DESCRIPTION OF LOCAL CHEMICAL ENVIRONMENTS IN MACHINE LEARNING POTENTIALS USING SPHERICAL BESSEL DESCRIPTORS

by

S.Emir Köçer

B.S., Mechanical Engineering, Bogazici University, 2017

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Mechanical Engineering

Boğaziçi University

2020

ACKNOWLEDGEMENTS

I would like to express my sincere thanks to my supervisors Hakan Erturk and Jeremy Kyle Mason for their support and the precious knowledge they shared with me.

Hakan Hocam, you were a true role model to me and your guidance starting from the undergraduate years encouraged me to pursue a career in academia. I learned the importance of being resilient from you, and this will be a great asset I'll carry through my life.

Jeremy Hocam, it was a great honour for me to have the chance of working with you. Beyond the tremendous scientific knowledge you shared, your ethical perception on life and academy have taught me priceless lessons. Thank you for your patience and motivation, I'll always remember your contribution to my personality and career.

At the beginning of my graduate studies, I was not very familiar with computational nanoscience, and did not know how to dive into research. If I could obtain the MSc degree with three publications starting from that zero point, I owe this to the inspiration I took from Tolga Akıner. As an ambitious PhD senior and a true friend, he was the one teaching me lots of know-how and making me gain the confidence I need. Without his hands on my shoulder, I could not get to this point. Doctor, thank you for everything.

Dedicating your life to something and pushing hard means a fluctuating life. I am so grateful to Esin Özlav, my love and my best friend, for the unconditional compassion and support she showed me through those tough days.

Finally, I would like to thank my family and all of my friends from Bogazici University for the love and wonderful memories we've shared.

ABSTRACT

DESCRIPTION OF LOCAL CHEMICAL ENVIRONMENTS IN MACHINE LEARNING POTENTIALS USING SPHERICAL BESSEL DESCRIPTORS

Molecular dynamics is an effective numerical simulation technique for investigating material behavior at an atomistic scale, where simulated motions of atoms and molecules within a physical system are governed by Newtonian mechanics. A primary concern in these simulations is to accurately calculate the potential energy surface of a chemical environment as this hypersurface is then differentiated to calculate the atomic forces. These potential energy surfaces are defined either using pre-defined analytical expressions or carrying out first-principle calculations based on the electronic structure of the system. The former is computationally inexpensive and therefore suitable for reaching large length and time scales with a compromise on accuracy, whereas the latter is restricted only to small systems containing few hundred atoms at most due to the tremendous computational burden but offers high accuracy. Recently, machine learning potentials emerged as a third option to predict potential energy surfaces using nonlinear regression. These potentials are purely data driven and rely on reconstructing the map between chemical environments and corresponding potential energy surfaces, and promise high accuracy and efficiency at the same time. A major step towards developing a machine learning potential is to describe chemical environments in terms of some real-valued numbers, called 'descriptors'. The performance and accuracy of the final potential strongly depends on the performance and accuracy of the description. Despite this vital importance, however, no canonical set of descriptors has yet appeared in the literature as a solid base that could satisfy all the desired mathematical properties for a robust description. In this thesis, a novel set of descriptors, referred to as 'Spherical Bessel descriptors', is introduced that are symmetrically invariant, continuous, differentiable and optimally complete; this is a set of features that does not

appear to be satisfied completely by any alternative. A systematic approach for testing completeness behavior in descriptors is devised. The performance of the presented Spherical Bessel descriptors is further validated in molecular dynamics simulations using neural network potentials, and compared to other commonly used descriptors in the literature.

ÖZET

MAKİNE ÖĞRENİMİ POTANSİYELLERİNDE KÜRESEL BESSEL TANIMLAYICILARI KULLANARAK YEREL KİMYASAL ORTAMLARIN TANIMLANMASI

Atomların ve moleküllerin simüle edilen hareketlerinin Newton mekaniğine göre kontrol edildiği moleküler dinamik, malzeme davranışını atomik ölçüde incelemek için etkili bir numerik simülasyon tekniğidir. Bu simülasyonlarda öncelikli ilgilerden biri potansiyel enerji yüzeyini doğru bir şekilde hesaplamaktır, çünkü bu hiperyüzey sonradan atomik kuvvetleri hesaplamak için türevlenir. Bu potansiyel enerji yüzeyleri ya önceden tanımlanmış analitik ifadeler kullanarak ya da sistemin elektronik yapısı üzerinden ilk-prensip hesaplamaları yaparak tanımlanırlar. İlki bilgisayımsal maliyeti olarak ucuz olsa da ve bu sebeple doğruluk noktasında taviz verse de daha büyük sistemleri daha uzun süreli çalışabilmek için uygun iken, ikincisi fazla bilgisayımsal yükünden ötürü yalnızca bir kaç yüz atomlu sistemlere sınırlıdır ama yüksek doğruluk sağlar. Son dönemlerde, doğrusal olmayan regresyon kullanan makine öğrenim potansiyelleri potansiyel enerji yüzeylerini tahmin etmenin üçüncü yolu olarak ortaya çıktı. Tamamen bilgi ile işleyen bu potansiyeller kimyasal ortamlarla onlara tekabül eden potansiyel enerji yüzeyleri arasındaki eşlemi yeniden düzenlemeye bel bağlıyorlar ve aynı anda hem yüksek doğruluk hem de verim vaat ediyorlar. Bir makine öğrenim potansiyeli geliştirmeye doğru büyük bir adım kimyasal ortamların 'tanımlayıcılar' denen bir takım reel-değerli sayılar tarafından tanımlanmasıdır. Son potansiyelin doğruluğu ve performansı tanımlamanın doğruluğu ve performansına güçlü bir şekilde bağlıdır. Bu büyük öneme rağmen, lakin, standartlaşabilmiş ve arzu edilen tüm matematiksel özelliklere sahip olabilmiş bir tanımlayıcı henüz literatürde ortaya çıkmamıştır. Bu tezde başka herhangi bir alternatif tarafından tümüyle karşılanamamış özellikler listesini karşılayan; simetrik-değişmez, sürekli, türevlenebilir ve en ideal biçimde eksiksiz olan 'Küresel Bessel tanımlayıcıları' tanıtılıyor. Tanımlayıcılarda eksiksizlik davranışını

test eden bir yaklaşım tasarlanıyor. Sunulan Küresel Bessel tanımlayıcılarının performansı daha sonra moleküler dinamik simülasyonlarında sinir ağı potansiyelleri kullanarak tasdik ediliyor ve literatürde yaygın olarak kullanılan diğer tanımlayıcılarla kıyaslanıyor.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZET vi
LIST OF FIGURES x
LIST OF TABLES xiii
LIST OF SYMBOLS xv
LIST OF ACRONYMS/ABBREVIATIONS xviii
1. INTRODUCTION 1
1.1. Motivation 1
1.2. Objective 7
1.3. Organization and Contributions 8
2. GREEN-KUBO ASSESSMENTS OF THERMAL TRANSPORT IN NANOCOLLOIDS BASED ON INTERFACIAL EFFECTS 10
2.1. Introduction 10
2.2. Methodology 12
2.2.1. Green-Kubo Relations 12
2.2.2. Problem and Simulation Details 14
2.3. Results and Discussion 15
2.3.1. Green-Kubo Error Analysis 15
2.3.2. Thermal Conductivity Calculations 17
2.3.3. Interfacial Effects 19
3. A NOVEL APPROACH TO DESCRIBE CHEMICAL ENVIRONMENTS IN HIGH-DIMENSIONAL NEURAL NETWORK POTENTIALS 25
3.1. Introduction 25
3.2. Method 28
3.2.1. Descriptors 28
3.2.2. Neural Network Potential 32
3.2.3. Training Data 36
3.3. Results and Discussion 37

