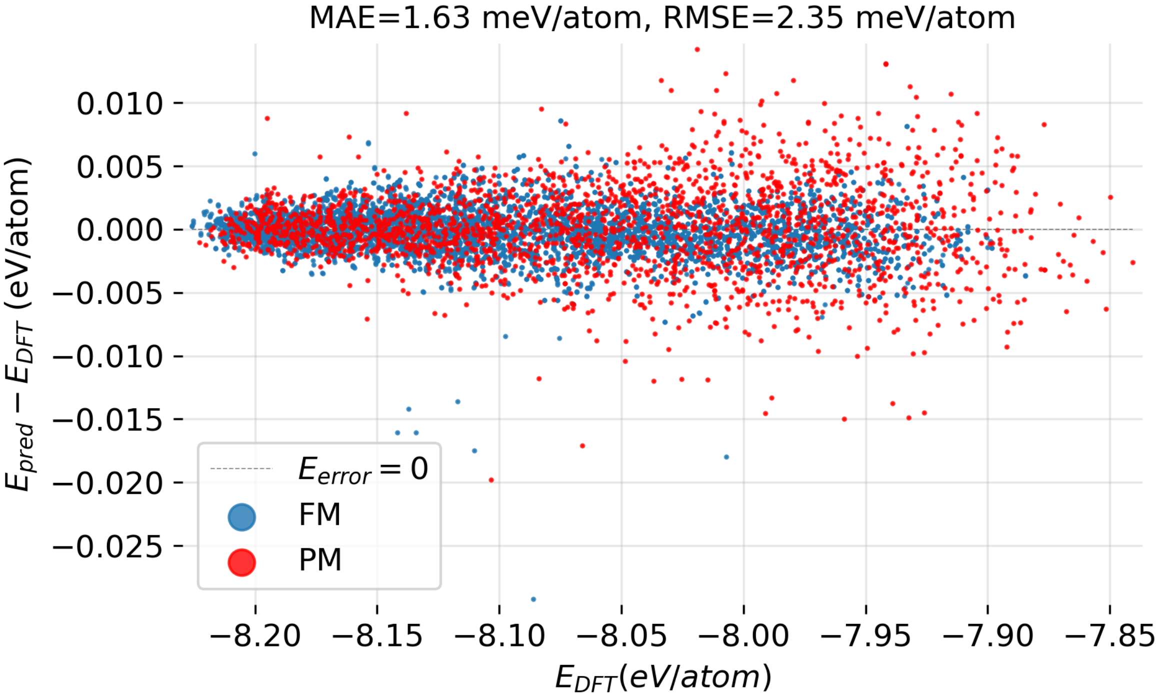


Figure 4.13: High spin variation aggregate prediction using SANNIP bases and BP NN.

Fig. 4.14 shows validation error histograms for BP and SANNIP, both using the BP NN. SANNIP again makes the mode of the distribution zero and makes it more normal.

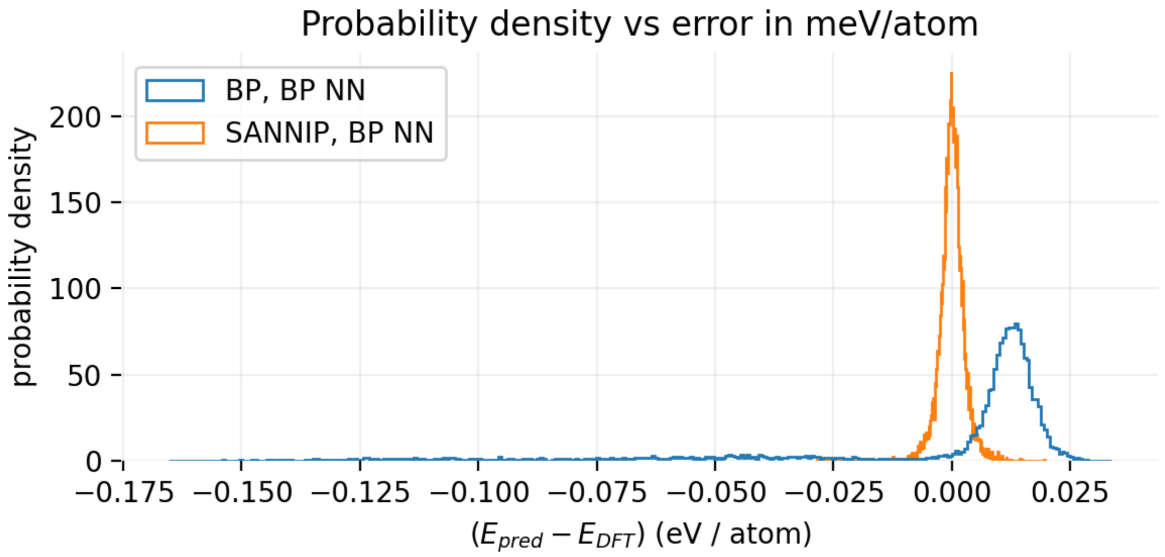


Figure 4.14: SANNIP vs BP, validation error histogram on high-spin variation dataset.

Table 4.2 shows results for the complete set of combinations of bases and NNs.

Bases	BP	BP	SANNIP	SANNIP
NN Type	BP NN	CNN	BP NN	CNN
Agg. RMSE (meV/atom)	31.65	31.67	2.35	2.44
Val. RMSE (meV/atom)	31.90	31.00	2.75	2.60
$\frac{\text{s}}{\text{epoch}}$	0.837	2.26	0.858	2.76
FOM$\times 10^2$	3.75	1.43	42.4	13.9
FOM/FOM$_{\text{BP, BP NN}}$	1.00	0.381	11.3	3.72

Table 4.2: High spin-variation dataset RMSE comparison, SANNIP vs BP.

Interestingly, in this case the novel CNN presented above did not improve the overall result compared to the conventional BP NN. There are many possible reasons, including but not limited to needing: more training epochs, a smaller network, or more samples from the data generation scheme. Further exploration of the cause of this result is left to future work.

For reference, the recent work by Novikov et al. presenting mMTP showed an energy prediction RMSE of 2.0 meV/atom on a dataset ranging from about -8.05 meV/atom to

-7.80 meV/atom. The ratio of RMSE to the energy domain spanned is therefore about $\frac{2}{250} = 0.008$. [58] SANNIP achieves an energy RMSE of 2.75 meV/atom over a domain span of 385 meV/atom, resulting an an RMSE per unit domain on comparable data of ~ 0.007.

Fig. 4.15 summarizes the results for the high-spin variation subset graphically. SANNIP using the BP NN is the best performer on a FOM-basis, followed by SANNIP with the new CNN—although there has not been demonstrated any accuracy advantage to using the new CNN with SANNIP. SANNIP is therefore a significant advance for users of MD who wish to model systems where large or extreme variations in spin state are expected.

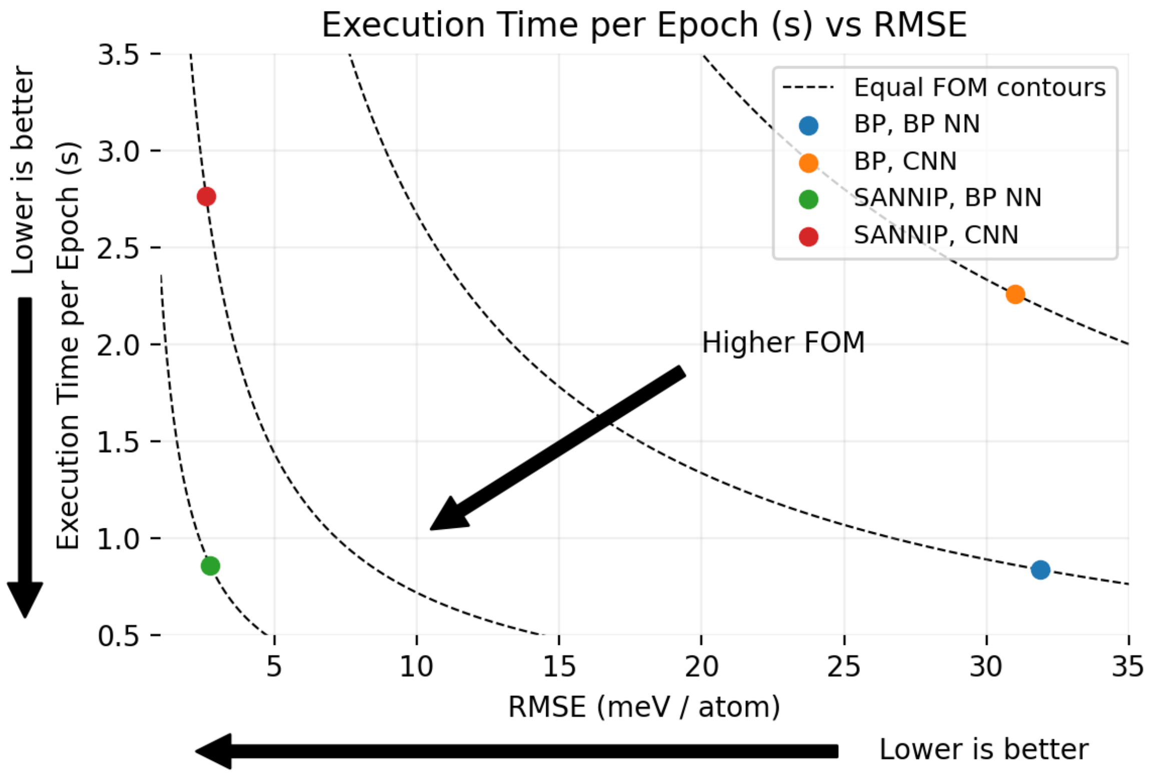


Figure 4.15: Execution time per epoch vs. RMSE across all methods applied to the low-geometry and high-spin variation dataset.

4.2.6 High Geometry-Variation, Low Spin-Variation Dataset

To round out the numerical evaluation of SANNIP, it is here asked what happens when considering a dataset with high variation in geometry but low variation in spin? This question is answered using a subset of the overall dataset focused on FM aligned configurations across a wide range of geometries including: vacancies, interstitials, and extreme distortions including tensile fracture and recombination. Fig. 4.16 shows BP and the BP NN applied to this dataset. The RMSE in this case is 28.5 meV/atom, with the highest errors in the lowest and highest energy configurations. For reference, there are only about 20k samples compared to the 38k used in the 2012 BP paper on Cu mentioned in the Background Chapter, so the relatively small sample count and magnetism of iron result in high RMSE.

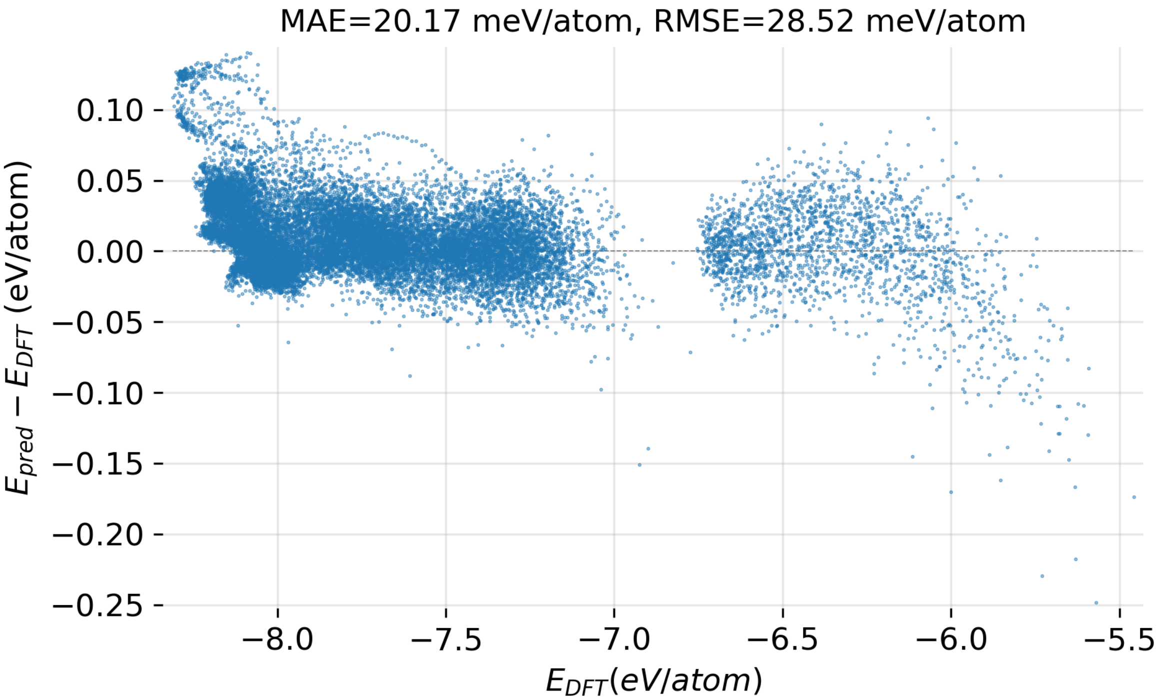


Figure 4.16: High geometry variation aggregate prediction using BP bases and NN.

Fig. 4.17 shows SANNIP and the BP NN applied to this dataset. Remarkably, even for FM aligned data, SANNIP reduces RMSE by 45%. Certain low energy states still have high error, however, as seen in the NN tutorial, this is an issue that can likely be fixed with sampling more intensely in that area of the training space.

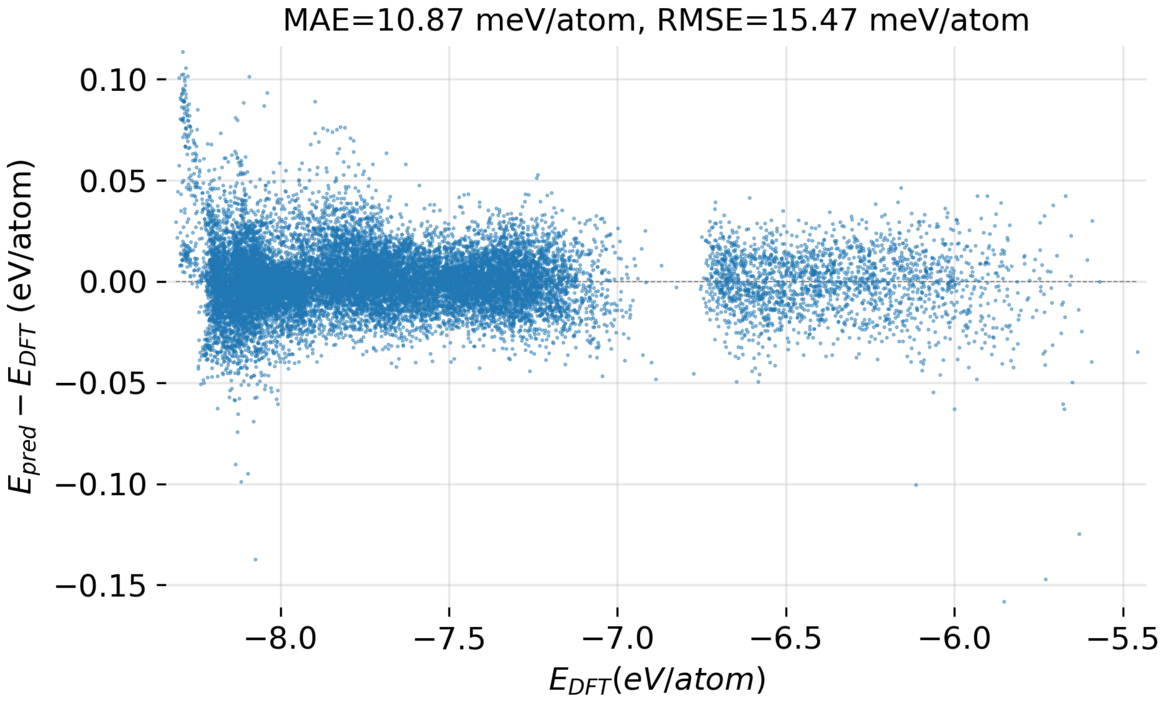


Figure 4.17: High geometry variation aggregate prediction using SANNIP bases and BP NN.