3.3.1. Behler–Parinello Descriptors 37
3.3.2. SOAP Descriptors 39
3.3.3. NNP Validations 42
4. CONTINUOUS AND OPTIMALLY COMPLETE DESCRIPTION OF CHEMICAL ENVIRONMENTS USING SPHERICAL BESSEL DESCRIPTORS . 45
4.1. Introduction 45
4.2. Spherical Bessel Descriptors 47
4.3. Completeness 52
4.4. Performance and Efficiency 58
5. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH 61
5.1. Conclusions 61
5.2. Recommendations For Future Research 63
REFERENCES 65
APPENDIX A: CHAPTER 3 86
A.1. Derivation of the Radial Basis Functions 86
A.2. Force Calculation 89
APPENDIX B: CHAPTER 4 92
B.1. Derivation of the Radial Basis Functions 92
B.2. Derivatives 94
B.2.1. Spherical Bessel Descriptors 94
B.2.2. SOAP Descriptors 95

LIST OF FIGURES

Figure 2.1. Cross-section of water-copper model with 5% volume fraction, 2.8 nm side length and 1.3 nm particle diameter. 14

Figure 2.2. Surface of HACF behavior for correlation time (τ) and for total simulation time (t) of a pure SPC/E water model. 16

Figure 2.3. Thermal conductivity of a pure SPC/E water model for a total simulation time of 8 ns. Inset figures are representing the thermal conductivities at 200 ps and 5 ns, respectively. They are produced by integration of individual ACFs within a correlation time interval τ. 17

Figure 2.4. Thermal conductivity results of 10 different initial velocity seeding for pure SPC/E water. 18

Figure 2.5. Two water blocks have been created in between three copper blocks to apply NEMD. 23

Figure 3.1. The values (a) and the first derivatives (b) of the radial basis functions $g_n(r)$ for $0 \leq n \leq 4$ and $r_c = 1$. The behavior of the functions close to $r = r_c$ indicates that the second derivatives vanish there as well. 32

Figure 3.2. Feed-forward neural network scheme used in this study. E^i is the atomic potential energy of the ith atom, G^i are the descriptors of the local environment, $\bar{r}^{ij}$ are the relative position vectors of the neighbors, N is the number of neighbors and N_d is the number of descriptors. 34

Figure 3.3. Performance of the BP descriptors and the proposed descriptors with increasing temperature. The left and right y-axis show the RMSE in meV and the average number of neighbors n, respectively. 38

Figure 3.4. Average values of the proposed descriptors and the BP descriptors for a single training data set consisting of 10^4 silicon configurations at 300 K. 39

Figure 3.5. Performance of the SOAP descriptors and the proposed descriptors with increasing temperature, with NN architectures of (18-10-1) and (16-10-1), respectively. σ_a is the standard deviation of the Gaussians used to generate the neighbor density function in angstroms. All other fitting parameters for the SOAP descriptors were taken from the literature [148]. 41

Figure 4.1. Contour plots on the yz plane of the basis functions used to construct (a) the previous SB descriptors [76] for $n = 0$, $l = 1$ and $m = 0$ ($g_{00}Y_1^0$), (b) the SOAP descriptors [148] for $n = 0$, $l = 1$ and $m = 0$, (c) the current SB descriptors for $n = 1$, $l = 1$ and $m = 0$ ($g_{01}Y_1^0$), and (d) the Zernike descriptors [172] for $n = 1$, $l = 1$ and $m = 0$. There is a visible discontinuity at the origin in (a) and (b). 50

Figure 4.2. The logarithmic singular values $\log(s_i)$ of $J^{[q]}$ for the p_{nl} descriptors as a function of q. The decay of the s_i to the machine precision for $i > 15$ indicates that the rank of J is 15. 55

Figure 4.3. The logarithmic singular values $\log(s_i)$ of $J^{[18]}$ for the p_{nl} descriptors as a function of q. Five configurations were generated by scaling the initial configuration such that the radial coordinate of the most distant atom ranged from $0.967r_c$ to r_c. 55

Figure 4.4. The logarithmic singular values $\log(s_i)$ of $J^{[q]}$ for the SOAP descriptors as a function of q indexed by $0 \leq n_1 \leq n_{\max}$, $0 \leq n_2 \leq n_{\max}$ and $0 \leq l \leq l_{\max}$ with the ordering described in the text. 56

Figure 4.5. The logarithmic singular values $\log(s_i)$ of $J^{[q]}$ for the SOAP descriptors as a function of q indexed by $0 \leq n_1 \leq n_{\max}$, $0 \leq n_2 \leq n_{\max}$ and $0 \leq l \leq l_{\max}$ with the additional constraint that $n_1 \geq n_2$. . . . 57

Figure 4.6. Comparison of NNP performance for (n-10-1) architectures, where n is the number of descriptors, for the four specified sets of descriptors. RMSE values are presented as a function of n for 1500 test points after $20,000$ training cycles. The values of n do not coincide because of differing indexing schemes. 58

Figure 4.7. RMSE values are presented as a function of training cycles e for NNPs with a fixed (25-10-1) architecture and 8500 training points. While all four converged by $20,000$ training cycles, the SOAP descriptors required the most cycles. 59

LIST OF TABLES

Table 2.1. Green-Kubo estimations of the thermal conductivity of water-Cu nanosuspensions with a single nanoparticle. $\bar{k}$ is the mean and $\sigma_{\bar{k}}$ is the standard error of the mean for different water models (Model), volume fractions (VF), number of particles (N_p), and particle diameter (D_p). Simulations 1-3 are pure water. Standard error of the means are calculated based on the long-time errors as mentioned above. σ_r in the last column is the ratio of $\sigma_{\bar{k}}$ to $\overline{k}$. 20

Table 2.2. Comparison of the Green-Kubo thermal conductivity estimations with the theoretical Maxwell limit and experimental result for water-Cu nanosuspension at the same volume fraction [124]. N_p, D_p and VF are the number of particles, particle diameter and volume fraction, respectively. k_{eff} is the thermal conductivity ratio of the nanosuspension, which is the ratio of nanofluid thermal conductivity to base fluid thermal conductivity. 21

Table 2.3. The thermal conductivities ($\overline{k}$) of water-Cu nanosuspensions with a single nanoparticle for different water models. ϵ is the interfacial energy parameter between oxygen and copper. D_p is the nanoparticle diameter. 22

Table 2.4. Thermal resistance R at water-Cu interface for different water potentials. σ_R is the associated error in the measurement. 24

Table 3.1. The minimum RMSE per atom for different temperatures T and NN architectures, where n is the average number of neighbors and N_d is the number of descriptors. All of the neural networks were trained on 8500 training points for 20000 epochs, and RMSE values were obtained on 1500 test configurations that are not included in the training set. 42

Table 3.2. Bulk modulus (K), shear modulus (S) and Poisson's ratio (PR) of solid-state silicon at 300 K as measured in MD simulations using the analytic SW potential and our NNP. 44

LIST OF SYMBOLS

b	Bias of a neural network layer
D_p	Nanoparticle diameter
E	Atomic potential energy
e	Atomic total energy
$\bar{F}$	Atomic force vector
f_c	Cutoff function
$\bar{G}$	Descriptor vector
g_n	Radial basis function with orders n and $l = 0$
g_{nl}	Radial basis function with orders n and l
h_α	Enthalpy of species α
J	Jacobian matrix
j_l	lth spherical Bessel function
K	Bulk modulus
k	Thermal conductivity
$\bar{k}$	Mean thermal conductivity
k_B	Stefan-Boltzmann constant
k_{eff}	Effective thermal conductivity
m	Atomic mass
n_{max}	Truncation parameter for radial expansion
N_L	Number of layers in a neural network
N_p	Number of nanoparticles
N_α	Number of atoms of species α
N_T	Number of training points
l_{max}	Truncation parameter for angular expansion
P_l	Legendre polynomial with order l
PR	Poisson's ratio
R	Thermal resistance
$\bar{r}$	Atomic position vector

r_c	Cutoff radius
r_s	A hyperparameter in Behler-Parinello formulation
r_Δ	A hyperparameter in SOAP formulation
$\bar{Q}$	Heat current vector
q''	Heat flux
S	Shear modulus
s	Singular value
T	Temperature
t	Simulation time
u_{ln}	nth root of lth spherical Bessel function
w	Weight of a neuron
V	Volume
VF	Volume fraction
$\bar{v}$	Atomic velocity vector
Y_l^m	Spherical harmonic with order l and degree m

α	Atomic species index
β	Sampling truncation parameter
ΔT	Temperature difference
ϵ	Interatomic energy parameter
η	A hyperparameter in Behler-Parinello formulation
Γ	Root mean square error
γ	Triplet angle
κ	SOAP kernel
λ	A hyperparameter in Behler-Parinello formulation
ν	Number of neighbors of an atom
ω	Atomic weight for multispecies chemical environments
ϕ	Azimuthal angle
Φ	Interatomic potential function
ρ	Neighbor density function around a central atom
σ	Interatomic distance parameter

σ_a	Atomic spreading in SOAP formulation
$\sigma_{\bar{k}}$	Standard error of the mean of thermal conductivity
σ_r	The ratio of $\sigma_{\bar{k}}$ to $\bar{k}$
σ_w	SOAP kernel hyperparameter
τ	Correlation time
θ	Elevation angle
ξ	SOAP kernel hyperparameter
ϖ	Sampling time interval
ζ	A hyperparameter in Behler-Parinello formulation

LIST OF ACRONYMS/ABBREVIATIONS

ACF	Autocorrelation function
AI	Artificial intelligence
ANN	Artificial Neural Network
BO	Born-Oppenheimer
BP	Behler-Parinello
CM	Coulomb matrix
DFT	Density Functional Theory
EMD	Equilibrium Molecular Dynamics
FFFN	Feed-forward Neural Network
fs	Femtosecond
GAP	Gaussian Approximation Potential
GPR	Gaussian process regression
GK	Green-Kubo
HACF	Heat autocorrelation function
KRR	Kernel Ridge Regression
LAMMPS	Large-scale Atomic/Molecular Parallel Simulator
LB	Lorentz-Berthelot
LJ	Lennard-Jones
MD	Molecular dynamics
ML	Machine learning
MLP	Machine Learning Potential
MTP	Moment Tensor Potential
nm	Nanometer
ns	Nanosecond
NEMD	Nonequilibrium Molecular Dynamics
NN	Neural Network
NNP	Neural Network Potential
NPT	Isobaric-isothermal ensemble

NVE	Microcanonical ensemble
PES	Potential energy surface
PPPM	Particle-particle-particle mesh
RMSE	Root mean square error
SB	Spherical Bessel
SOAP	Smooth Overlap of Atomic Positions
SW	Stillinger-Weber
QM	Quantum mechanical
QSAR	Quantitative Structure-Activity Relationship
VMD	Visual Molecular Dynamics

1. INTRODUCTION

1.1. Motivation

The invention of the first computers in the late 1950s ushered in a new era of computational materials science research. Although experiments are the final arbiter of truth with regard to physical phenomena, restrictions on the capabilities of experimental devices together with the cost and reproducibility of experiments limit studies of material behavior at the molecular scale. Following the emergence of novel computational solutions to a variety of problems in engineering and natural sciences, materials science research also experienced a major breakthrough with the introduction of 'molecular dynamics (MD)' which models molecular interactions in materials by computer simulations based on Newton's laws of motion [1–5]. In MD, one can generate an artificial representation of an atomic system and simulate it through time at a desired thermodynamic state. These simulations, regarded as numerical experiments, can reveal significant information about various material properties such as thermal and electrical conductivity and shear viscosity, and other physical phenomena such as chemical reactions and phase changes. When compared with physical experiments, MD has the advantage of being able to track and investigate even atomic interactions in a material. One shortcoming is the computational demand, which increases with the increasing number of molecules in a simulated system and restricts the time and length scales of the simulation. In parallel to the exponential growth in the capacity of computational resources, the use of MD has increased in different research fields such as chemistry [6, 7], biology [8–10] and materials science [11, 12].

The physics in MD simulations is precisely that of Newtonian mechanics, where at each timestep Newton's second law of motion is solved for each molecule in the system to obtain their respective accelerations. These are then integrated to calculate velocities and positions. This approach neglects internal structure of atoms and treats them as though they were point particles - relying on the so-called Born-Oppenheimer (BO) approximation [13] which states that the motion of nuclei and electrons can

be treated separately in an atomic environment. Solving Newton's second law requires knowledge of a force field that defines the forces exerted on molecules within the system. In MD simulations, this force field is calculated by differentiating the potential energy surface (PES) - a multidimensional hypersurface that defines the energy state of an atomic environment as a function of atomic positions. The accuracy and computational efficiency of MD strongly depends on how the PES is defined and calculated during the simulation as it directly affects the calculation of the atomic forces that govern the system dynamics. Therefore, the calculation of the PES in MD is the most important step both for the accuracy and efficiency of simulations.

There are two main approaches to calculating the PES that dominate the current MD literature and subdivide the technique into *classical MD* and *ab initio MD*. In the former, the PES is a pre-defined function of atomic coordinates that is usually referred to as *potential function*. These *empirical* potential functions often include two-body and three-body terms that model the interactions between atomic pairs and triplets, respectively. Any parameters are also pre-defined for each type of interaction within the system, and are usually optimized based on experimental findings or quantum mechanical (QM) simulations. A major advantage of classical MD is its extensibility to larger domains and longer simulation times due to significant computational saving from fixing the functional form of the PES beforehand. However, this comes with the price of restricting the flexibility of the potential function as a consequence of optimizing and fixing the parameter set separately for each distinct phase of the system. As a result of this restriction, classical MD is usually not suitable for investigating complex physical phenomena involving interfacial dynamics, multi-species systems, phase transitions and reaction synthesis. Some of the frequently used empirical potential functions are the Lennard-Jones (LJ) 6-12 [14], Tersoff [15], Stillinger-Weber [16], ReaxFF [17] and embedded-atom method [18].

In ab initio MD [19], the PES is recalculated on-the-fly at each timestep by means of a QM method. Quantum mechanical methods are based on electronic structure calculations involving the solution of Schrödinger's wave equation [20], which is a nonlinear probabilistic model describing the distribution of electrons in an atomic

system. As opposed to neglecting electrons and treating molecules with restricted internal degrees of freedom, QM calculations focus solely on electrons in the system. Since the Schrödinger wave equation does not neglect QM effects and relies on more fundamental physics, the calculation of the PES is more accurate in ab initio MD than in the classical approach. However, the calculations are not as straight-forward as in classical MD and demand much more computational power. This tremendous computational demand usually constrains the length and time scales in ab initio simulations even with current computational resources, and limits the application of the technique to smaller systems containing no more than several hundred atoms, many fewer than is typically required to model complex physical phenomena. One of the most widely used ab initio theories is the 'Density Functional Theory' (DFT) that integrates the time-independent Scrödinger wave equation using electron density functionals [21–26]. The technique has been integrated into a variety of computational materials science studies in the last three decades [27–33], and appears to be the most accurate way for calculating the PES in the literature.