Fig. 4.18 shows validation error histograms for BP and SANNIP, both using the BP NN. SANNIP again makes the mode of the distribution zero and makes it more normal.

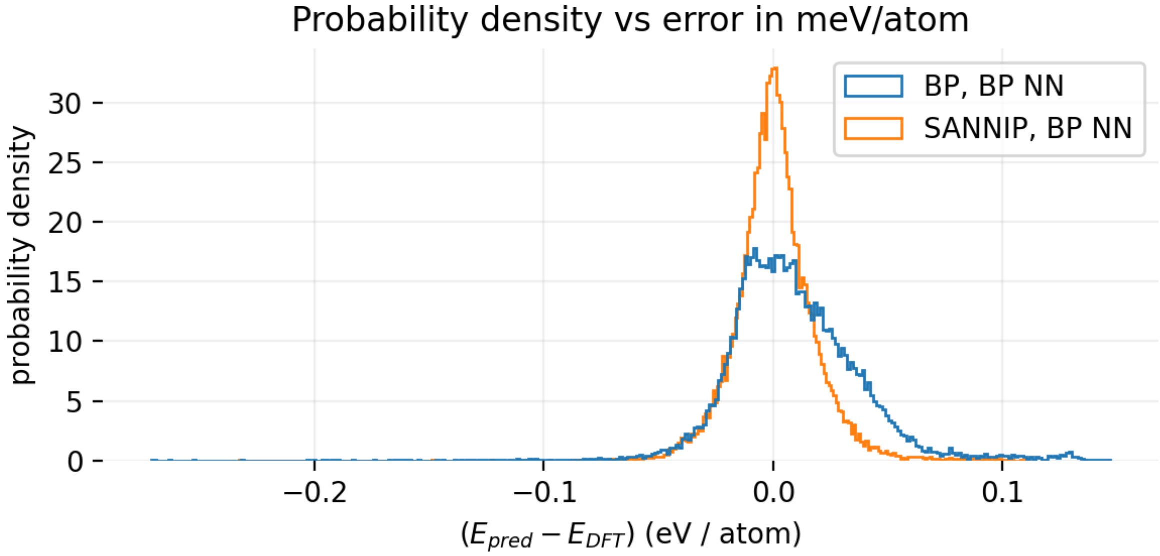


Figure 4.18: SANNIP vs BP, validation error histogram on high-geometry variation dataset.

Table 4.3 shows results for the complete set of combinations of bases and NNs. The CNN reduces RMSE slightly, but at great computational expense.

Bases	BP	BP	SANNIP	SANNIP
NN Type	BP NN	CNN	BP NN	CNN
Agg. RMSE (meV/atom)	28.52	24.63	15.47	13.97
Val. RMSE (meV/atom)	30.28	26.22	17.50	17.16
$\frac{\text{s}}{\text{epoch}}$	4.01	10.9	4.22	12.6
FOM $\times 10^3$	8.24	3.51	13.6	4.62
FOM/FOM$_{\text{BP, BP NN}}$	1.00	0.426	1.64	0.561

Table 4.3: High geometric-variation dataset RMSE comparison, SANNIP vs BP.

Fig. 4.19 shows the above tabulated results graphically. Again, SANNIP with the BP NN outperforms all other bases on a FOM basis.

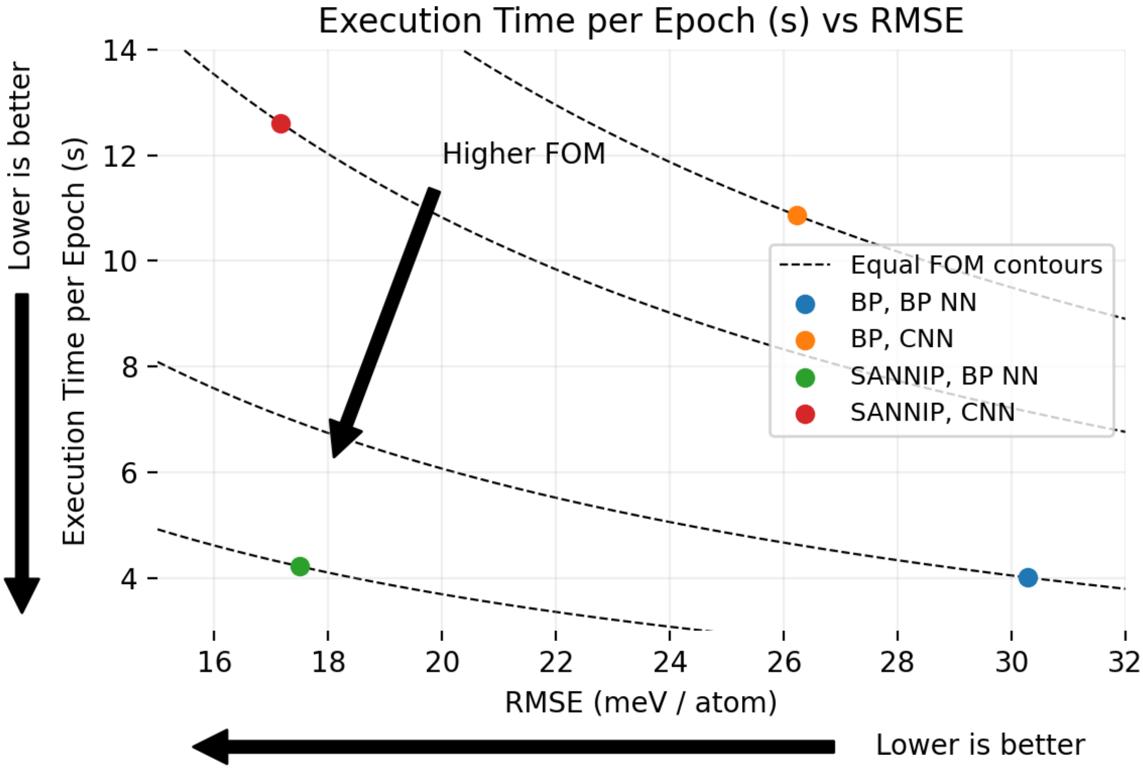


Figure 4.19: Execution time per epoch vs. RMSE across all methods applied to the low-geometry and high-geometry variation dataset.

4.3 Monte Carlo Spin Relaxations

Since the unique capability of SANNIP (compared to BP) is its direct treatment of atomic spins, a significant component of achieving POC is phenomenologically accurate physical modeling of magnetic phenomena. This work focuses on magnetic properties that are emergent and based on proper behavior of the aggregate or ensemble, because those properties to connect the Å-scale domain of IPs with macro-scale observables. By fixing a perfect lattice and relaxing its spin state from various initial conditions and at various temperatures, it is shown that SANNIP does indeed offer new functionality that is both consistent with physical observation and enabling of meaningful predictions.

4.3.1 Spin Monte Carlo

The Monte Carlo algorithm presented in this work is defined as follows:

1. Define system geometry. This is fixed throughout the simulation.
2. Define the initial atomic spins. These are set to perfect +z-aligned FM, perfect ±z-aligned AFM, or PM.
3. Define the MC move. This will be zero-th order or first order. The zero-th order move is an unbiased, random draw using same distribution as the PM initialization. The first order move is a small angle perturbation from the atom's present alignment and a random draw for magnitude (again, from $(\mu_{||\vec{s}||})$).

Once the MC process is well-defined, the main algorithm proceeds:

1. Propose an MC move.
2. Evaluate the energy of the new configuration
3. Accept the move if the new energy is lower than the pre-move energy.
4. Obtain a pseudo-random number $\in \mathcal{U}[0,1)$ and accept the move if that number is $< e^{-\frac{\Delta E}{k_B T}}$.
5. Repeat until a specified quantity of moves is accepted.

This methodology allows us to initialized a lattice to a specific spin state and allow it to relax to equilibrium, given sufficient MC moves.

Further implementation details are parameter selection (initial lattice, initial spin distribution, spin distribution as a function of MC move, etc.) and limits on move-size, because NN predictions are only valid within the training domain.

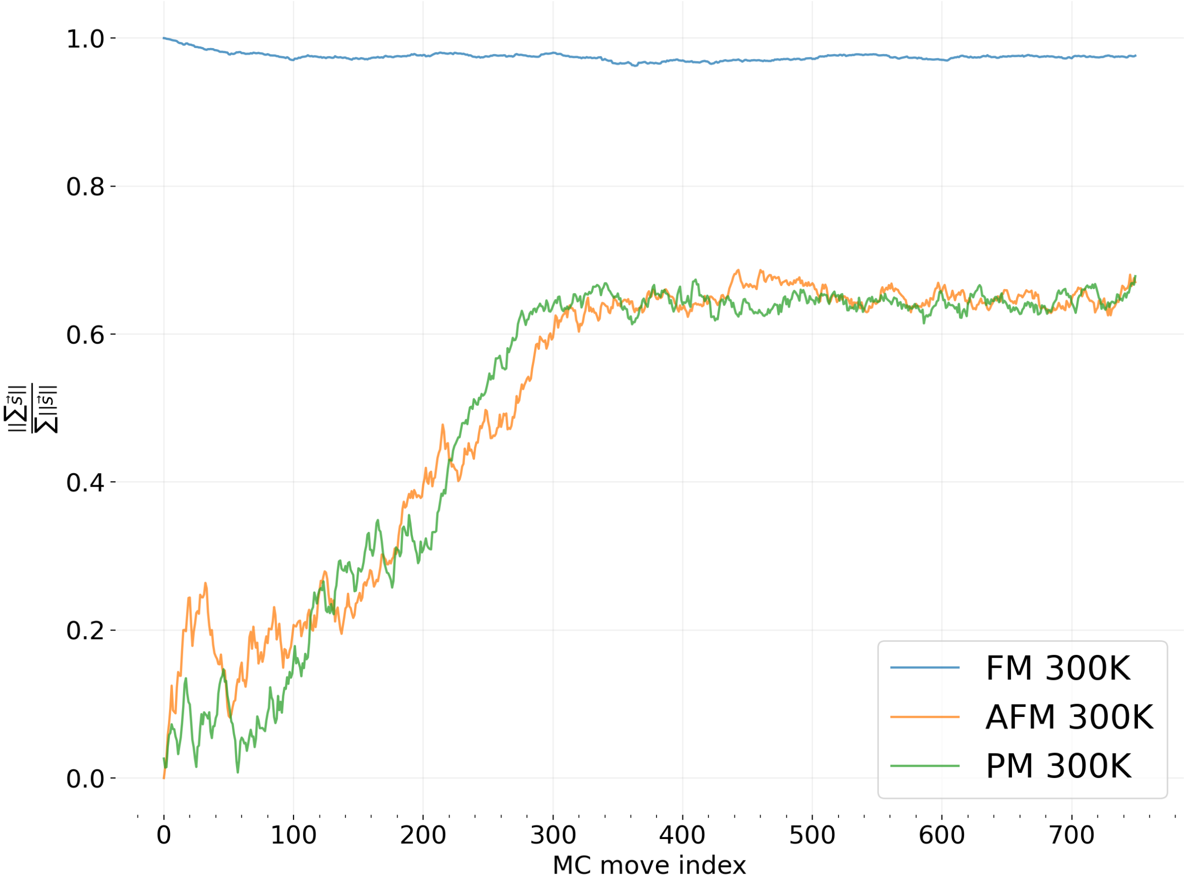
4.3.2 Spin Relaxation Consistency with Magnetic Hysteresis

Magnetic hysteresis occurs when a magnetic material is capable of maintaining different magnetization states, depending on the history of its magnetic environment and state. For example, if a sample of pure iron is magnetized in a high magnetic field and then the externally applied field strength is attenuated to zero (without changing its direction), then the iron sample will settle at and retain a certain magnetization. If on the other hand a sample of iron heated above its Curie Temperature is quenched in the absence of a magnetic field, the resulting magnetization is significantly different.

There is a large energy penalty associated with creating disordered atomic magnetic moment alignment in FM materials at low temperature, and similarly a large energy penalty associated with non-equilibrium spin magnitudes. If a strongly magnetized material is subject to external fields in the direction opposite the moment of the magnetized material, and confined so as to prevent macroscopic movement, then the material's atomic magnetic moments become disordered and frustrated, and some of that energy is transferred into lattice perturbations that are perceptible as heat. This heat is known referred to as "hysteresis loss" in many electrical engineering applications, for example, in transformer cores.

Since SANNIP is a methodology that explicitly treats spin state, it is reasonable to ask: "does SANNIP relax different initial spin conditions to different relaxed states?" Fig. 4.20 shows MC using SANNIP relaxes atomic magnetic moments to different end states, in a manner that is dependent on the initial magnetization state—as represented by the system's net magnetization. This result is consistent with physical observations of hysteresis, in that highly magnetized systems stay magnetized, and systems that relax from frustrated states become only partially magnetized.

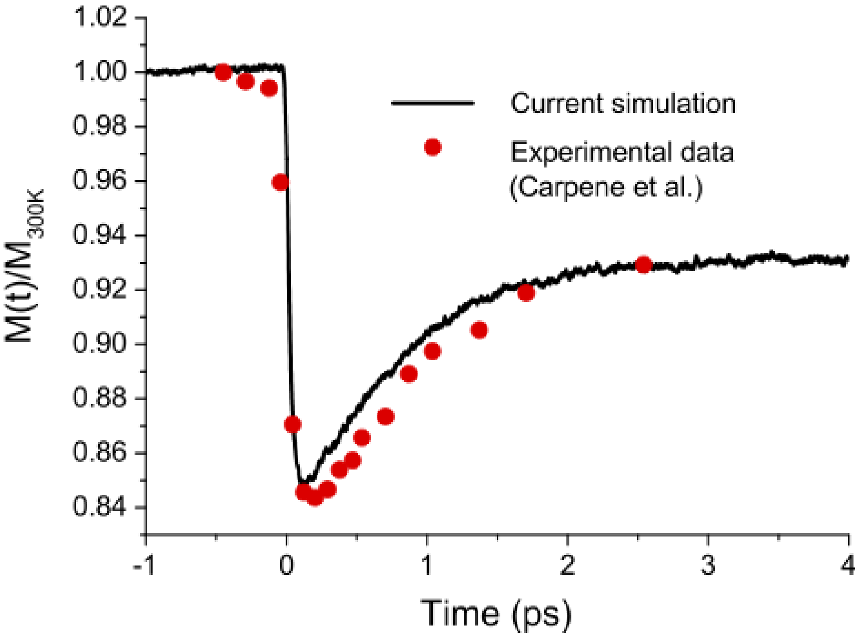
Figure 4.20: SANNIP relaxes different spin initializations to different spin end states.