The question of how to combine ab initio accuracy with the speed of classical MD simulations has attracted a great deal of interest in the last couple of decades. One recent attempt involves employing machine learning (ML) [34–36] algorithms to develop data-driven predictive potentials to calculate the PES in MD simulations. After proving a great success in applications like voice recognition [37], computer vision [38] and virtual reality [39], ML resurged lately as a powerful sub-branch of artificial intelligence (AI) [40] and started to be utilized in scientific research studies as well. What made ML algorithms attractive specifically in computational chemistry and materials science was their capability to extract nonlinear functional relationships and patterns from high-dimensional data sets. This advanced ML as an alternative way to calculate a PES, where the energy of an atomic environment is almost always a complex nonlinear function of the local atomic coordinates. Usually referred to as 'machine learning potentials' (MLPs), this approach to predicting a PES by means of ML algorithms has lately been viewed as a third way to model molecular interactions in MD and is employed frequently [41–47].

MLP construction consists of three main stages: initial data generation, data transformation, and conditioning a selected ML algorithm on data. First, a reference data set needs to be generated that contains a variety of atomic configurations and the respective atomic energies (and perhaps forces as well). These atomic energies and forces are preferably calculated by means of ab initio methods such as DFT. This makes preparation of the reference set often the most computationally demanding part of MLP development. However, compared to ab initio MD, there is no need for any additional QM calculations on-the-fly once the potential is developed. The atomic configurations represented in the data set define the validity region of the final potential, as ML algorithms are notoriously unreliable when they extrapolate. A general-purpose MLP that can be applied in a variety of thermodynamic conditions should have data points in the reference set that span the configurational space effectively.

Once the raw reference data is collected, the next step is to transform it into a suitable format. Contrary to classical and ab initio potentials which are constructed to mimic physical laws, MLPs are purely mathematical concepts that are unaware of any physics relating to atomic systems. Even though a lack of bias can provide high transferability and flexibility in some cases, a major shortcoming of a solely mathematical nature is that MLPs are not inherently aware of the three physical symmetries (translational, rotational and permutational) of the configurational space. This can potentially lead to inaccuracies since atomic configurations are mostly provided in Cartesian coordinates and do not inherently satisfy the three physical symmetries either. Specifically, a MLP could map two chemically-identical configurations to different potential energies or force fields, resulting in unphysical behavior. One way to overcome this issue is by extending the reference set with many different symmetrically-equivalent versions of the existing data points. From a computational point of view, however, this is a very impractical process that might significantly decrease the efficiency of the MLP. Another approach is to transform the initial Cartesian atomic coordinates into a more symmetric form before feeding them into the ML algorithm. This transformation generates a set of real-valued symmetrically invariant numbers, called *structural descriptors* or *descriptors* in the literature.

After collecting and transforming the data, the third and final step is to select a ML algorithm and condition it on the data set. Since the aim is to predict atomic energies or forces, the ML algorithm acts as a nonlinear regressor in the context of MLPs. The regression process, commonly called *training*, involves the algorithm learning the topography of the PES by adjusting a parameter set via multidimensional optimization. In practice, the reference set is split into two parts called the *training set* and the *test set* before the training. The prediction accuracy of the trained potential is tested at the end on the test configurations, which are not included in the training set, to give an unbiased performance evaluation.

Among these three, the development of descriptors has lately been one of the main focuses of the MLP community, and no one of the pioneering descriptors can yet be regarded as canonical. The first examples of descriptors in the literature mostly appeared in applications for quantitative structure-activity relationships (QSAR) [48] and structure identification of molecules [49], where the description encoded molecular fingerprints of small systems containing only a few molecules. These descriptors are also called *molecular descriptors*, and are commonly employed in computational chemistry and bioinformatics studies [50–53]. The description of a PES requires a high-dimensional perspective though, since a typical MD simulation contains hundreds of atoms that effectively contribute to the dimension of the potential energy hypersurface of the chemical system. In an earlier low-dimensional attempt for fitting a PES using MLPs, Hobday et al. [54] used chemical metadata such as bond lengths, torsional angles and second neighbor information as inputs for a MLP rather than a sophisticated description scheme. They used neural networks (NNs) as regressors and could obtain successful fits for a limited number of systems with poor computational efficiency.

In 2007, Behler and Parinello [42] published a milestone study on descriptors that introduced high-dimensional MLPs. The first significant contribution in the paper was the represention of the total energy of the system as a sum over all individual atomic energies within the system, and modeling the atomic energies using identical NN architectures for a given species. This discrete atom-centered approach decreased the

computational demand significantly and enabled the development of high-dimensional MLPs. Second, they proposed a new set of symmetrically-invariant descriptors, referred to as the Behler-Parinello (BP) symmetry functions, to describe a chemical environment around a central atom. Their approach was shown to have comparable accuracy to DFT for silicon configurations with significantly less computational demand. Since then, the BP symmetry functions and their modified versions have been frequently used in other MLP studies [44, 55–61].

Bartók et al. [43] later proposed a descriptor vector for a given local chemical environment using the bispectrum of the neighbor density. They integrated their bispectrum descriptors into Gaussian process regression (GPR) algorithms [62] and developed the so-called 'Gaussian Approximation Potentials' (GAPs) to predict atomic energies and forces, and reached comparable accuracy to DFT. Three years later, Bartók et al. published a comprehensive study [63] on the description of chemical environments, where they reviewed various techniques and provided some means to evaluate the performance of a given descriptor set. Additionally, they introduced a new set of descriptors, called the 'Smooth Overlap of Atomic Positions' (SOAP) descriptors, which have been frequently used in MLP studies [47, 64–67].

Rupp et al. [68] introduced a novel way of generating descriptors, using Coulomb matrices (CMs) that consisted of pairwise Coulomb repulsion operators between nuclei of atoms. The CMs were then fed into a kernel ridge regression (KRR) algorithm [69] to predict the PES [70, 71]. In a recent study, Shapeev [72] devised 'Moment Tensor Potentials' (MTPs) using simple linear regression where invariant polynomials of the Cartesian coordinates are used as descriptors.

The above mentioned approaches share the common aim of describing chemical environments via a transformation from Cartesian space into a descriptor or feature space, effectively transforming atomic coordinates into symmetrically invariant forms. The accuracy and efficiency of the description, and consequently of the MLP, strongly depend on how this transformation is performed. An ideal description would satisfy several mathematical properties. First, descriptors should be invariant with respect to

translations and rotations of the system and permutations of labels for atoms of a given species. The usual way to satisfy translational invariance is to define the atomic coordinates relative to a central atom. This atom-centered approach introduced by Behler and Parinello [42] relies on the assumption that the energy of a central atom depends only on the relative positions of its neighbors within a cutoff radius. Constraining chemical environments by a cutoff radius is similar to the empirical potentials in classical MD, and is justified by the weakening interactions with increasing interatomic distance. Second, descriptors should be continuous and have continuous first and second derivatives throughout the entire domain. This is required as forces and elastic constants in MD simulations are defined by the first and second derivatives of the energies, respectively. Any discontinuity in the first or second derivatives could lead to erroneous force calculations in simulations. Third, a description should be *complete* in the sense that any given atomic environment can be reconstructed up to symmetry given the corresponding descriptors. Mathematically speaking, completeness of descriptors is satisfied if the atomic configuration space can be smoothly embedded into the descriptor space. In addition to these three central properties, other valuable properties of a descriptor set include efficient numerical evaluation and having as few adjustable parameters as possible. The former is required as the descriptors need to be evaluated at each time step for each atom in a MD simulation. If the calculation of descriptors is computationally expensive, the biggest advantage of the technique over ab initio MD could disappear. The number of parameters is a measure of complexity of the descriptors, and since they are largely system-dependent they need to be readjusted and optimized for each studied system. Therefore, having a formulation with fewer parameters improves transferability. Finally, each of the descriptors should carry unique information. To the best of the author's knowledge, no set of descriptors already in the literature satisfies all of these properties.

1.2. Objective

Even though interest in descriptors has lately increased dramatically, there is not a robust description of chemical environments that satisfies all the mathematical

requirements enumerated above and is widely accepted by the community. Motivated by this insufficiency, this thesis mainly aims to systematically develop a novel way to describe chemical environments in atomic systems. The new descriptors, called the Spherical Bessel (SB) descriptors, are similarly constructed to the bispectrum descriptors introduced by Bartok et al. [43] by expanding neighbor densities of atoms over radial and angular basis functions. Spherical harmonics [73] are used for the angular part similar to Bartok et al. [43], whereas for the radial part a linear combination of spherical Bessel functions [74] has been designed that inherently satisfies orthogonality and has vanishing first and second derivatives at the cutoff boundary. The SB descriptors are symmetrically invariant, orthogonal and twice differentiable, and have been shown to be continuous and complete. Moreover, they satisfy the necessary condition for optimal completeness, a notion that implies a given set of descriptors could be complete with the minimum possible number of descriptors, and has not been discussed before in the literature.

After being introduced, the performance and validity of the SB descriptors is tested in MD simulations by integrating them into neural network potentials (NNPs). They are then compared to other commonly used descriptors in the literature in terms of accuracy and computational efficiency.

1.3. Organization and Contributions

This thesis is structured as a compilation of three published manuscripts [75–77] that are presented in a chronological order in the three body chapters after the introduction.

Prior to MLPs and descriptors, the author investigated nanoscale heat transfer using equilibrium molecular dynamics (EMD) in classical MD simulations. The findings of this study shed light on the insufficiencies associated with empirical potentials, and therefore they will be presented in the second chapter to emphasize the author's motivation to study MLPs. Thermal conductivities of various water-copper nanofluid systems with differing basefluid potentials and nanoparticle features are predicted and

examined based on interfacial effects. The obtained results indicate that the ad-hoc parametrization of interfacial potentials based on standard mixing rules seems to be the main cause for the anomalously high thermal conductivities observed in some nanofluid studies and that classical MD is an insufficient choice for modeling complex phenomena such as interfacial dynamics.

The third and fourth chapters comprise studies on descriptors and MLPs. A new set of invariant and orthogonal descriptors for an atomic environment is introduced in the third chapter that improves upon Bartok's bispectrum approach [43], and uses a novel radial basis using spherical Bessel functions. The performance of the proposed descriptors are then compared to that of the BP descriptors [42] and SOAP descriptors [63], which are the two most frequent descriptors in the MLP literature. Finally, they are implemented in a NNP for solid-state silicon, and tested in MD simulations. Neural networks using the proposed descriptors are found to outperform ones using the BP and SOAP descriptors.

The subsequent chapter considers the continuity and completeness properties of the descriptors. First, a discontinuity occurring in the basis functions of the SOAP descriptors and the previously proposed descriptor is identified. An updated version of the latter, called for the first time the Spherical Bessel (SB) descriptors, is then provided that resolves this discontinuity. Following this, the notions of completeness and optimal completeness for descriptors are described, and evidence is provided that the updated version of the SB descriptors satisfies completeness and a necessary condition for optimal completeness. The standard construction of the SOAP descriptors is shown to not satisfy the condition for optimal completeness, and moreover it is found to be an order of magnitude slower to compute than the SB descriptors.

The fifth and final chapter contains the recommendations for future work and conclusions.

2. GREEN-KUBO ASSESSMENTS OF THERMAL TRANSPORT IN NANOCOLLOIDS BASED ON INTERFACIAL EFFECTS

2.1. Introduction

Colloidal suspensions have recently been the focus of numerous scientific studies due to their potential engineering applications [78]. Studies of particulate systems can involve rheology, interface science and colloidal chemistry [79], and could lead to advances in foams, gels, paints, coatings and wetting-dispersing agents [80]. One recently-developed colloidal system is known as a *nanofluid*, which is a suspension of nanoparticles designed for enhancing heat transfer [81]. The physics of thermal transport in these systems requires further study, and simulations are frequently used for this purpose because of experimental limitations at the relevant length scale.

Molecular dynamics (MD) is a powerful approach to investigate nanoscale dynamics where interatomic forces govern the system behaviour. There are two different schemes that can be used to measure the heat transport properties in MD simulations: non-equilibrium molecular dynamics (NEMD) [82] and equilibrium molecular dynamics (EMD) [83]. A heat flux is imposed on the system in NEMD and the thermal conductivity is calculated based on the resulting temperature gradients. However, finite size effects are severe as a result of very high temperature gradients, and careful consideration is required to obtain reliable temperature profiles [84]. Equilibrium molecular dynamics instead calculates thermal conductivity from the time decay of heat flux fluctuations based on the fluctuation–dissipation theorem. While this requires more computational power, the method does not suffer from the drawbacks of NEMD and is widely used in the literature.

The thermal conductivities of Lennard–Jones liquids [85], water [86–88], methane hydrate [89], Ar–Cu nanofluids [90–92], and a water–platinum nanofluid [93] have been

estimated with EMD, though most of these studies do not include a detailed error analysis. This is significant because the fluctuation–dissipation theorem is based on the chaotic movements of particles or atoms, which introduces statistical noise into the calculation of the autocorrelation function (ACF). Porter and Yip [94] and Chen et al. [95] proposed to truncate the integration time of the heat current autocorrelation function (HACF) to minimize the statistical noise. Other approaches include curve fitting [96], block averaging [97], random walk modelling [98], and spectral analysis of time series [99]. These studies provide fundamental insights about the statistical nature of the EMD method, and should be carefully considered to estimate the errors associated with Green–Kubo (GK) calculations.

In most nanofluid studies, Maxwell's relation [100] is used as a reference for the effective thermal conductivity of a simulated system. The Maxwell model neglects Brownian motion, nanolayering and agglomeration [101] as possible heat transfer enhancement mechanisms for nanofluids, and provides a lower limit by only considering the thermal conductivity of the mediums. Anomalous thermal enhancement values with respect to the Maxwell limit have been reported in some nanofluid studies that use the GK method, where the contribution of various mechanisms to this enhancement has been investigated by comparing simulations and theoretical calculations [92, 93, 102–104]. However, inconsistent results and recent developments [95, 96, 99] concerning GK calculations in the literature indicate that there is no generally accepted GK algorithm, and that the effect of the GK parameters on the thermal transport has not yet been fully understood.

The anomalous thermal conductivity of nanofluids as found by GK calculations has been hypothesized to be the effect of Brownian motion [92] or nanolayering [103]. Recent studies have found an insignificant contribution of Brownian motion of the nanoparticles to the heat transfer enhancement of nanocolloids [105, 106]. As for the nanolayering effect, both experimental and theoretical investigations have tried to quantify the thermal conductivity of the adsorbed liquid layer [107–109]. However, investigating nanolayering with classical MD simulations can be difficult since the interface potential is expected to have a considerable effect on the thermal transport at

the solid-liquid surface. Most EMD studies use the Lorentz-Berthelot (LB) rules at the interface for lack of an alternative without adequately studying the effect of the surface parameters, even though the LB rules have no physical justification. It has been recently shown that LB can lead to an overestimation of the interfacial thermal resistance for a hBN-water system [110], and an optimization process carried out by fitting potential parameters is required to accurately represent the interface.

This study presents a comprehensive evalution of the Green-Kubo approach by considering the thermal transport in a nanocolloid system, where several inconsistencies and anomalous thermal conductivity results have been reported in the literature. Using a standard force-field as the interatomic potential and a common modeling scheme for interface interactions, possible sources of the observed anomalities are revealed. A statistical assessment of several sources of error is presented and the effect of different water potentials on the thermal transport calculations is tested. The effect of the interface parameters on the thermal conductivity calculations is investigated. Finally, thermal resistances at the solid-liquid interfaces are quantified to further investigate interfacial effects. We believe that the presented results clarify the contributions of surface phenomena and the associated interfacial thermal resistance to the anomalous thermal enhancement found with Green–Kubo calculations.