As mentioned in the Introduction, Ma, Dudarev, and Woo showed magnetization as a function of time for a sample of BCC iron after a laser pulse as shown in Fig. 4.21. [52] Magnetization decreased after the laser pulse and then rebounded, but the rebound setted a lower magnetization than the initial because of increased temperature. Ma and Dudarev did not make any claims as to whether or not their result exhibits magnetic hysteresis, although they could compare the final magnetization to an SLD run started at a higher temperature. In any case, the IP that produced their result used four components:

1. the Dudarev and Derlet 2005 empirical potential, [15]
2. a fitted Heisenberg exchange function governing spin direction changes, [61]
3. a set of Landau coefficients governing spin magnitude changes, [93] and
4. an electron heat capacity relation. [94]

Figure 4.21: SPILADY laser demagnetization pulse result showing good agreement with experiment. [52]



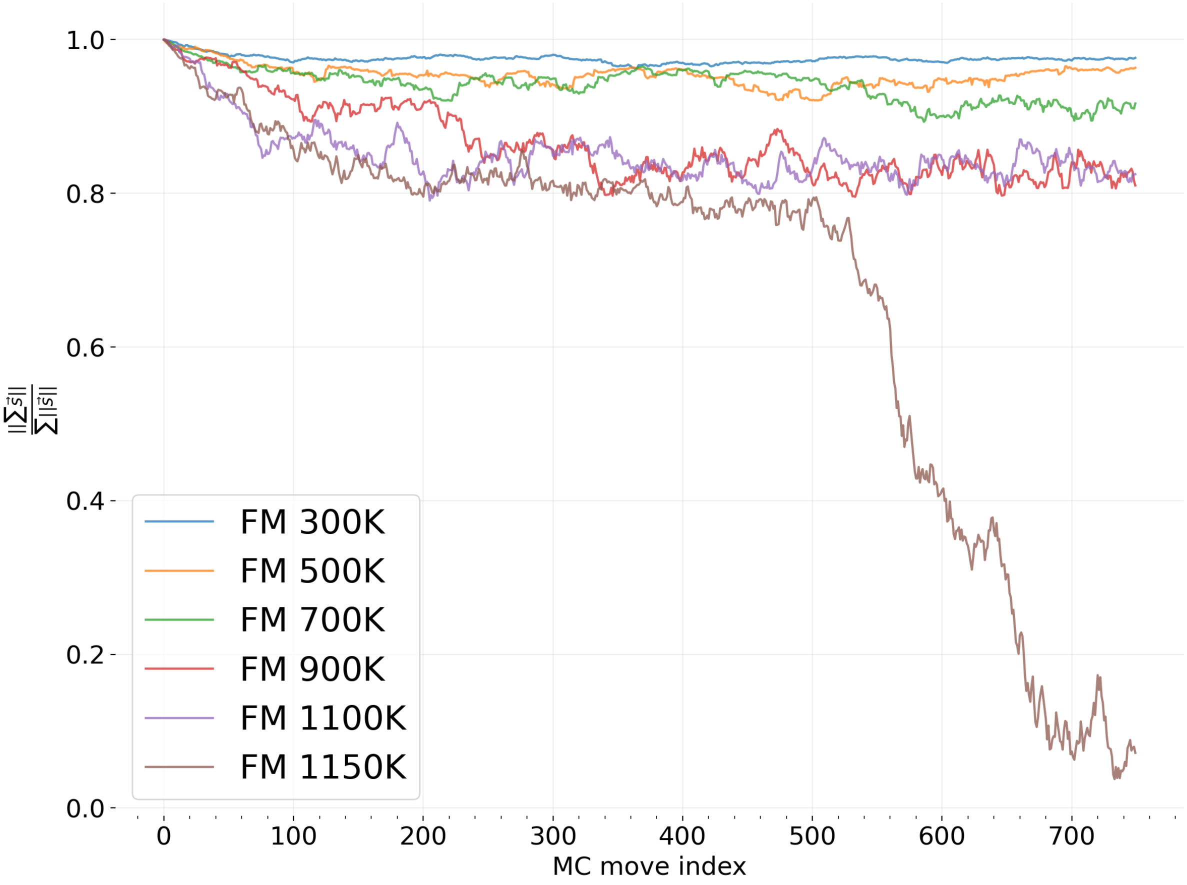
In contrast, the IP of this work is trained on features constructed from coordinates, spins and DFT energy values—and no information was injected into the model via the choice of an exchange function. SANNIP does not need to be changed to be relevant for configurations outside the domain of an arbitrary exchange function.

4.3.3 Resolution of Curie Temperature of Iron

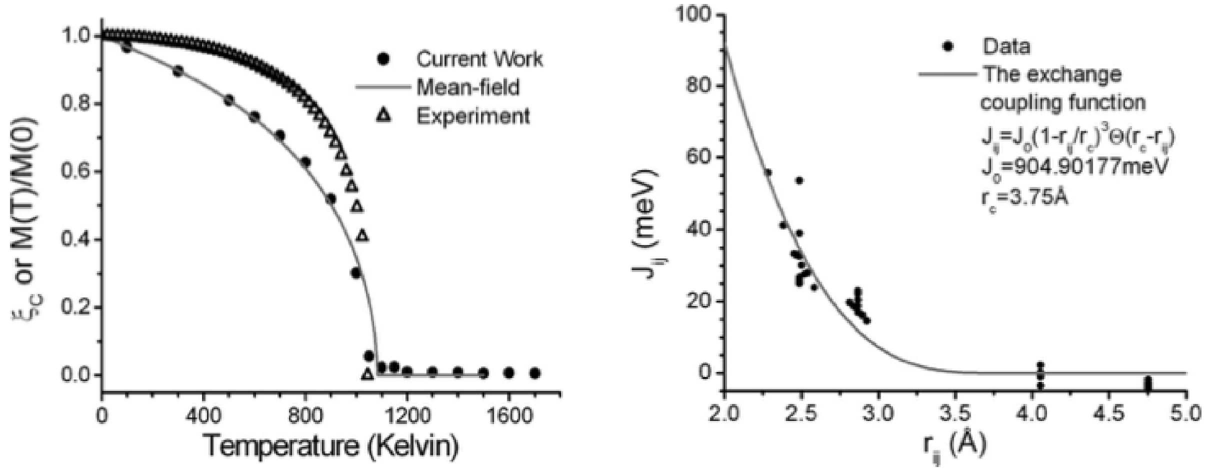
When the magnetic field strength of spontaneously magnetized FM materials are measured as a function of temperature, the magnetic field strength diminishes with increasing temperature. Once the material's temperature reaches a critical threshold it no longer exhibits spontaneous magnetism and becomes PM. This threshold is known as the material's Curie Temperature. [95]

Fig. 4.22 shows the result of MC relaxation of perfectly FM initialized systems as a function of temperature. SANNIP produces two significant results. First, the relaxed states can be rank-ordered according to their temperature, which is in agreement with physical observations. Second, and more importantly, when the MC temperature reaches 1150K, the end state becomes almost fully PM.

Figure 4.22: MC determination of Curie Point of iron.



All previous work resolving the Curie Temperature of iron using atomistic simulation has relied upon an exchange function of fixed functional form which is specific to a particular context, for example BCC iron with a FM ground state as shown in Figs. 4.23a and 4.23b. [61–63] The fitted exchange functions used ensure that their results are smooth even though they are fitted with a relatively small database compared to that of the present work. This highlights the a key difference between SANNIP and empirical spin-aware potentials—while NNs require densely sampled data to learn appropriate functional forms, their formlessness can readily adapt to any lattice type, geometric configuraiton, or magnetic ground state without need of a case-specific intermediate approximation.



(a) Ma et al. showing magnetization as a function of Temperature. [61]

(b) Ma et al. showing their exchange function and fitting data. [61]

Figure 4.23: Ma et al. work on Spin-Lattice Dynamics from 2008. [61]

Chapter 5

Conclusions and Future Work

5.1 Conclusions

This work presents novel basis functions that capture atomic magnetic moment information and featurize it for input into neural networks that can express a potential energy surface as a function of geometric and spin configurations. This work also presents novel neural network architectures that exhibit improved ability to learn and express potential energy surfaces. It is shown that the novel spin basis functions enable a transformational improvement in the numerical accuracy of BP methods in terms of energy prediction RMSE over training datasets that include the broad distribution in spin required for spin-aware NN methods to function effectively. Lastly, SANNIP learns and exhibits spin/energy relationships that are consistent with experimental observations of hysteresis, and it successfully resolves the Curie Temperature of iron to between 1100 and 1150K which is reasonable given the actual T_C of 1043K.

Fig. 5.1 shows that SANNIP using the BP NN offers an excellent balance of speed and accuracy compared to BP or SANNIP using the new CNN. This will likely appeal to a broad user base, and users who need just a little bit more accuracy can find that using the CNN.

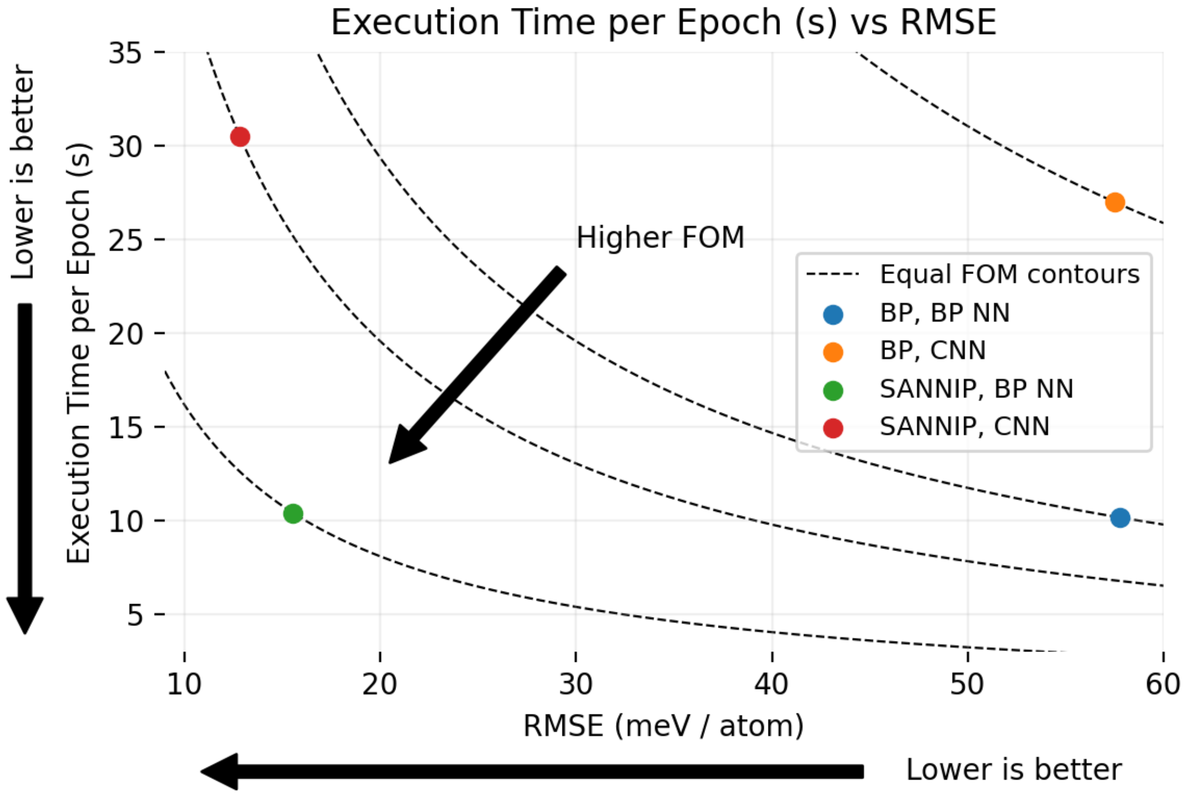


Figure 5.1: Execution time per epoch vs. RMSE across all methods applied to the high-geometry and high-spin variation dataset.

Because SANNIP does not rely on a fitted exchange function, it is has the capacity to be as general as the dataset used to train it—including configurations with defects, surfaces, voids, amorphous solid states, etc.

This work therefore concludes that the novel basis functions and neural networks are a new and significant contribution to the field of atomistic simulation and enabling of future work will now be described.

5.2 Future Work

It is commonly said that some questions, when answered, lead to more and potentially greater questions. With regard to this thesis, that appears to be the case, as outlined below.

5.2.1 Feature PCA Microstructure Characterization

Software packages such as LAMMPS and Ovito implement what is known as Common Neighbor Analysis (CNA). [55,96,97] While CNA classifies atoms by local crystal structure, feature PCA characterization could go much farther. As mentioned above in discussing PCA of BP and SANNIP features, clustering and other methods can be used to group and distinguish between geometries and coupled geometry / spin configurations—for example, identifying atoms adjacent to vacancies. [91, 92] BP and SANNIP could also be used to similar and further ends. Special basis functions also could be constructed and used to recognize particular geometric and spin configurations in a biased manner. Unbiased approaches could be used to discover new groups of configurations that are present only in specific situations or near certain defects.

This could be critical to understanding the role of spin in defect migration trajectories, radiation cascades, and defect aggregation / dissolution. The spin state during defect vacancy could be studied in an automated fashion and obtain statistically relevant conclusions that would be both tedious and difficult to perform manually. For example, it requires some care to identify vacancies and examine the spin states of all nearby atoms—but an automated method might notice patterns that a human will not. The added dimensionality of spin makes such automation even more valuable.

Methodologies such as these are not dependent on the IP used to generate configurations or trajectories; they can be applied simply as a characterization procedure.

Given sufficient study of feature PCA microcharacterization methods, a database could be created with annotated data to train ML classifiers (logistic regressor, SVM, NN, etc.).

5.2.2 Atomistic Simulation under Externally Applied Magnetic Fields

By expanding the spin bases to include explicit treatment of externally applied magnetic fields, it would be possible to directly model hysteresis and its consequences at the atomic scale. It could be useful to understand how defect formation, migration, and interactions change under various magnetic fields. One possible application could be defect removal using specific externally applied magnetic field trajectories, for example evaporating voids or dislocation loops. Conversely, it may turn out to be the case that defect formation is accelerated, or new kinds of defects form.

One potential practical aim would be to design a “defect sink” that operates in a manner simliar to cathodic protection but protects against defect accumulation instead of oxidation, by using a confluence of externally applied electric and/or magnetic fields, laser pulses, mechanical stress/strain, and phonon excitation to simultaneously (1) increase defect mobility, (2) reduce annihilation reaction energy barriers, and (3) cause defects to exit bulk lattice at surfaces.

Electroplating and surfactants are other examples of using electrical and chemical manipulations of the PES at material interfaces.

Beyond that concept, magneto-laser-strain effects may be useful in producing large single crystals or otherwise dissolving grain boundaries. This could enable lower cost casting and machining to form complex and precise shapes and still producing single crystals.

Processes may be envisioned that compete with hot isostatic processing (HIP), allowing similar results at lower cost and by smaller facilities.

5.2.3 New Spin Bases

While meaningful success has been achieved in this work, the author does not expect this to be the final word regarding the development of spin bases (even forgetting about externally applied fields). It could be that there are relationships specific to certain structures that could be detected by purpose built bases which enable higher resolution of key properties. For example, it may be that di-interstitials are best detected by a particular quantitative measure that is not emphasized by the current set of spin bases.

As of this writing, several specific spin bases are proposed. They are inspired by the effort to quantify the magnetic character of the DFT training data.

The first prospective basis is an analog of system net spin, a GRBF transform of the sum of the product of spins and cutoffs:

$$S_i^{F1} = e^{-\eta(\sum_{j\neq i}^{N} ||\vec{S}_j|| f_c(R_{ij}) - \alpha)^2} \tag{5.1}$$

The second prospective basis is a GRBF transform of the magnitude of the sum of the product of the spins and cutoffs:

$$S_i^{F2} = e^{-\eta(|| \sum_{j\neq i}^{N} \vec{S}_j f_c(R_{ij}) || - \alpha)^2} \tag{5.2}$$

The third prospective basis is an alternative to Spin Basis 001 S_i^2, and a transform of the 000 S_i^2:

$$S_i^{F3} = e^{-\eta(\vec{S}_i \cdot \sum_{j\neq i}^{N} \vec{S}_j f_c(R_{ij}) - \alpha)^2} \tag{5.3}$$

Fourth is a proxy of magnetization inside the cutoff:

$$S_i^{F4} = \frac{|| \sum_j^N \vec{S}_j f_c(R_{ij}) ||}{\sum_j^N ||\vec{S}_j|| f_c(R_{ij})} \tag{5.4}$$

The fifth proposed basis is a proxy of average polar angle inside the cutoff:

$$S_i^{F5} = \sum_k^N \frac{\vec{S}_k \cdot \sum_j^N \vec{S}_j f_c(R_{ij})}{||\vec{S}_k|| || \sum_j^N \vec{S}_j f_c(R_{ij}) ||} \tag{5.5}$$

Lastly, a proxy of variance of polar angle inside the cutoff is proposed:

$$S_i^{F6} = \sum_i^N \left(\frac{\vec{S}_i \cdot \sum_{j \neq i}^N \vec{S}_j f_c(R_{ij})}{||\vec{S}_i|| || \sum_{j \neq i}^N \vec{S}_j f_c(R_{ij})||} \right)^2 \tag{5.6}$$

5.2.4 Combination of Spin Bases and Gauss-Fourier Bases

While presented in the appendices, the Gauss-Fourier (GF) bases yield improvement in learning geometry compared to conventional BP bases. They could be combined with the spin bases and evaluated for efficacy. This is left to future work because it is not necessary to prove the validity of the spin bases.