2.2. Methodology

2.2.1. Green-Kubo Relations

The Green-Kubo method relies on the fluctuation-dissipation theorem and linear-response theory [83], and describes the system behavior using time autocorrelation functions (ACF). Integrating an ACF in time allows transport properties in the equilibrium state to be extracted [111]. The thermal conductivity is given in the GK formalism as:

$$k = \frac{1}{3k_B V T^2} \int_0^\infty \left\langle \bar{Q}(0) \cdot \bar{Q}(\tau) \right\rangle d\tau \tag{2.1}$$

where k_B is Boltzmann's constant, T is the absolute temperature, V is the system volume, $\bar{Q}$ is the heat current vector, and the integrand is the heat autocorrelation function (HACF). The HCAF is an ensemble average, which relates the thermal conductivity to the heat current in an equilibrated system. Equation 2.1 involves integrals in the continuous case, but since time is discrete in MD simulations, the integral is replaced by a summation in practice. The heat current vector is defined as:

$$\bar{Q} = \frac{1}{V}\left\{\sum_i e^i \bar{v}^{\,i} + \frac{1}{2}\sum_{i<j}\left[\bar{F}^{\,ij}\cdot(\bar{v}^{\,i}+\bar{v}^{\,j})\right]\bar{r}^{\,ij} - \sum_\alpha h_\alpha \sum_i \bar{v}^{\,i}\right\} \tag{2.2}$$

where $\bar{v}^{\,i}$ is the velocity of atom i, $\bar{r}^{\,ij}$ and $\bar{F}^{\,ij}$ are distance and force vectors between atoms i and j, α is the atomic species, and h_α is the enthalpy of that species. The total energy e^i is the sum of the kinetic and potential energies which can be expressed as:

$$e^i = \frac{1}{2}m^i(\bar{v}^{\,i})^2 + \frac{1}{2}\sum_j \Phi(\bar{r}^{\,ij}) \tag{2.3}$$

where $\Phi(\bar{r}^{\,ij})$ is the interatomic potential energy. The enthalpy exclusion in the last term of Equation 2.3 for multi-phase systems was introduced by Babaei et al. [112] who studied a methane-Cu colloidal system using EMD, and was not common in earlier GK studies [92, 93, 104]. This term is subtracted from the heat flux because it represents the energy carried by the mediums but not transported between them. The species enthalpy is defined as:

$$h_\alpha = \frac{1}{N_\alpha}\sum_{i=1}^{N_\alpha}\left[e^i + \frac{1}{3}\left(m^i(\bar{v}^{\,i})^2 + \frac{1}{2}\sum_{j=1}^{N_\alpha}\bar{r}^{\,ij}\cdot\bar{F}^{\,ij}\right)\right] \tag{2.4}$$

where N_α is the number of atoms of species α.

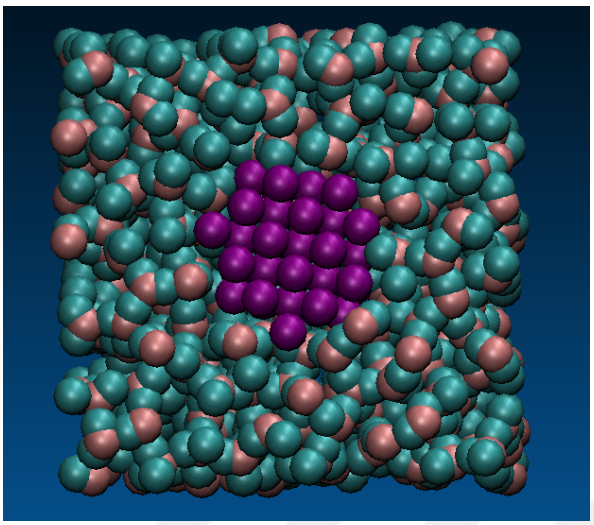

Figure 2.1. Cross-section of water-copper model with 5% volume fraction, 2.8 nm side length and 1.3 nm particle diameter.

2.2.2. Problem and Simulation Details

Pure water and water-copper models with a single copper nanoparticle (diameter 1.3 or 1.8 nm) in the middle of cubic simulation domains with a side length of 2.8 nm and 4 nm were generated to investigate thermal transport as shown in Fig. 2.1. Volume fraction calculations are not trivial at the nanoscale, considering that the clearence at the solid-liquid interface is not well-defined. We calculated the volume fraction to be 5% using mass fractions and the bulk densities of the phases, and 4.5% using the sum of volumes of the Voronoi polyhedra. This was kept fixed for all nanofluid systems, and is reported as 5% throughout the text. TIP3P (flexible) [113], TIP4P/2005 (rigid) [114] and SPC/E (rigid) [115] potentials were used to define the water interactions. Pure water systems of the same dimensions were used to estimate the thermal conductivity of the base fluid. The SHAKE algorithm [116] was used to enforce the bond and angle constraints in the rigid water models, and long-range Coulombic interactions were solved using particle-particle-particle-mesh (PPPM) summations [117]. Atomic interactions within copper nanoparticles were defined using a LJ 6-12 potential with ϵ_{Cu-Cu} =9.4353 kcal/mole and σ_{Cu-Cu} =0.2338 nm [118]. The LB rule was used for interactions between oxygen atoms and copper atoms, and the timestep was set to 1 fs. Periodic boundary conditions were applied in all directions.

Each model was first equilibrated for 100 ps in the isobaric-isothermal (NPT) ensemble, followed by a further 2 ns in the microcanonical ensemble (NVE) at 300 K. The GK calculations were performed in 5-8 ns production runs (depending on the simulation) with HACFs calculated in 20 ps intervals, and the reported results are the average of 20 independent simulations with different initial velocity seedings. All simulations are carried out using the Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) [119], and Figure 2.1 was generated using Visual Molecular Dynamics (VMD) [120].

2.3. Results and Discussion

2.3.1. Green-Kubo Error Analysis

The confidence interval usually reported in the literature is for a single ACF, and corresponds to the fluctuations in the thermal conductivity within a single MD simulation. We denote this type of error as 'short-time error' in this study. Another source of error is denoted as 'long-time error', and can be observed in the scatter of calculated thermal conductivities for different velocity seedings. The heat flux fluctuations are governed by the local properties of the PES, and a single MD simulation explores a relatively small part of this surface over a period of ns. This effectively introduces a random error that depends more on the initialization, or where the system begins on the PES, than on the fluctuations of a single ACF. While in principle a single very long MD simulation would explore a sufficiently large region of configuration space, in practice the computational requirements are reduced by changing the initial velocity distribution of the atoms to obtain different trajectories and molecule orientations. This procedure is also followed in several other studies that average the thermal conductivities of independent simulations [102,121,122], yet is not the standard procedure in the literature.

The short-time errors occur because of the statistical noise in an ACF, and can be minimized by following the decay of the ACF. The time evaluation of the normalized HACF for pure water with the SPC/E potential (2.8 nm simulation cell in one

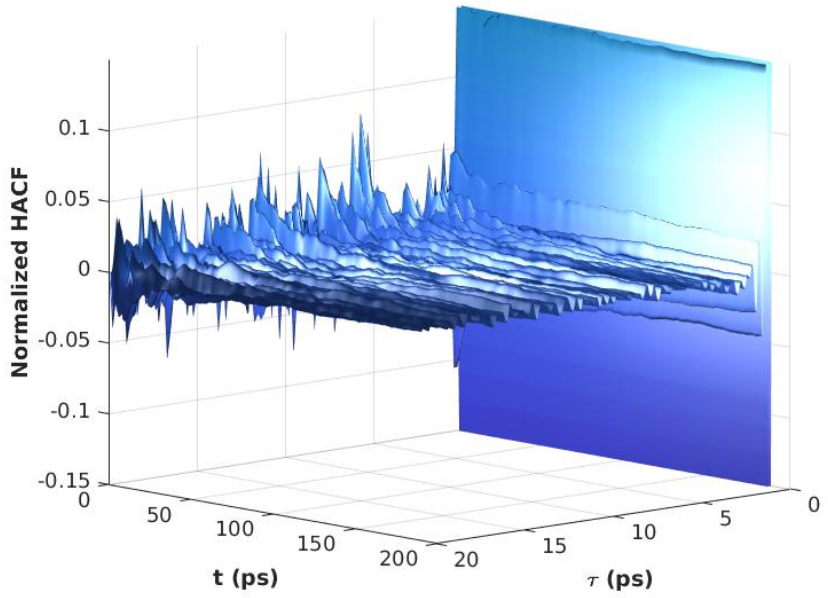

Figure 2.2. Surface of HACF behavior for correlation time (τ) and for total simulation time (t) of a pure SPC/E water model.

dimension) is presented in Figure 2.2 as a function of the correlation time τ, which represents the time interval within which the ACFs are calculated and integrated, and of the simulation time t, which is the point in the simulation at which the integration begins. A single HACF converges to zero over a 20 ps interval [88], and smoother overall HACF behavior is achieved by beginning the calculation after a short interval of simulation time (200 ps). The effect of the HACF decay can be also observed in Fig. 2.3 where the estimated thermal conductivity for a total simulation time of 8 ns is presented. Each data point is obtained by integrating the HACF over a 20 ps interval. The convergence of the integration can be observed by comparing the two inset figures at 200 ps and 5 ns, and the short-time error is reported as the standard error of the mean for the converged region of Fig. 2.3 between 5-8 ns. The value of this quantity is 0.002 W/mK, which is insignificant compared to the long-time error discussed below. The total production run is therefore set to 5 ns to save computation time.

The long-time error is calculated by sampling some number of relatively short GK results with different velocity initializations instead of performing one long simula-

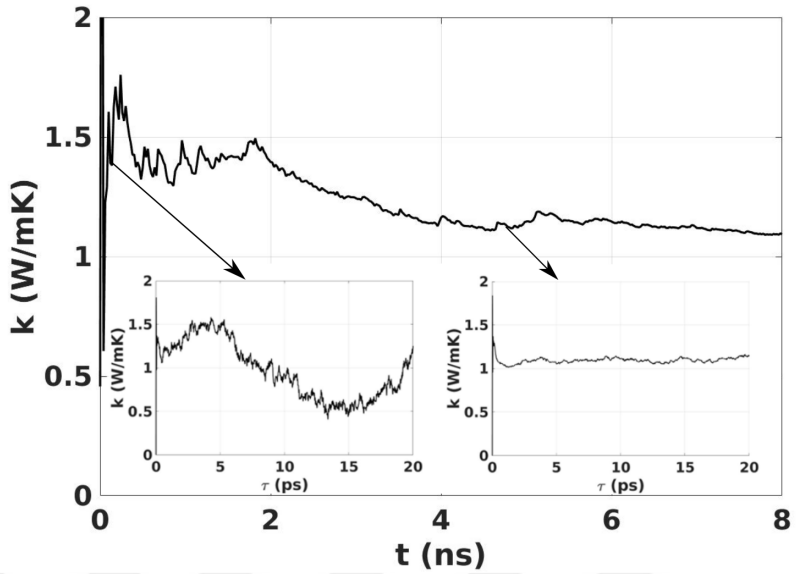

Figure 2.3. Thermal conductivity of a pure SPC/E water model for a total simulation time of 8 ns. Inset figures are representing the thermal conductivities at 200 ps and 5 ns, respectively. They are produced by integration of individual ACFs within a correlation time interval τ.

tion which would require more computation time. Green-Kubo results of the thermal conductivity of independent simulations with different initial velocity seedings for the SPC/E water model (2.8 nm simulation cell in one dimension) are shown in Fig. 2.4, where the thermal conductivity is seen to change significantly with different initializations. The standard error of the mean $\sigma_{\overline{k}}$ for the long-time error is 0.03 W/mK, significantly larger than the standard error of the mean for short-time error given above (0.002 W/mK). Therefore, the confidence interval of the thermal conductivity results reported below is constructed based on the long-time error.

2.3.2. Thermal Conductivity Calculations

The thermal conductivities of pure water and water-copper nanocolloid systems are calculated following the procedure described in Section 2.2.1. The results for different water potentials, particle diameters and number of particles are presented in Table 2.1. The thermal conductivities of pure water (Simulations 1-3) are in agreement with

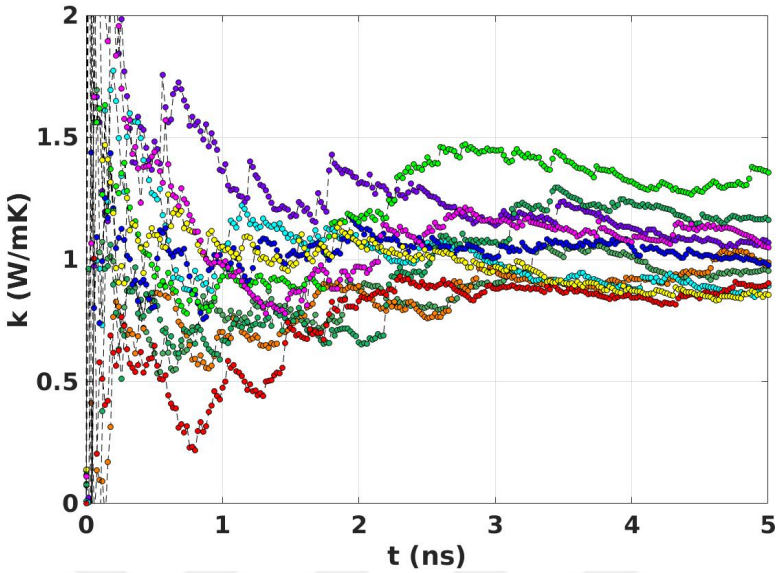

Figure 2.4. Thermal conductivity results of 10 different initial velocity seeding for pure SPC/E water.

the reported results in the literature [123]. The thermal conductivity enhancements (thermal conductivity of the nanofluid divided by that of pure water) of a water-copper nanofluid with a rigid water model (Simulations 5 and 6) are in agreement with the findings of Muraleedheran et al. [102] who used the rigid SPC/E water model. It is significant that the same trend is not observed with the flexible TIP3P model (Simulation 4).

The anomalous thermal enhancement observed with rigid water models exceeds both theoretical and experimental results, and has been attributed to an artificial particle correlation effect arising from the use of a single particle with periodic boundary conditions [102]. The suggested solution is to use multiple nanoparticles to break the symmetry of the system. This was tested for the rigid SPC/E water model using a three-nanoparticle system with the same volume fraction and particle diameter. Contrary to the hypothesis that artificial particle correlations are responsible for the anomalous enhancement, a further increase in the thermal conductivity was observed for the rigid SPC/E water model (Simulation 8). Moreover, an insignificant difference was found between the one- and three-nanoparticles cases with the flexible TIP3P

model (Simulations 4 and 7), suggesting that there may be some other effects responsible for the observed anomalous increase. The difference between water models persists and is even exacerbated for larger system sizes at a fixed volume fraction, with the larger single particle system having a slightly higher thermal conductivity for the TIP3P model (Simulations 4 and 9) and more than double the thermal conductivity for the SPC/E model (Simulations 5 and 10). The abnormal thermal conductivity values associated with SPC/E models tend to increase with increasing particle surface area within the system as additional particles are introduced or particle size is increased (Simulations 8 and 10). Given the reasonable thermal conductivities of the pure water models, these findings suggest that the interfacial interactions that obey the LB rules and that are different for each system are the cause of observed anomalous thermal conductivities.