Since GF angular bases can be constructed to detect specific coordinations, it may be the case that special spin bases can be constructed along similar lines, or additionally that geometric and spin bases could be explicitly designed to work together to detect more complex phenomena.

5.2.5 Spin-Aware GAP & SNAP

Just as GAP and SNAP use 4D spherical harmonics to represent the projection of the scalar field for position, we propose that the same approach be applied to each component of spin, and the resultant featurization be input into a: linear regressor, NN, and kernel regression.

5.2.6 DFT Studies of Spin-State Energy Barriers

In the complex, multi-body environment of an atom inside a 6 Å cutoff with up to $>$ 80 neighbors, the displacements of neighbors from their geometric equilibrium positions is likely to significantly influence the spin-space location of spin ground state and the quantity, spin-space proximity, energy-barriers to, and energy levels of neighboring spin-space minima. In lattices more characteristic of low temperature, the energy barrier height and spin-space distances between spin-space local minima may be large. However, those barriers and distances may be lower in lattices more characteristic of higher temperature—potentially yielding atomistic insight into the phenomenon of the Curie Temperature as a confluence of lower energy barriers between spin states and higher prevalence of excited states. Detailed investigation along these lines could be a subject for DFT studies using constrained magnetic moment calculations which is left to future work.

5.2.7 Hyper-Parameter Optimization

Detailed, case-specific hyper-parameter optimization is left to future work. While the hyper-parameters presented are sufficient for POC, it is certainly worth exploring an accuracy and execution time comparison between fully-optimized applications of BP and SANNIP. Both conventional ML hyper-parameters and featurization basis set parameters should be considered.

5.2.8 Point and Extended Defect Modeling in α-Iron

The methodology of this thesis can be used to investigate the configuration and energetics of point defects, screw and edge dislocations, grain boundaries, and other structures to compare results against existing literature using EAM and other methods.

5.2.9 Expansion to Multi-Component Systems

While this work has focused on the particular unary system if pure iron, future work to expand its application space to include multiple elements is an immediate next step. The MIMONN described in Appendix E is envisioned as one way to proceed in that endeavor.

Bibliography

[1] G. H. Ebel. Reliability physics in electronics: a historical view. *IEEE Transactions on Reliability*, 47(3):SP379–SP389, Sep. 1998.

[2] A. G. Crocker. Defects in crystalline materials and their relation to mechanical properties. *Experimental Mechanics*, 6(5):266–272, May 1966.

[3] I. M. Neklyudov, O. V. Borodin, V. V. Bryk, and V. N. Voyevodin. Problem of radiation resistance of structural materials of nuclear power. *Progress in HighEnergy Physics and Nuclear Safety*, pages 259–277, Dordrecht, 2009. Springer Netherlands.

[4] A. Einstein, Translated R. W. Lawson, Published H. Holt Relativity: the special and general theory, 1947.

[5] Feynman, R. P. *The Feynman lectures on physics.* Reading, Mass. : Addison-Wesley Pub. Co., c1963-1965., c1963-1965.

[6] I. Newton. *Philosophiae naturalis principia mathematica.* J. Societatis Regiae ac Typis J. Streater, 1687.

[7] Y. Ikeda, B. Grabowski, and F. Körmann. Ab initio phase stabilities and mechanical properties of multicomponent alloys: A comprehensive review for high entropy alloys and compositionally complex alloys. *Materials Characterization*, 147:464–511, 2019.

[8] L. A. Zepeda-Ruiz, A. Stukowski, T. Oppelstrup, N. Bertin, N. R. Barton, R. Freitas, and V. V. Bulatov. Atomistic insights into metal hardening. *Nature Materials*, 20(3):315–320, 2021.

[9] E. Fermi, P. Pasta, S. Ulam, and M. Tsingou. Studies of non-linear problems. *Los Alamos Technical Report*, 5 1955.

[10] S. B. Sinnott and D. W. Brenner. Three decades of many-body potentials in materials research. *MRS Bulletin*, 37(5):469–473, 2012.

[11] N. Metropolis. The beginning of the monte carlo method. *Los Alamos Science*, pages 125–130, 1987.

[12] P. Zhang and D. R Trinkle. Database optimization for empirical interatomic potential models. *Modelling and Simulation in Materials Science and Engineering*, 23(6):065011, 2015.

[13] M. Planck. Ueber das gesetz der energieverteilung im normalspectrum. *Annalen der Physik*, 309(3):553–563, 1901.

[14] J. A. Pople. Nobel lecture: Quantum chemical models. *Reviews of Modern Physics*, 71(5):1267–1274, 10 1999.

[15] S. L. Dudarev and P. M. Derlet. A ‘magnetic’ interatomic potential for molecular dynamics simulations. *Journal of Physics: Condensed Matter*, 17(44):7097–7118, 2005.

[16] G. J. Ackland. Two-band second moment model for transition metals and alloys. *Journal of Nuclear Materials*, 351(1):20–27, 2006.

[17] M.-C. Marinica, F. Willaime, and J.-P. Crocombette. Irradiation-induced formation of nanocrystallites with $c15$ laves phase structure in bcc iron. *Phys. Rev. Lett.*, 108:025501, 2012.

[18] G. Rollmann, P. Entel, and S. Sahoo. Competing structural and magnetic effects in small iron clusters. *Computational Materials Science*, 35(3):275–278, 2006.

[19] P. Ma and S. L. Dudarev. Constrained density functional for noncollinear magnetism. *Physical Review B*, 91(5):054420–, 02 2015.

[20] T. P. C. Klaver, R. Drautz, and M. W. Finnis. Magnetism and thermodynamics of defect-free fe-cr alloys. *Physical Review B*, 74(9):094435–, 09 2006.

[21] J. Tranchida, S.J. Plimpton, P. Thibaudeau, and A.P. Thompson. Massively parallel symplectic algorithm for coupled magnetic spin dynamics and molecular dynamics. *Journal of Computational Physics*, 372:406–425, 2018.

[22] G. J. Ackland, D. J. Bacon, A. F. Calder, and T. Harry. Computer simulation of point defect properties in dilute Fe-Cu alloy using a many-body interatomic potential. *Philosophical Magazine A*, 75(3):713–732, 1997.

[23] G. Bonny, N. Castin, and D. Terentyev. Interatomic potential for studying ageing under irradiation in stainless steels: the FeNiCr model alloy. *Modelling and Simulation in Materials Science and Engineering*, 21(8):085004, 2013.

[24] G. Bonny, R. C. Pasianot, N. Castin, and L. Malerba. Ternary Fe–Cu–Ni many-body potential to model reactor pressure vessel steels: First validation by simulated thermal annealing. *Philosophical Magazine*, 89(34-36):3531–3546, 2009.

[25] G. Bonny, R. C. Pasianot, and L. Malerba. Fe–Ni many-body potential for metallurgical applications. *Modelling and Simulation in Materials Science and Engineering*, 17(2):025010, 2009.

[26] H. Chamati, N.I. Papanicolaou, Y. Mishin, and D.A. Papaconstantopoulos. Embedded-atom potential for fe and its application to self-diffusion on fe(100). *Surface Science*, 600(9):1793 – 1803, 2006.

[27] L. Proville, D. Rodney, and M.-C. Marinica. Quantum effect on thermally activated glide of dislocations. *Nat. Mater.*, 11:845–849, 2012.

[28] M. I. Mendelev, S. Han, D.J. Srolovitz, G.J. Ackland, D.Y. Sun, and M. Asta. Development of new interatomic potentials appropriate for crystalline and liquid iron. *Phil. Mag.*, 83(35):3977–3994, 2003.

[29] X. W. Zhou, H. N. G Wadley, R. A. Johnson, D. J. Larson, N. Tabat, A. Cerezo, A. K. Petford-Long, G. D. W Smith, P. H. Clifton, R. L. Martens, and T. F. Kelly. Atomic scale structure of sputtered metal multilayers. *Acta Materialia*, 49(19):4005–4015, 11 2001.

[30] X. W. Zhou, R. A. Johnson, and H. N. G. Wadley. Misfit-energy-increasing dislocations in vapor-deposited CoFe/NiFe multilayers. *Physical Review B*, 69:144113, 2004.

[31] S. L. Dudarev and P. M. Derlet. Erratum: A 'magnetic' interatomic potential for molecular dynamics simulations. *Journal of Physics: Condensed Matter*, 19(23):239001, 2007.

[32] L. A. Girifalco and V. G. Weizer. Application of the Morse potential function to cubic metals. *Physical Review*, 114:687–690, 1959.

[33] M. Müller, P. Erhart, and K. Albe. Analytic bond-order potential for bcc and fcc iron—comparison with established embedded-atom method potentials. *Journal of Physics: Condensed Matter*, 19(32):326220, 2007.

[34] R. S. Elliott and E. B. Tadmor. Knowledgebase of Interatomic Models (KIM) application programming interface (API). `https://openkim.org/kim-api`, 2011.

[35] E. B. Tadmor, R. S. Elliott, J. P. Sethna, R. E. Miller, and C. A. Becker. The potential of atomistic simulations and the Knowledgebase of Interatomic Models. *JOM*, 63(7):17, 2011.

[36] P. Hohenberg and W. Kohn. Inhomogeneous electron gas. *Physical Review*, 136(3B):B864–B871, 11 1964.

[37] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140:A1133–A1138, 1965.

[38] U. v. Barth and L. Hedin. A local exchange-correlation potential for the spin polarized case. i. *Journal of Physics C: Solid State Physics*, 5(13):1629–1642, jul 1972.

[39] M.M. Pant and A.K. Rajagopal. Theory of inhomogeneous magnetic electron gas. *Solid State Communications*, 10(12):1157–1160, 1972.

[40] J Kubler, K H Hock, J Sticht, and A R Williams. Density functional theory of non-collinear magnetism. *Journal of Physics F: Metal Physics*, 18(3):469–483, mar 1988.

[41] Juan E. Peralta, Gustavo E. Scuseria, and Michael J. Frisch. Noncollinear magnetism in density functional calculations. *Physical Review B*, 75(12):125119–, 03 2007.

[42] P. H. Dederichs, S. Blügel, R. Zeller, and H. Akai. Ground states of constrained systems: Application to cerium impurities. *Physical Review Letters*, 53(26):2512–2515, 12 1984.

[43] T. Tanaka and Y. Gohda. Prediction of the curie temperature considering the dependence of the phonon free energy on magnetic states. *npj Computational Materials*, 6(1):184, 2020.

[44] Y. Zuo, C. Chen, X. Li, Z. Deng, Y. Chen, J. Behler, G. Csányi, A. V. Shapeev, A. P. Thompson, M. A. Wood, and S. P. Ong. Performance and cost assessment of machine learning interatomic potentials. *The Journal of Physical Chemistry A*, 124(4):731–745, 01 2020.

[45] J. Behler and M. Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.*, 98:146401, 2007.

[46] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi. Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. *Physical Review Letters*, 104(13):136403–, 04 2010.

[47] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, USA, 2014.

[48] D. Dragoni, T. D. Daff, G. Csányi, and N. Marzari. Achieving dft accuracy with a machine-learning interatomic potential: Thermomechanics and defects in bcc ferromagnetic iron. *Physical Review Materials*, 2(1):013808–, 01 2018.

[49] A.P. Thompson, L.P. Swiler, C.R. Trott, S.M. Foiles, and G.J. Tucker. Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials. *Journal of Computational Physics*, 285:316–330, 2015.

[50] M. A. Wood and A. P. Thompson. Extending the accuracy of the snap interatomic potential form. *The Journal of Chemical Physics*, 148(24):241721, 2018.

[51] A. Shapeev. Moment tensor potentials: A class of systematically improvable interatomic potentials. *Multiscale Model. Simul.*, 14:1153–1173, 2016.

[52] P. Ma, S.L. Dudarev, and C.H. Woo. Spilady: A parallel cpu and gpu code for spin–lattice magnetic molecular dynamics simulations. *Computer Physics Communications*, 207:350–361, 2016.

[53] W. Heisenberg. Mehrkörperproblem und resonanz in der quantenmechanik. *Zeitschrift für Physik*, 38(6):411–426, 1926.

[54] P. Adrien, M. Dirac, and R. H. Fowler. On the theory of quantum mechanics. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 112(762):661–677, 1926.

[55] S. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*, 117(1):1–19, 1995.

[56] S, Nikolov, M. A. Wood, A. Cangi, J. Maillet, M.C. Marinica, A. P. Thompson, M.P. Desjarlais, and J. Tranchida. Quantum-accurate magneto-elastic predictions with classical spin-lattice dynamics, 2021.

[57] J. W. Gibbs. *Elementary principles in statistical mechanics developed with especial reference to the rational foundation of thermodynamics.* C. Scribner,, New York :, 1902.

[58] I. Novikov, B. Grabowski, F. Kormann, and A. Shapeev. Machine-learning interatomic potentials reproduce vibrational and magnetic degrees of freedom, 2020.

[59] Lammps benchmarks webpage. https://lammps.sandia.gov/bench.html

[60] L. P. Kaelbling. 6.862: Lecture on convolutional neural networks. Introduction to topic at beginning of class, 2018.

[61] P. Ma, C. H. Woo, and S. L. Dudarev. Large-scale simulation of the spin-lattice dynamics in ferromagnetic iron. *Physical Review B*, 78(2):024434–, 07 2008.

[62] P. Ma, S. L. Dudarev, and J. S. Wróbel. Dynamic simulation of structural phase transitions in magnetic iron. *Physical Review B*, 96(9):094418–, 09 2017.

[63] Y. Zhou, J. Tranchida, Y. Ge, J. Murthy, and T. S. Fisher. Atomistic simulation of phonon and magnon thermal transport across the ferromagnetic-paramagnetic transition. *Physical Review B*, 101(22):224303–, 06 2020.

[64] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden,

M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[65] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, J. Reynolds, A. Melnikov, N. Lunova, and O. Reblitz-Richardson. Pytorch captum. `https://github.com/pytorch/captum`, 2019.

[66] K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

[67] P. Horowitz and W. Hill. *The Art of Electronics*. Cambridge University Press, Cambridge, 3 edition, 2015.

[68] J E Lennard-Jones. On the determination of molecular fields.—i. from the variation of the viscosity of a gas with temperature. *Proc. R. Soc. Lond. A*, 106(738):441–462, 1924.

[69] J E Lennard-Jones. On the determination of molecular fields. —ii. from the equation of state of a gas. *Proc. R. Soc. Lond. A*, 106(738):463–477, 1924.

[70] N. Artrith and A. Urban. An implementation of artificial neural-network potentials for atomistic materials simulations: Performance for tio2. *Computational Materials Science*, 114:135–150, 2016.

[71] J. Behler. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *The Journal of Chemical Physics*, 134(7):074106, 2011.

[72] G. Kresse and J. Furthmüller. Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. *Computational Materials Science*, 6(1):15–50, 1996.

[73] P. C. Kainen, V. Kůrková, and M. Sanguineti. Complexity of gaussian-radial-basis networks approximating smooth functions. *Journal of Complexity*, 25(1):63–74, 2009.

[74] R. L. Bramblett, R. I. Ewing, and T. W. Bonner. A new type of neutron spectrometer. *Nuclear Instruments and Methods*, 9(1):1–12, 1960.

[75] R. Khabaz and H. M. Hakimabad. Determination of 241am-be spectra using bonner sphere spectrometer by applying shadow cone technique in calibration. *Journal of Applied Sciences*, 11(15):2849–2854, 2011.

[76] F. H. Stillinger and T. A. Weber. Computer simulation of local order in condensed phases of silicon. *Physical Review B*, 31(8):5262–5271, 04 1985.

[77] N. Artrith, T. Morawietz, and J. Behler. High-dimensional neural-network potentials for multicomponent systems: Applications to zinc oxide. *Physical Review B*, 83(15):153101–, 04 2011.

[78] N. Artrith and J. Behler. High-dimensional neural network potentials for metal surfaces: A prototype study for copper. *Phys. Rev. B*, 85:045439, 2012.

[79] N. Artrith, A. Urban, and G. Ceder. Efficient and accurate machine-learning interpolation of atomic energies in compositions with many species. *Phys. Rev. B*, 96:014112, 2017.

[80] R. Lot, F. Pellegrini, Y. Shaidu, and E. Küçükbenli. Panna: Properties from artificial neural network architectures. *Computer Physics Communications*, 256:107402, 2020.

[81] I. Dzyaloshinsky. A thermodynamic theory of “weak”ferromagnetism of antiferromagnetics. *Journal of Physics and Chemistry of Solids*, 4(4):241–255, 1958.

[82] T. Moriya. Anisotropic superexchange interaction and weak ferromagnetism. *Physical Review*, 120(1):91–98, 10 1960.

[83] D. Treves and S. Alexander. Observation of antisymmetric exchange interaction in yttrium orthoferrite. *Journal of Applied Physics*, 33(3):1133–1134, 1962.

[84] A. N. Bogdanov and U. K. Rößler. Chiral symmetry breaking in magnetic thin films and multilayers. *Physical Review Letters*, 87(3):037203–, 06 2001.

[85] U. K. Rößler, A. N. Bogdanov, and C. Pfleiderer. Spontaneous skyrmion ground states in magnetic metals. *Nature*, 442(7104):797–801, 2006.

[86] B. Dupé, M. Hoffmann, C. Paillard, and S. Heinze. Tailoring magnetic skyrmions in ultra-thin transition metal films. *Nature Communications*, 5(1):4030, 2014.

[87] J. D. Jackson. *Classical Electrodynamics*. Wiley, New York, NY, 3rd ed. edition, 1999.

[88] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[89] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, pages 1097–1105, Red Hook, NY, USA, 2012.

[90] M. Liu and J. R. Kitchin. Singlenn: Modified behler–parrinello neural network with shared weights for atomistic simulations with transferability. *The Journal of Physical Chemistry C*, 124(32):17811–17818, 08 2020.

[91] F. J. Domínguez-Gutiérrez, J. Byggmästar, K. Nordlund, F. Djurabekova, and U. von Toussaint. On the classification and quantification of crystal defects after energetic bombardment by machine learned molecular dynamics simulations. *Nuclear Materials and Energy*, 22:100724, 2020.

[92] F. J. Domínguez-Gutiérrez and U. von Toussaint. On the detection and classification of material defects in crystalline solids after energetic particle impact simulations. *Journal of Nuclear Materials*, 528:151833, 2020.

[93] P. Ma and S. L. Dudarev. Dynamic magnetocaloric effect in bcc iron and hcp gadolinium. *Physical Review B*, 90(2):024425–, 07 2014.

[94] D M Duffy and A M Rutherford. Including the effects of electronic stopping and electron–ion interactions in radiation damage simulations. *Journal of Physics: Condensed Matter*, 19(1):016207, 2006.

[95] N. W. Ashcroft and N. D. Mermin. *Solid State Physics*. Holt-Saunders, 1976.

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

[97] A. Stukowski. Visualization and analysis of atomistic simulation data with OVITO-the Open Visualization Tool. *Modeling and Simulation in Materials Science and Engineering*, 18(1), 2010.

Appendix A

Basis Function Parameter Sets

There are several basis sets that were used over the course of this thesis. The BP comparator sets are taken from literature. We have created and tested over 100 different basis sets and parameter sets, and present the most significant cases.

A.1 Spin-agnostic Fe, BP comparator to SANNIP

The radial parameters are all possible combinations of:

$$\eta \in \{1.5, 6.25, 25.0\} \tag{A.1}$$

$$R_s \in \{1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0\}\text{Å} \tag{A.2}$$

$$R_c = 6.0\text{Å} \tag{A.3}$$

The angular parameters are all possible combinations of:

$$\eta \in \{0.000357, 0.028569, 0.089277\} \tag{A.4}$$

$$\lambda \in \{-1, 1\} \tag{A.5}$$

$$\xi \in \{1, 2, 4\} \tag{A.6}$$

$$R_c = 6.0\text{Å} \tag{A.7}$$

A.2 Spin-aware Fe, SANNIP

SANNIP results presented in this thesis used the same geometric bases and parameters as A.1, and added 8 S_i^1 bases and 12 S_i^2 bases. The S_i^1 bases all use $\eta_1 = 0.035711$, and the following list of (ξ, α_1):

$$(1, 0.0),\ (2, 0.0),\ (1, 1.0),\ (2, 1.0),\ (1, 2.0),\ (2, 4.0),\ (1, 3.0),\ (2, 9.0) \tag{A.8}$$

SANNIP S_i^2 basis parameters were all combinations of:

$$\eta_2 \in \{0.000357, 0.035711, 0.028569, 0.089277\} \tag{A.9}$$

$$\alpha_2 \in \{0.0, 35.0, 70.0\} \tag{A.10}$$

$$R_c = 6.0 \tag{A.11}$$

Appendix B

Dataset Manifest

Below, “m” is the number of samples in the dataset, and “n” is the number of atoms per sample.

ID	m	n	size	type	potential	Description
013	700	54	3x3x3	BCC	7	Full lattice Tour de Fer, FM aligned
014	700	127	4x4x4	BCC	7	Mono-vacancy Tour de Fer, FM aligned
015	700	126	4x4x4	BCC	7	Di-vacancy Tour de Fer, FM aligned
016	700	129	4x4x4	BCC	7	SIA110 Tour de Fer, FM aligned
017	700	129	4x4x4	BCC	7	SIA111 Tour de Fer, FM aligned
018	700	129	4x4x4	BCC	7	SIA 100 Tour de Fer, FM aligned
019	700	108	3x3x3	FCC	7	Full lattice Tour de Fer, FM aligned

ID	m	n	size	type	potential	Description
020	2000	16	2x2x2	BCC	7	MM = 0 0 4.0
021	2000	32	2x2x2	FCC	7	MM = 0 0 4.0
022	2000	16	2x2x2	BCC	7	MM = 111r
023	2000	32	2x2x2	FCC	7	MM = 111r
024	2000	16	2x2x2	BCC	7	MM = 000r
025	2000	32	2x2x2	FCC	7	MM = 000r

ID	m	n	size	type	potential	Description
034	705	54	3x3x3	BCC	0	001
035	705	54	3x3x3	BCC	0	1
036	705	54	3x3x3	BCC	0	2
037	705	54	3x3x3	BCC	0	000r
038	705	54	3x3x3	BCC	0	1.5
039	705	54	3x3x3	BCC	0	4

ID	m	n	size	type	potential	Description
040	5000	16	2x2x2	BCC	0	300-10000K — FM
041	5000	16	2x2x2	BCC	0	300-10000K — 1
042	5000	16	2x2x2	BCC	0	300-10000K — 2
043	5000	16	2x2x2	BCC	0	300-10000K — PM
044	5000	16	2x2x2	BCC	0	300-10000K — 1.5
045	5000	16	2x2x2	BCC	0	300-10000K — 4

ID	m	n	size	type	potential	Description
056	705	54	3x3x3	BCC	0	Tour de Fer, AFM + -
057	705	54	3x3x3	BCC	0	Tour de Fer, AFM + + - -
058	705	54	3x3x3	BCC	0	Tour de Fer, AFM random
059	5000	16	2x2x2	BCC	0	300-10000K — AFM + -
060	5000	16	2x2x2	BCC	0	300-10000K — AFM + + - -
061	5000	16	2x2x2	BCC	0	300-10000K — AFM random

Nanowire cases:

ID	m	n	size	type	potential	Description
062	2000	20	$(12\text{Å})^3$	nanowire	0	FM
063	2000	20	$(12\text{Å})^3$	nanowire	0	1
064	2000	20	$(12\text{Å})^3$	nanowire	0	1.5
065	2000	20	$(12\text{Å})^3$	nanowire	0	2
066	2000	20	$(12\text{Å})^3$	nanowire	0	4
067	2000	20	$(12\text{Å})^3$	nanowire	0	PM
068	2000	20	$(12\text{Å})^3$	nanowire	0	AFM + -
069	2000	20	$(12\text{Å})^3$	nanowire	0	AFM + + - -
070	2000	20	$(12\text{Å})^3$	nanowire	0	AFM random

Appendix C

Novel Spin-Agnostic Bases

In this appendix we present new and novel basis functions that we had developed prior to our work on spin aware basis functions.

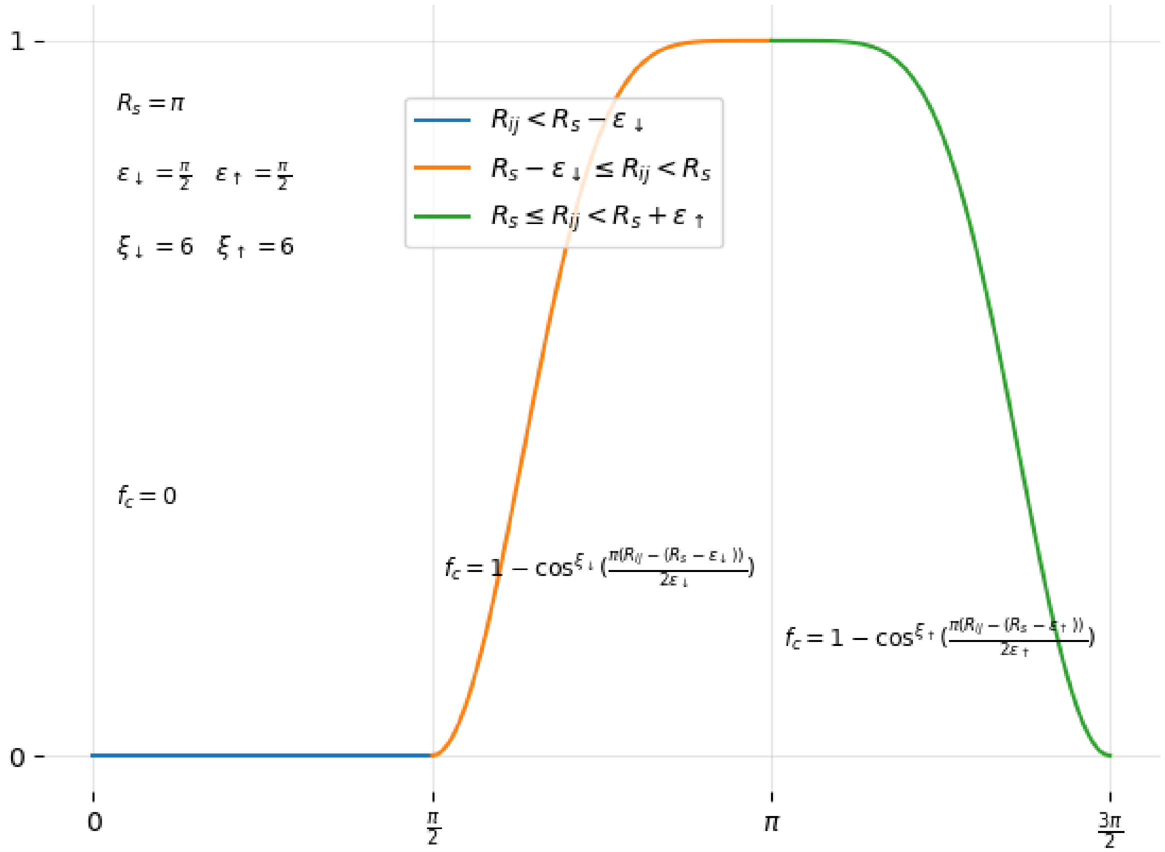
C.1 "Fourier Spline" (FS)

While the conventional BP cutoff has proven merit, its form is arbitrary. Therefore, we explore other ways to meet the need to smoothly limit the domain of a basis function. Before embarking on this task, let's consider some guiding principles.

First, we may wish to have a cutoff that allows us to focus on particular radial subdomains and isolate them from others. In reactor physics something similar is accomplished for cross-sections by integrating over several energy ranges to obtain group cross-sections. The BP cutoff at best gives a radial picture somewhat analogous to the neutron spectrum produced by Bonner Spheres, which are good and clearly interpretable by the BP NN—but we postulate that some of the NN's capacity is consumed by translating the Bonner Sphere-likeness instead of having inputs more directly related to the energy calculation.

Second, we may wish to alter the functional form of the underlying basis. The functional form of the BP cutoff is fixed; we wish to enable explicit control of asymmetry.

Figure C.1: “Fourier-Spline” presented in this work.



Second, the cutoff function can also in principle introduce counterproductive distortion of the basis function that it is cutting off. For example, the cutoff’s derivative can sometimes have its maximum absolute magnitude precisely in the region where the underlying basis is most sensitive and active in capturing information. For example, if the BP cutoff’s R_x is equal to the GRBF’s offset R_s then the shape of the GRBF is distorted maximally—and distorted differently from the GRBFs with other offsets.

$$\text{Parameters: } \epsilon_{\downarrow}, R_s, \epsilon_{\uparrow}, \xi_{\downarrow}, \xi_{\uparrow} \tag{C.1}$$

$$f_B\left(R_{ij}\right) = \begin{cases} 0 & \text{for} \quad R_{ij} < R_s - \epsilon_{\downarrow} \\ 1 - \cos^{\xi_{\downarrow}}\left(\frac{\pi(R_{ij}-(R_s-\epsilon_{\downarrow}))}{2\epsilon_{\downarrow}}\right) & \text{for} \quad R_s - \epsilon_{\downarrow} \leq R_{ij} < R_s \\ 1 - \cos^{\xi_{\uparrow}}\left(\frac{\pi(R_{ij}-(R_s-\epsilon_{\uparrow}))}{2\epsilon_{\uparrow}}\right) & \text{for} \quad R_s \leq R_{ij} < R_s + \epsilon_{\uparrow} \\ 0 & \text{for} \quad R_s + \epsilon_{\uparrow} > R_{ij} \end{cases} \tag{C.2}$$

$$\xi_{\downarrow}, \xi_{\uparrow} \in 2\mathbb{N} \quad \text{(even numbers)} \tag{C.3}$$

$$\frac{\partial f_B}{\partial R_{ij}} = \begin{cases} 0 & \text{for} \quad R_{ij} < R_s - \epsilon_{\downarrow} \\ \xi_{\downarrow}\frac{\pi}{2\epsilon_{\downarrow}} \sin\left(\frac{\pi(R_{ij}-(R_s-\epsilon_{\downarrow}))}{\epsilon_{\downarrow}}\right) \cos^{\xi_{\downarrow}-2}\left(\frac{\pi(R_{ij}-(R_s-\epsilon_{\downarrow}))}{2\epsilon_{\downarrow}}\right) & \text{for} \quad R_s - \epsilon_{\downarrow} \leq R_{ij} < R_s \\ \xi_{\uparrow}\frac{\pi}{2\epsilon_{\uparrow}} \sin\left(\frac{\pi(R_{ij}-(R_s-\epsilon_{\uparrow}))}{\epsilon_{\uparrow}}\right) \cos^{\xi_{\uparrow}-2}\left(\frac{\pi(R_{ij}-(R_s-\epsilon_{\uparrow}))}{2\epsilon_{\uparrow}}\right) & \text{for} \quad R_s \leq R_{ij} < R_s + \epsilon_{\uparrow} \\ 0 & \text{for} \quad R_s + \epsilon_{\uparrow} > R_{ij} \end{cases} \tag{C.4}$$

The cutoff derivative formula here uses the double angle formula to reduce the exponent of cos.

While this cutoff does come at an increased computational expense compared to BP, it allows for significant reduction (or even complete elimination) of cross-talk between bases whose offsets are far apart. This may improve information capture and increase the benefit of using some of the advanced network architectures proposed below.

(Note, we also have several animations of this cutoff that exhibit part of the dynamic range it can span.)

C.2 "Gauss-Fourier" Spatial Bases

Since the functional forms of BP are so few, we sojourn ever so briefly into the vast and uncountably infinite space of mathematics applicable to our purpose.