This argument is supported by the results presented in Table 2.2, where the estimated thermal conductivities for the three different water models are compared with the Maxwell limit [100] and experimental results of Xuan and Li [124] at the same volume fraction. Considering the larger particle diameter in the experiment, our effective thermal conductivity result with the TIP3P model is quite reasonable. However, switching the water potential to SPC/E or TIP4P results in anomalous thermal conductivity ratios exceeding both experiment and theory, implying that the parametrization of the system dynamics is responsible for the overestimation rather than any physical mechanism. When the results in Tables 2.1 and 2.2 are considered together, our conclusion is that the surface interactions of solid–liquid interfaces in nanocolloids need to be accurately represented to avoid any unphysical interfacial thermal transport.

2.3.3. Interfacial Effects

The effect of surface interaction parameters on the effective thermal conductivity of solid–liquid systems has been investigated before in several NEMD studies [110,125], but the effect of these parameters on GK calculations has not been studied to our knowledge. An earlier NEMD study [110] showed up to a 12% change in thermal conductivity using an interface coupling tuned to satisfy the experimentally observed

Table 2.1. Green-Kubo estimations of the thermal conductivity of water-Cu nanosuspensions with a single nanoparticle. $\bar{k}$ is the mean and $\sigma_{\bar{k}}$ is the standard error of the mean for different water models (Model), volume fractions (VF), number of particles (N_p), and particle diameter (D_p). Simulations 1-3 are pure water. Standard error of the means are calculated based on the long-time errors as mentioned above. σ_r in the last column is the ratio of $\sigma_{\bar{k}}$ to $\overline{k}$.

Simulation	Model	$VF(\%)$	N_p	$D_p(nm)$	$\overline{k}(W/mK)$	$\sigma_{\bar{k}}$	σ_r
1	TIP3P	-	-	-	0.86	0.03	0.03
2	SPC/E	-	-	-	1.02	0.03	0.03
3	TIP4P/2005	-	-	-	1.05	0.03	0.03
4	TIP3P	5	1	1.3	1.12	0.04	0.04
5	SPC/E	5	1	1.3	3.73	0.12	0.03
6	TIP4P/2005	5	1	1.3	3.29	0.11	0.03
7	TIP3P	5	3	1.3	1.19	0.04	0.03
8	SPC/E	5	3	1.3	5.31	0.2	0.04
9	TIP3P	5	1	1.8	1.31	0.04	0.03
10	SPC/E	5	1	1.8	7.78	0.22	0.03

Table 2.2. Comparison of the Green-Kubo thermal conductivity estimations with the theoretical Maxwell limit and experimental result for water-Cu nanosuspension at the same volume fraction [124]. N_p, D_p and VF are the number of particles, particle diameter and volume fraction, respectively. k_{eff} is the thermal conductivity ratio of the nanosuspension, which is the ratio of nanofluid thermal conductivity to base fluid thermal conductivity.

	VF	N_p	$D_p(nm)$	k_{eff}
Maxwell [100]	5	-	-	1.15
Xuan and Li [124]	5	1	100	1.55
This study (TIP3P)	5	1	1.3	1.3
This study (SPC/E)	5	1	1.3	3.66
This study (TIP4P/2005)	5	1	1.3	3.13

contact angle as compared to one using LB rules. Surface interactions could have an even larger effect for EMD calculations since GK considers the energy flow associated with the motion of each atom at each time step by means of Eq. 2.2, whereas NEMD uses an average temperature calculated by means of kinetic theory.

A Lennard-Jones 6-12 function is employed for the interfacial interactions:

$$\Phi(r^{ij}) = 4\epsilon\left[\left(\frac{\sigma}{r^{ij}}\right)^{12} - \left(\frac{\sigma}{r^{ij}}\right)^{6}\right] \tag{2.5}$$

where ϵ is an energy scale and σ is a characteristic length. Ideally, ϵ and σ should be derived from experimental findings or first principle calculations; however, in most nanofluid studies they are calculated based on LB mixing rules. Our intention is to identify whether this could be the source of the anomalous thermal conductivity rather than some other simulation input. Existing investigations consider the effects of varying ϵ [125, 125–127] and the same procedure is followed here. The effect on the thermal conductivity of varying ϵ at the water-Cu interface is reported in Table 2.3, and

Table 2.3. The thermal conductivities ($\overline{k}$) of water-Cu nanosuspensions with a single nanoparticle for different water models. ϵ is the interfacial energy parameter between oxygen and copper. D_p is the nanoparticle diameter.

Simulation	Model	$\epsilon(kcal/mole)$	$D_p(nm)$	$\overline{k}(W/mK)$
1	SPC/E	1.21	1.3	3.73
2	SPC/E	1.21	1.8	7.78
3	SPC/E	0.98	1.3	0.91
4	SPC/E	0.98	1.8	0.99
5	TIP3P	0.98	1.3	1.11
6	TIP3P	0.98	1.8	1.31

indicates that the interfacial energy parameter would easily be able to account for the overestimation of the thermal conductivity in Table 2.1. Specifically, changing ϵ from the 1.21 kcal/mole given by the LB rule for the SPC/E potential to the 0.98 kcal/mole for the TIP3P potential (Simulations 1 and 3) significantly decreased the thermal conductivity. Although these values of ϵ do not have any particular pyhsical basis, they do establish that the thermal conductivity is very sensitive to surface ϵ, and that using the values predicted by the LB rule is likely a significant source of error. This reinforces the necessity of calibrating the interface potential based on experimental findings before using the Green-Kubo method to quantify thermal transport in nanocolloids.

Another observation is that the anomalous thermal conductivity values observed with both rigid models in Tables 2.1 and 2.2 were not observed with the flexible TIP3P model. While this could appear as a possible reason for the abnormally high thermal conductivities, this conclusion is not supported by the reasonable thermal conductivities of the base fluids in Table 2.1. Moreover, Table 2.3 provides direct evidence that an inaccurate interfacial potential could be responsible for the anomalous thermal conductivities.

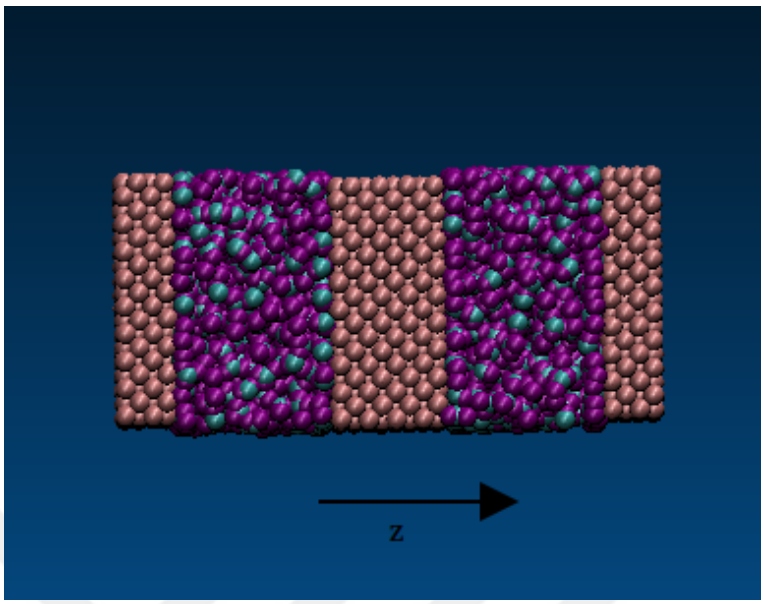

Figure 2.5. Two water blocks have been created in between three copper blocks to apply NEMD.

The significance of the interface to the observed dependence of thermal transport on the interatomic potential could also be expressed in terms of interfacial thermal resistance. Based on prior work [128], this is measured using the NEMD method. Three different water-copper systems were generated with the geometry shown in Figure 2.5 and the Müller-Plathe algorithm [82] was applied to create a heat flux in the z-direction. The temperature was calculated using a methodology presented elsewhere [128]. The interfacial thermal resistance