C.2.1 "Gauss-Fourier" Radial (GFR) Basis

The first complete basis offered in this work is the combination of the conventional Gaussian radial basis function with the GF Cutoff.

$$B_{ij}^{R} = \sum_{j \neq i}^{N} e^{-\eta(R_{ij}-R_{s_r})^2} f_B\big(R_{ij}\big) \tag{C.5}$$

$$\frac{\partial B_{ij}^{R}}{\partial R_{ij}} = -\sum_{j \neq i}^{N} \Big[2\eta(R_{ij}-R_{s_r}) f_B\big(R_{ij}\big) + f_B^{'}\big(R_{ij}\big)\Big] e^{-\eta(R_{ij}-R_{s_r})^2} \tag{C.6}$$

Parameters

$$\text{Cutoff: } \epsilon_{\downarrow}, R_s, \epsilon_{\uparrow}, \xi_{\downarrow}, \xi_{\uparrow} \tag{C.7}$$

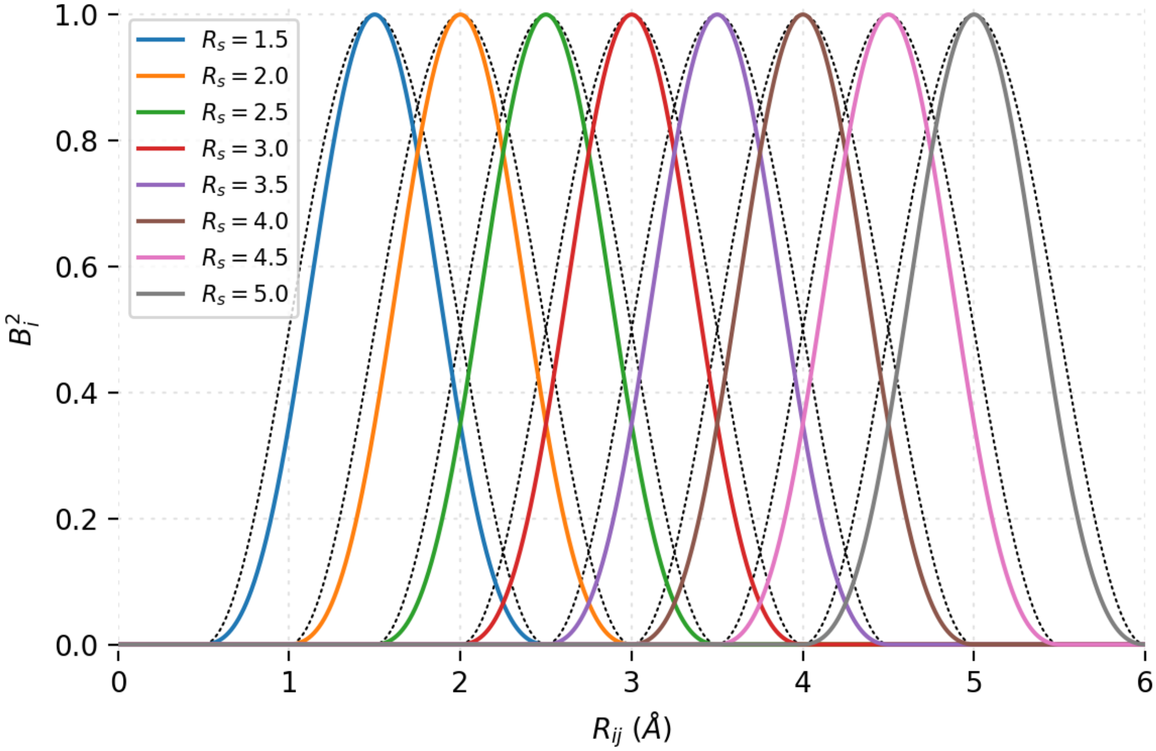
$$\text{Radial: } \eta, R_{s_r} \tag{C.8}$$

The radial basis offset value R_{s_r} may be the same as the cutoff offset value R_s, in which case it will be represented simply as R_s. This arrangement is generally in keeping with the design concept for the cutoff, and is practiced throughout this work—although we acknowledge that this relationship is arbitrary and other practice may be preferable in other contexts.

Fig. C.2 clearly shows the underlying Gaussian radial basis function's form is preserved throughout the domain. This is more dynamic and controllable than the conventional BP form, and can even eliminate cross-talk between bases (meaning that basis with large offsets can have zero value when basis with short offsets are completely activated, and vice-versa), thus disambiguating the radial information presented to the NN.

Fig. 2.16 shows normalized BP radial bases, highlighting the distortion caused by the BP cutoff function.

Figure C.2: "Gauss-Fourier" Radial Basis presented in this work. Cutoff in black.



C.2.2 Gauss-Fourier Bi-Radial Bases

The first new and novel three-body term in this work is the bi-radial basis:

$$B_{ijk}^{B} = \sum_{j\neq i}^{N} \sum_{k\neq i,j}^{N} e^{-\eta_{ij}(R_{ij}-R_{s_{ij}})^2} e^{-\eta_{ik}(R_{ik}-R_{s_{ik}})^2} f_B\big(R_{ij}\big) f_B\big(R_{ik}\big) \tag{C.9}$$

While this is a three body term and falls short of expressing angular relationships, it may nonetheless be useful as a unique descriptor of atomic environments and calculates faster than angular terms (particularly when calculating derivatives).

Regrettably, we have not experimented extensively (or in our own opinion sufficiently) with this basis, although we do present it as part of the overall picture and body of work of this thesis.

C.3 "Gauss-Fourier" Angular Bases

The design concept for this set of angular basis functions is to be tunable such that maximal activation of a particular basis corresponds to a particular geometric relation at equilibrium. For example, in any lattice structure there is always a nearest neighbor atom group, and angles between members of that group. There is always a second-nearest neighbor atom group, and angles between nearest neighbors and second-nearest neighbors as well as angles between different second-nearest neighbors.

These bases can be interpreted as "coordination detectors" or "molecular bond detectors."

C.3.1 Angular Basis 000

The first new angular basis of this work is comprised of five terms: one angular term, two radial terms, and two cutoffs. The radial terms are essentially as above, and need not have the same offset (R_s) or decay (η) parameters. The angular term supplies an additional attenuation based on a difference between an actual angle and a reference angle characteristic of a specific coordination number and structure.

$$B^A_{ijk} = \sum_{j\neq i}^{N} \sum_{k\neq i,j}^{N} e^{-\eta_\theta(\theta_{ijk}-\theta_s)^2} e^{-\eta_{ij}(R_{ij}-R_{s_{ij}})^2} e^{-\eta_{ik}(R_{ik}-R_{s_{ik}})^2} f_B\big(R_{ij}\big) f_B\big(R_{ik}\big) \tag{C.10}$$

As before when differentiating a product five terms, we differentiate each term separately, starting with the angular term.

$$\theta_{ijk} = \arccos\left(\frac{\vec{R}_{ij}\cdot\vec{R}_{ik}}{R_{ij}R_{ik}}\right) \quad \omega = \frac{\vec{R}_{ij}\cdot\vec{R}_{ik}}{R_{ij}R_{ik}} = \cos\theta_{ijk} \tag{C.11}$$

$$\frac{\partial\theta_{ijk}}{\partial z_i} = \frac{\partial\theta_{ijk}}{\partial\omega}\frac{\partial\omega}{\partial z_i} = \frac{-1}{\sqrt{1-\omega^2}}\frac{\partial\omega}{\partial z_i} \tag{C.12}$$

$$\frac{\partial\theta_{ijk}}{\partial z_j} = \frac{\partial\theta_{ijk}}{\partial\omega}\frac{\partial\omega}{\partial z_j} = \frac{-1}{\sqrt{1-\omega^2}}\frac{\partial\omega}{\partial z_j} \tag{C.13}$$

$$\frac{\partial\theta_{ijk}}{\partial z_k} = \frac{\partial\theta_{ijk}}{\partial\omega}\frac{\partial\omega}{\partial z_k} = \frac{-1}{\sqrt{1-\omega^2}}\frac{\partial\omega}{\partial z_k} \tag{C.14}$$

Right away we see trouble. The divisors to the partials of angle with respect its cosine have asymptotes when $\cos\theta_{ijk} = \pm 1$. That means that a line of atoms will break the derivatives of this basis.

$$\alpha = e^{-\eta_\theta(\theta_{ijk}-\theta_s)^2} \tag{C.15}$$

$$\frac{\partial\alpha}{\partial z_i} = \frac{\partial\alpha}{\partial\theta_{ijk}}\frac{\partial\theta_{ijk}}{\partial z_i} = -2\eta_\theta(\theta_{ijk}-\theta_s)\alpha\frac{\partial\theta_{ijk}}{\partial z_i} \tag{C.16}$$

$$\beta = e^{-\eta_{ij}(R_{ij}-R_{s_{ij}})^2} \tag{C.17}$$

$$\frac{\partial\beta}{\partial R_{ij}} = -2\eta_{ij}(R_{ij}-R_{s_{ij}})\beta \tag{C.18}$$

$$\gamma = e^{-\eta_{ik}(R_{ik}-R_{s_{ik}})^2} \tag{C.19}$$

$$\frac{\partial\gamma}{\partial R_{ik}} = -2\eta_{ik}(R_{ik}-R_{s_{ik}})\gamma \tag{C.20}$$

$$\psi = f_B\big(R_{ij}\big) \tag{C.21}$$

$$\frac{\partial\psi}{\partial R_{ij}} = \text{(see above)} \tag{C.22}$$

$$\chi = f_B\big(R_{ik}\big) \tag{C.23}$$

$$\frac{\partial\chi}{\partial R_{ik}} = \text{(see above)} \tag{C.24}$$

$$\frac{\partial B^A_{ijk}}{\partial x_i} = \sum_{j\neq i}^{N} \sum_{k\neq i,j}^{N} \left[\frac{\partial \alpha}{\partial x_i}\beta\gamma\psi\chi + \alpha\frac{\partial \beta}{\partial x_i}\gamma\psi\chi + \alpha\beta\frac{\partial \gamma}{\partial x_i}\psi\chi + \alpha\beta\gamma\frac{\partial \psi}{\partial x_i}\chi + \alpha\beta\gamma\psi\frac{\partial \chi}{\partial x_i} \right] \tag{C.25}$$

$$\frac{\partial B^A_{ijk}}{\partial x_j} = \sum_{j\neq i}^{N} \sum_{k\neq i,j}^{N} \left[\frac{\partial \alpha}{\partial x_j}\beta\gamma\psi\chi + \alpha\frac{\partial \beta}{\partial x_j}\gamma\psi\chi + \alpha\beta\gamma\frac{\partial \psi}{\partial x_j}\chi \right] \tag{C.26}$$

$$\frac{\partial B^A_{ijk}}{\partial x_k} = \sum_{j\neq i}^{N} \sum_{k\neq i,j}^{N} \left[\frac{\partial \alpha}{\partial x_k}\beta\gamma\psi\chi + \alpha\beta\frac{\partial \gamma}{\partial x_k}\psi\chi + \alpha\beta\gamma\psi\frac{\partial \chi}{\partial x_k} \right] \tag{C.27}$$

Parameters

$$\text{Radial } ij\text{: } \epsilon_\downarrow, R_s, \epsilon_\uparrow, \xi_\downarrow, \xi_\uparrow, \eta \tag{C.28}$$

$$\text{Radial } ik\text{: } \epsilon_\downarrow, R_s, \epsilon_\uparrow, \xi_\downarrow, \xi_\uparrow, \eta \tag{C.29}$$

$$\text{Angular: } \eta_\theta, \theta_s \tag{C.30}$$

This setup can be okay if one does not care about the analytic derivatives, because of the asymptotic behavior of `arccos` for perfect linear arrangements of atoms. While these do not exist in nature, they happen in MD and therefore this basis is not suitable for MD. The following was created to work around that problem.

C.3.2 Angular Basis 001

In order to have a smoothly differentiable basis suitable for implementation in MD that uses an angular offset, and no approximations in the way that PANNA does, we propose the following: [80] (PANNA explains in a reference close to the end that an approximation is made in MD because their "modified-BP" angular basis does not have an asymptote-free analytic derivative.)

$$B^A_i = \sum_{j\neq i}^{N} \sum_{k\neq i,j}^{N} e^{-\eta_\theta(1-\cos(\theta_{ijk}-\theta_s))} e^{-\eta_{ij}(R_{ij}-R_{s_{ij}})^2} e^{-\eta_{ik}(R_{ik}-R_{s_{ik}})^2} f_B\big(R_{ij}\big) f_B\big(R_{ik}\big) \tag{C.31}$$

$$\alpha = e^{-\eta_\theta(1-\cos(\theta_{ijk}-\theta_s))} = e^{-\eta_\theta(1-\cos\theta_{ijk}\cos\theta_s-\sin\theta_{ijk}\sin\theta_s)} \tag{C.32}$$

$$\frac{\partial\alpha}{\partial x_i} = -\eta_\theta\left(1-\frac{\partial\cos\theta_{ijk}}{\partial x_i}\cos\theta_s-\frac{\partial\sin\theta_{ijk}}{\partial x_i}\sin\theta_s\right)\alpha \tag{C.33}$$

Please see the Appendices for the partial derivatives of sin and cos of θ_{ijk}. The remaining derivatives are as before.

$$\frac{\partial B^A_{ijk}}{\partial x_i} = \sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_i}\beta\gamma\psi\chi+\alpha\frac{\partial\beta}{\partial x_i}\gamma\psi\chi+\alpha\beta\frac{\partial\gamma}{\partial x_i}\psi\chi+\alpha\beta\gamma\frac{\partial\psi}{\partial x_i}\chi+\alpha\beta\gamma\psi\frac{\partial\chi}{\partial x_i}\right] \tag{C.34}$$

$$\frac{\partial B^A_{ijk}}{\partial x_j} = \sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_j}\beta\gamma\psi\chi+\alpha\frac{\partial\beta}{\partial x_j}\gamma\psi\chi+\alpha\beta\gamma\frac{\partial\psi}{\partial x_j}\chi\right] \tag{C.35}$$

$$\frac{\partial B^A_{ijk}}{\partial x_k} = \sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_k}\beta\gamma\psi\chi+\alpha\beta\frac{\partial\gamma}{\partial x_k}\psi\chi+\alpha\beta\gamma\psi\frac{\partial\chi}{\partial x_k}\right] \tag{C.36}$$

C.3.3 Angular Basis 002

This basis is designed as before, except for the omission of the Gaussian terms. Thus, it and its derivatives are faster to compute.

$$B^A_i = \sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N} e^{-\eta(1-\cos(\theta-\theta_s))}f_B(R_{ij})f_B(R_{ik}) \tag{C.37}$$

$$\frac{\partial B^A_{ijk}}{\partial x_i} = \sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_i}\psi\chi+\alpha\frac{\partial\psi}{\partial x_i}\chi+\alpha\psi\frac{\partial\chi}{\partial x_i}\right] \tag{C.38}$$

$$\frac{\partial B^A_{ijk}}{\partial x_j} = \sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_j}\beta\gamma\psi\chi+\alpha\beta\gamma\frac{\partial\psi}{\partial x_j}\chi\right] \tag{C.39}$$

$$\frac{\partial B^A_{ijk}}{\partial x_k} = \sum_{j\neq i}^{N}\sum_{k\neq i,j}^{N}\left[\frac{\partial\alpha}{\partial x_k}\beta\gamma\psi\chi+\alpha\beta\gamma\psi\frac{\partial\chi}{\partial x_k}\right] \tag{C.40}$$

Regrettably, we have not conducted extensive testing of this basis, although we present it as part of this work.

Appendix D

Novel Spin-Agnostic Bases Results vs BP

D.1 Numerical Evaluation

When we consider the power of any computational method, we must always consider the practical reality of the "wall time" required before calculations are complete. This value is always dependent on manifold factors: motherboard, CPU, memory size, memory speed, GPU, SSD/HDD, OS, and programming environment (C/C++/python/PyTorch). We do not claim to be expert in optimizing the performance of this full-stack from bare metal to interpreter script, although we do expect our results to be reasonably comparable to common deployment environments throughout academia and industry.

In the case of ML IPs we break overall end user wall-time down into the:

1. Featurization time t_f
2. Energy calculation time t_e.

t_f is dependent on:

1. The cutoff radii, and by extension the number of atoms inside the cutoff radii of an atom of interest. This is the largest factor directly affecting featurization time.

2. The number of one body bases. It is critical to identify and include any one-body parameter that can significantly improve predictive efficacy because these nearly always have the least cost.
3. The number of two-body bases. These are limited, of course, in their ability to capture information because they cannot resolve detail beyond the radial, or details integrated over the radial dimension. Despite that, again, considering their relatively small computational cost it is of high import to deploy them to maximal effectiveness.
4. The number of three-body bases.
5. The number of higher-order bases.

In the simple case of a fully-connected, single-task bilayer NN, t_e is dependent on:

1. The number of inputs to the first hidden layer, i.
2. The number of neurons in the first hidden layer, h_1.
3. The activation function of the first hidden layer.
4. The number of neurons in the first second layer, h_1.
5. The activation function of the first second layer.

The activation function that computes most quickly is ReLU, which we have used extensively because of the exploratory nature of this work and the search space we are sampling. We therefore choose the following metrics of performance:

$$FOM_1 = \frac{1}{\text{RMSE} \times t_f} \tag{D.1}$$

$$FOM_2 = \frac{1}{\text{RMSE} \times t_f^2} \tag{D.2}$$

The reader will note that we have omitted t_e, making these metrics most valid for comparisons using equal or roughly equivalent NNs. Since featurization is accomplished en masse prior to training, these values are readily available quantities.

$t_f + t_e$ is available from our MC application, and used to define the following:

$$FOM_3 = \frac{1}{\text{RMSE} \times (t_f + t_e)} \tag{D.3}$$

$$FOM_4 = \frac{1}{\text{RMSE} \times (t_f + t_e)^2} \tag{D.4}$$

D.2 {GF Bases / FS Cutoff} vs BP

Title	hours	rad	ang	R_c max	RMSE
BP 2012 Artrith/Behler	1.022	8	43	6.000	26.300
BP 2016 Artrith/Behler	0.849	8	18	6.500	28.400
BP This Work 0	0.789	15	32	6.000	11.071
BP This Work 1	0.791	33	32	6.000	1.635
BP This Work 2	1.059	50	32	8.171	1.795
New Bases 1	0.108	30	4	7.198	2.075
New Bases 2	0.171	30	4	7.698	2.086

Title	RMSE	hours	FOM_1	FOM_2	$\frac{FOM_1}{FOM_{1ref}}$	$\frac{FOM_2}{FOM_{2ref}}$
BP 2012 Artrith/Behler	26.300	1.022	0.037	0.036	1.000	1.000
BP 2016 Artrith/Behler	28.400	0.849	0.041	0.049	1.114	1.341
BP This Work 0	11.071	0.789	0.114	0.145	3.077	3.984
BP This Work 1	1.635	0.791	0.774	0.979	20.788	26.865
BP This Work 2	1.795	1.059	0.526	0.497	14.140	13.647
New Bases 1	2.075	0.108	4.460	41.275	119.839	1133.083
New Bases 2	2.086	0.171	2.802	16.373	75.279	449.473

Appendix E

Multi-Input Multi-Output Neural Networks (MIMONN)

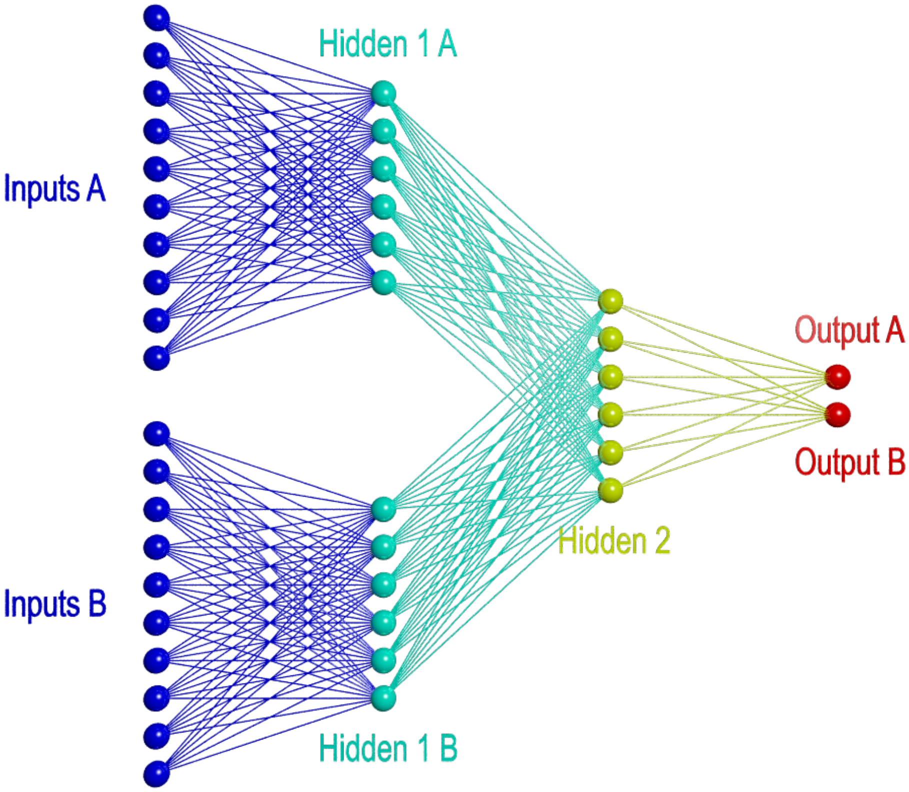
Fig. E.1 shows a concept presented in this work for a new type of network that uses multiple input stages, a common middle, and multiple outputs. This is a simple expression of the concept, and each component (input/midle/output) can be expanded into more complex structures.

The insight here is that common interfaces can be defined to allow for a significant amount of modularity. This occurs broadly in software, telecommunications, and other technologies. So long as every one of the "input stages" has the same number of connections to the "common middle," then each "input stage" can have any number of input features, layers, and structure/connectivity—so long as those input features still map to the same label.

This could enable a *single MIMONN network* to learn and express the relationships between geometry, spin, and energy *for many elements.* Each element would use its own "input stage" and can have a unique number of features and parameters optimized for it—and its own output.

The hope is that the "common middle" section learns what is universal or common to all of the elements of the system.

Figure E.1: Multi-Input Multi-Output Neural Network Architecture



The MIMONN could also be used in an iterative way, for example using one input and output to calculate part of an input used by another stage. For example, a feature vector for one element could be used to predict the partial charge of that element. Then the partial charge value could be used by another input stage to predict that atom's contribution to system potential energy. Something similar has been proposed by Behler recently, but again uses separate networks—herein it is proposed to do that with a unified network.

Appendix F

Parameter Choice Policy and Data Pre-/Post-Processing

F.1 Uniformly Distributed Parameter Selection

The early work on BP tended to use uniform distributions to set parameters. For example the decay parameter η tended to be uniform in log-space. The ξ parameter tended be 2 raised to an integer series.

While BP defined an R_s parameter (Gaussian RBF offset), they only used zero-valued offsets. In our very first familiarization with BP, we applied it to H_2O and used R_s values at or near the equilibrium OH bond distance and equilibrium intramolecular HH distance. When working in solid state systems we experimented with uniform, linearly distributed R_s.

F.2 Random Parameter Selection

In the literature, basis set size and basis parameter selection are generally considered arbitrary. Indeed, they are, and justifiably so if one wishes to use a general, application-non-speciifc, and human unbiased approach.

In the spirit of attempting to be unbiased, we chose to use random numbers from an arbitrary range for basis set parameters. The resulting set would be used to featurize the dataset (at the time) and then the correlation coefficients between the features would be calculated.

The objective was to minimize the sum of the absolute value of the correlation coefficient matrix. This rewards having the least "quantity" of linear correlation between features, and in theory contains the most information—since a feature that can be expressed as a linear combination of other features does not contain information.

While this appealed from a conceptual point of view, it can be cost prohibitive, depending on one's computational resources.

F.3 Physically Motivated Parameter Selection

The radial bases are communicating to the NN the integrals of the product of the radial basis and the radial distribution function (RDF) of the atoms surrounding an atom of interest. (Reactor physics makes frequent use of similar such recompositions, both spatially and in energy.) Therefore the radial basis set is essentially a basis decomposition of that local RDF. Bonner spheres accomplish the same for measuring neutron spectra.

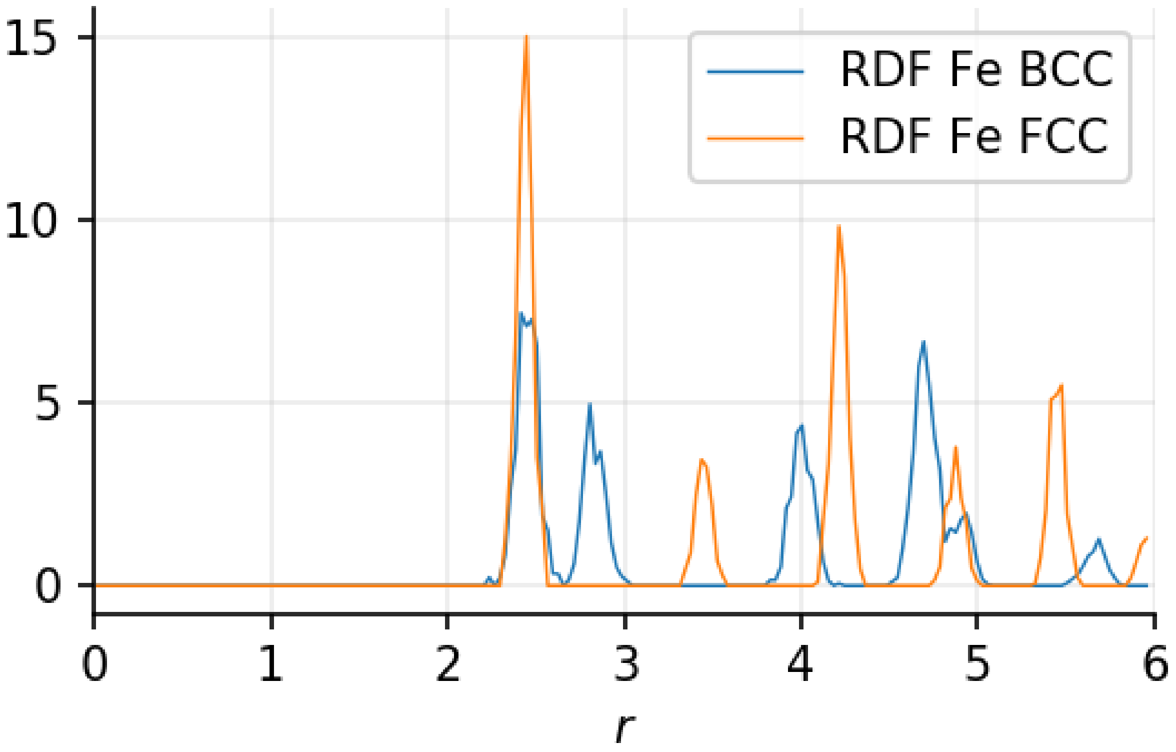
We explore a design concept for choosing basis set size and parameters as a basis set decomposition of the materials actual RDF. Given this, we pay particular attention to the peaks of the material's actual RDF—hoping to achieve improvement in both accuracy and computational efficiency.

To those who argue that this introduces bias and narrows application space, we agree. All potentials have bias in how the data samples were chosen and the parameters of their evaluation, and all potentials are tailored to particular domains of application. Therefore we do not see this an any different from common practice in the field of interatomic potential development. While our approach may add a few steps to the development process, these steps are straightforward and outlined below.

F.3.1 RDF, Fe

Since the early days of this work, Fe was of primary interest. Having generated some AIMD of Fe in both BCC and FCC configurations, we used Ovito to calculate the RDFs of each case, shown in Fig. F.1 below.

Figure F.1: RDF of BCC and FCC Fe



The modes of the RDF at equilibrium are given using some basic geometric relationships, shown in the Table F.1:

Mode #	BCC	FCC	BCC $\lambda = 2.83163$	FCC $\lambda = 3.44557$
1	$\frac{\sqrt{3}}{2}\lambda$	$\frac{\sqrt{2}}{2}\lambda$	2.45226	2.43639
2	λ	λ	2.83163	3.44557
3	$\sqrt{2}\lambda$	$\frac{\sqrt{6}}{2}\lambda$	4.00453	4.21994
4	$\frac{\sqrt{11}}{2}\lambda$	$\sqrt{2}\lambda$	4.69573	4.87277
5	$\sqrt{3}\lambda$	$\frac{\sqrt{10}}{2}\lambda$	4.90453	5.44792
6	2λ	$\sqrt{3}\lambda$	5.66326	5.96790

Table F.1: Nearest neighbor distances in perfect BCC & FCC lattices.

F.3.2 R_s at RDF Modes

Given that energy penalties are the result of perturbations from equilibrium, and Gaussian radial basis functions allow for fine tuning of their offsets, we asked why not set the offsets R_s equal to the modes of the RDF?

Figure F.2: BP R_s set to RDF modes.

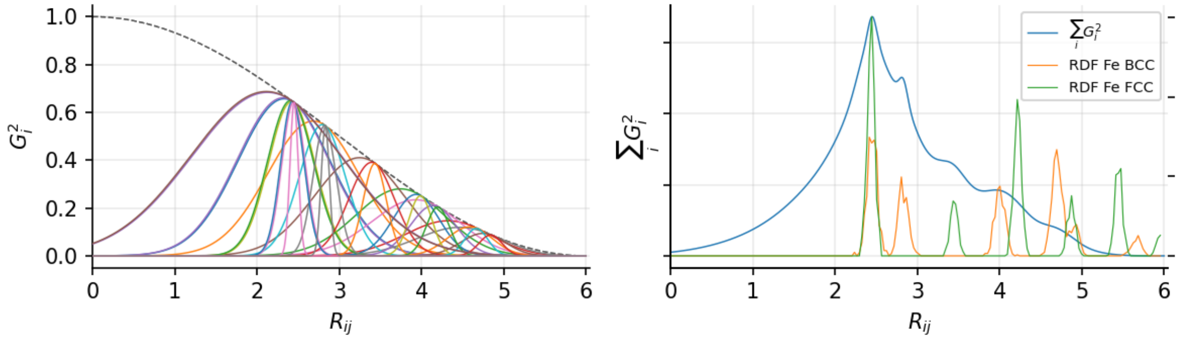
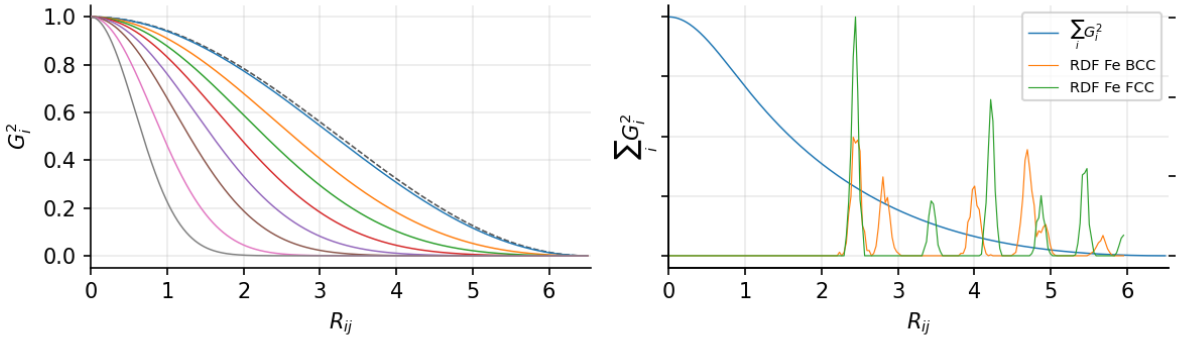


Fig. F.2 show one such result. In contrast, Fig. F.3 below shows the conventional approach published in older BP literature. [45, 70, 78] More recent BP literature does use $R_s \neq 0$. [44]

Figure F.3: BP radial bases centered on the origin.

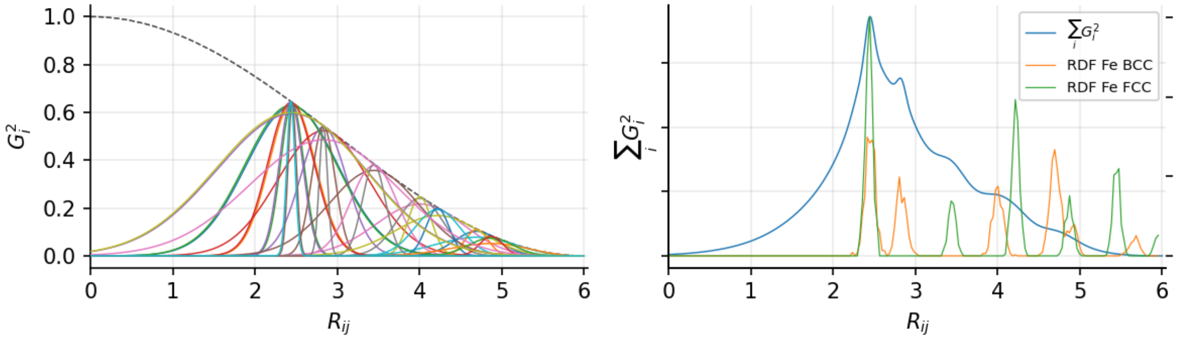


We postulated that the former arrangement would more easily and efficiently glean accurate energy penalty relations for perturbations from equilibrium than the latter.

F.3.3 Exact maxima at RDF Modes

Realizing that the maxima of BP radial bases are not R_s except when $R_s = 0$, we also tried an alternate formulation of the above. The exact position of the maximum is dependent on both R_s and η, so the overall radial basis set in Fig. F.4 looks very different from what is shown in Fig. F.2. Interestingly, we did not find this exact form performed better than when R_s was set to RDF modes.

Figure F.4: BP radial bases' actual maxima set to RDF modes.



F.4 Calibration of Parameters

Beyond using physically-motivated parameters as described above, we can also apply our physics / engineering common sense toward a methodology for setting reasonable starting points for optimization of other basis set parameters that materially affect performance.

Once we have an initial guess, we can search neighboring values in a linear or logarithmic fashion as appropriate.

F.4.1 Radial Terms

Since radial terms are so efficient to calculate, it is really worth every effort we can make *a priori* to improve their ability to capture information in ways that help the NN learn and express PESs.

F.4.1.1 η, Set by Absolute Displacement

Nearest neighbors in either the BCC or FCC equilibrium cases (zero-K, zero-pressure infinite medium lattice parameter) are less than 2.5 Å away. A huge displacement from equilibrium would be 1.0 Å. So, if we want the exponential part of the radial basis functions to attenuate to $\frac{1}{8}$ of the zero displacement value (i.e. when $R_{ij} = R_s$) when $R_{ij} - R_s = \pm 1.0$ Å then:

$$0.125 = e^{-\eta \cdot 1^2} \tag{F.1}$$

$$\ln 0.125 = -\eta \cdot 1^2 \tag{F.2}$$

$$\eta = -\ln 0.125 = -2.07944154167983 \sim 2 \tag{F.3}$$

The reader may ask why $\frac{1}{8}$? Why not $\frac{1}{4}$ or $\frac{1}{2}$? Frankly, this was simply our chosen starting point guided by the idea that most of the sensitivity that matters should happen before an atom gets anywhere close to being displaced by 1.0 Å from its equilibrium lattice site. It also leaves some room for movement past that, since extreme distortions and high-energy situations are within the domain of interest for some users.

F.4.1.2 η, Set by Relative Displacement

Another concept for η is to tie it to R_s for that particular basis function. This way a lattice distortion applied universally will actually permute arguments for different basis functions to the same magnitude. For example, 0.1 shear strain would displace an atom at any number of lattice distances by $0.1R_{ij}$, whereas the absolute distances differ by the ratio of lattice units $(0.1\lambda \neq 0.2\lambda)$. This thinking leads to a different calibration. Let's say we want to attenuate a 0.2 relative distance $\left(\frac{R_{ij}-R_s}{R_s} = 0.2\right)$ to $\frac{1}{4}$ of the equilibrium value:

$$0.25 = e^{-\eta \cdot 0.2^2} = e^{-\eta \cdot \left(\frac{R_{ij}-R_s}{R_s}\right)^2} \tag{F.4}$$

$$\ln 0.25 = -\eta \cdot 0.2^2 \tag{F.5}$$

$$\eta = -\frac{\ln 0.25}{0.2^2} = \frac{1.386}{0.2^2} \sim 34.657 \tag{F.6}$$

In practice, to achieve that η using Equation 2.18 it would be necessary to divide 34.657 by R_s^2.

F.4.1.3 $\epsilon_\downarrow$ and $\epsilon_\uparrow$, Absolute and Relative

$\epsilon_\downarrow$ and $\epsilon_\uparrow$ directly set the span of perception left and right of the offset. Therefore guidelines on absolute displacement sensitivity can be readily implemented. We do wish to draw to the attention of the reader the fact that $\xi_\downarrow$ and $\xi_\uparrow$ strongly affect the form of attenuation between the offset and the inner and outer cutoffs, so that setting $\epsilon_\downarrow$ and $\epsilon_\uparrow$ to be larger than the intended domain of interest can be advantageous.

If we wish to define parameters for Equation C.2 using relative displacements from the offset values, then we can cast ϵ as a fraction of R_s. This can be accomplished by scaling the argument for the cutoff:

$$\frac{\pi(R_{ij} - (R_s - \epsilon_\downarrow))}{2\epsilon_\downarrow} \rightarrow \frac{\pi(R_{ij} - (R_s - R_s\epsilon_\downarrow))}{2R_s\epsilon_\downarrow} \tag{F.7}$$

This was experimented with, however it has the effect of increasing the cutoff radius which is the fastest way to increase execution time and therefore must only be considered as a last resort.

F.4.2 Angular Terms

Angular basis term parameter calibration can explicitly enable coordination number detection. In the Table F.2, 1→1 denotes the relationship between most proximal nearest neighbors. This means the two nearest neighbors that are also most proximal to each other (or any set of most proximal atoms in a perfect lattice). 1→2 denotes the relationship between a nearest neighbor and a second nearest neighbor that is also in the group of atoms closest to the first-nearest neighbor.

		R_{ij}	Degrees	Radians		
1→1	$\frac{\sqrt{3}}{2}\lambda$	2.45226	70.53^o	1.230959	Fe	BCC
1→2	λ	2.83163	54.74^o	0.955317	Fe	BCC
1→1	$\frac{\sqrt{2}}{2}\lambda$	2.43639	60^o	1.047198	Fe	FCC
1→2	λ	3.44557	45^o	0.785398	Fe	FCC

Table F.2: Angles between nearest and second-nearest neighbors in BCC/FCC lattices.

F.4.2.1 $1 \rightarrow 1$ Angular Terms

Both $1 \rightarrow 1$ relationships are at roughly the same radial distance, so differentiating between BCC and FCC entails detecting the angle between those atoms. Therefore, we start calibration setting an attenuation target of 0.125 at $\Delta\theta = 0.18376$ (Radians):

$$0.125 = e^{-\eta_\theta \cdot 0.18376^2} \tag{F.8}$$

$$\ln 0.125 = -\eta_\theta \cdot 0.18376^2 \tag{F.9}$$

$$\eta_\theta = \frac{-\ln 0.125}{0.18376^2} = 61.57947 \tag{F.10}$$

The BCC and FCC 1 → 1 R_{ij} (closest neighbor distances) are nearly the same, so we cannot set an attenuation target using that. Instead, the the BCC 1 → 2 R_{ij} is the next best target. Therefore radial η's can be tuned with an attenuation target of 0.125 at $\Delta R = 0.37937$.

$$0.125 = e^{-\eta_{ij} \cdot 0.37937^2} \tag{F.11}$$

$$\ln 0.125 = -\eta_{ij} \cdot 0.37937^2 \tag{F.12}$$

$$\eta_{ij} = \frac{-\ln 0.125}{0.37937^2} = 14.4487 \tag{F.13}$$

Smaller values of η_{ij} increase the importance of η_θ, and vice-versa.

We might alternatively consider what magnitude of perturbation we wish to calibrate for and set the limits that way—forcing the NN to disambiguate what's what on its own (vs minimizing crosstalk). So long as the basis function decompositions of different geometrically perturbed systems are unique, then in principle any approach could work.

The BCC and FCC 1 → 1 basis are differentiated most by their angular offset.

F.4.2.2 BCC 1 → 2 Angular Term

For the BCC 1 → 2 case we don't need quite so sharp an attenuation because the radial distances are more different. Said differently, there aren't any atoms at 2.8 Å from an atom of interest in FCC Fe, so this basis applied to an FCC context should already be strongly attenuated by radial terms.. Therefore we choose an attenuation target of 0.25 at $\Delta\theta = 0.27564$ to reflect the differences in angle between the nearest source of radial interference (which is BCC 1 → 1):

$$0.25 = e^{-\eta_\theta \cdot 0.27564^2} \tag{F.14}$$

$$\ln 0.25 = -\eta_\theta \cdot 0.27564^2 \tag{F.15}$$

$$\eta_\theta = \frac{-\ln 0.25}{0.27564^2} = 18.24614 \tag{F.16}$$

Radial attenuation is set as for the 1 → 1 cases.

F.4.2.3 FCC $1 \rightarrow 2$ Angular Term

The FCC $1 \rightarrow 2$ angular term is similar to BCC $1 \rightarrow 2$, in that there are no other neighbor distances close (BCC or FCC) to the second-closest neighbor distance. Therefore we set the angular attenuation target to be 0.25 at $\Delta\theta = 0.169919$.

$$0.25 = e^{-\eta_\theta \cdot 0.169919^2} \tag{F.17}$$

$$\ln 0.25 = -\eta_\theta \cdot 0.169919^2 \tag{F.18}$$

$$\eta_\theta = \frac{-\ln 0.25}{0.169919^2} = 48.01441 \tag{F.19}$$

Radial attenuation is set as for the 1 → 1 cases.

F.4.2.4 Notes

Application of this method is not restricted to solid state. CHARMM uses sets of equilibrium values (which can be implemented in our methodology using offsets) and looks up parameters by the participating atoms' element and relationship (distance, angle, dihedral angle, etc.). For example, to detect hydrogen bonding in water a term could have an offset of 180^o and use the equilibrium distance to along the OH-bond and the distance between the H and neighboring O. Specific basis functions can be engineered as detectors for any molecular configuration: intramolecular OH bonds, intramolecular NH bonds, all known examples of hydrocarbon bonding, group specific bonds as in amides, etc.

This enormous number of possible bases for dealing with typical biological environments may be reasonably accomplished using a multi-input, common-middle, multi-output architecture as we describe in future work.

F.4.3 Pre- and Post-Processing of “Raw” Featurized Data (Single Element Systems)

While ML methods are renowned for being general and applicable to new scenarios with little to no changes in the workflow, that is not the complete story. ML methods, particularly randomly initialized and trained methods such as NNs, often benefit in terms of efficiency by changing the distribution of the data seen by the ML method. This applies to input features, and also to output labels in the case of regressors.

The difference between training on “raw” data and pre-processed data can easily be an order of magnitude in training time to achieve the same accuracy. This is the case for numerical reasons, for example if a bias is initialized to zero and the actual mean of an input or output is very large then a NN will take many iterations to approach the correct offset. Said in a different way, the average value (zero-eth order prediction) for a prediction has to be correct before any of the function’s shape (higher order terms in the prediction) can be fitted.

If is possible to use any technique for any feature or label, and to use one approach for a particular feature or label and a different approach for a other features or labels. The only rule is that whatever has been done to preprocess must be undone to convert NN outputs to the actual predictions that can be interpreted directly without further mathematical processing.

For our case, the pre-processing and post-processing of features are straightforward, but also significant due to their role in calculating forces. We also face some complexity specific to our problem in that individual networks calculate energy on a per atom basis, but the quantity we have from our data is the system energy. While this is just a slight evolution of basic pre-processing for a single element system, the multi-element picture becomes more complex.

In this Section, the non-italic “NN” subscript indicates the variable is for direct input to the neural network, while the non-subscripted E or G indicates the raw value from DFT or basis function calculations, respectively. M indicates the number of samples, and N_i indicates the quantity of atoms in a given sample of index i.

F.5 Standarization

Standardization is common practice in many ML applications, and generally means shifting and scaling a feature so that its distribution as seen by the ML predictive model has zero mean and unity variance.

While we started using standardization, we found that scaling by standard deviation left too many feature and energy values far above unity (indicative of high kurtosis)—which degraded performance in terms of both accuracy of the predictive model and training time.

F.5.1 Feature Standardization

Since features are calculated on a per atom basis, the mean and standard deviation can be calculated in a straightforward manner.

$$\mu_G = \frac{1}{\sum_{i=1}^{M} N_i} \sum_{i=1}^{M} \sum_{j=1}^{N_i} G_{i,j} \tag{F.20}$$

$$\sigma_G = \sqrt{\frac{1}{\left(\sum_{i=1}^{M} N_i\right) - 1} \sum_{i=1}^{M} \sum_{j=1}^{N_i} (G_{i,j} - \mu_G)^2} \tag{F.21}$$

Then, for any given feature:

$$G_{\mathrm{NN}} = \frac{G_i - \mu_G}{\sigma_G} \tag{F.22}$$

$$G_i = \sigma_G G_{\mathrm{NN}} + \mu_G \tag{F.23}$$

F.5.2 Energy Standardization

Since an energy value is for a collection of atoms in a system, which is allowed to vary in number of constituent atoms, we must obtain a per atom average energy and analog of standard deviation:

$$\mu_e = \frac{\sum_{i=1}^{m} E_i}{\sum_{i=1}^{m} n_i} \tag{F.24}$$

$$\sigma_e = \sqrt{\frac{\sum_{i=1}^{m} \left(\frac{E_i}{n_i} - \mu_E\right)^2}{m-1}} \tag{F.25}$$

$$E_{\text{NN}} = \frac{E - n\mu_e}{n\sigma_e} = \frac{E}{n\sigma_e} - \frac{\mu_e}{\sigma_e} \tag{F.26}$$

$$E_i = E_{\text{NN}} n\sigma_e + n\mu_e = n(E_{\text{NN}}\sigma_e + \mu_e) \tag{F.27}$$

F.6 Pre- and Post-Processing of Energy Labels for Multi-Element Systems

When there are multiple elements, calculation of average per-atom energies is still straightforward. Here is presented the method used in this work.

Given 4 elements where μ_i denotes the average per-atom energy for the i-th element and n_i denotes the number of atoms of that element, each data sample gives an equation:

$$\mu_1 n_1 + \mu_2 n_2 + \mu_3 n_3 + \mu_4 n_4 = E \tag{F.28}$$

All of the μ_i values are therefore easily found using an overdetermined least squares fit (assuming there are many more data samples than elements).

Herein is proposed two ways to calculate the per-atom scaling factor:

$$\sigma_1^2 n_1 + \sigma_2^2 n_2 + \sigma_3^2 n_3 + \sigma_4^2 n_4 = \left(E - (\mu_1 n_1 + \mu_2 n_2 + \mu_3 n_3 + \mu_4 n_4)\right)^2 \tag{F.29}$$

$$\sigma_1 n_1 + \sigma_2 n_2 + \sigma_3 n_3 + \sigma_4 n_4 = \sqrt{\left(E - (\mu_1 n_1 + \mu_2 n_2 + \mu_3 n_3 + \mu_4 n_4)\right)^2} \tag{F.30}$$

Neither of these “σ”’s are really a standard deviation, but they function as a proxy. Pre- and post-processing are completed using standardization or rescaling. For standardization:

$$E_{\mathrm{NN}} = \frac{E_i - (\mu_1 n_1 + \mu_2 n_2 + \mu_3 n_3 + \mu_4 n_4)}{\sigma_1 n_1 + \sigma_2 n_2 + \sigma_3 n_3 + \sigma_4 n_4} \tag{F.31}$$

$$E_i = E_{\mathrm{NN}}(\sigma_1 n_1 + \sigma_2 n_2 + \sigma_3 n_3 + \sigma_4 n_4) + \mu_1 n_1 + \mu_2 n_2 + \mu_3 n_3 + \mu_4 n_4 \tag{F.32}$$

For rescaling the per-atom scaling factor as defined in F.29 or F.30 are not necessary: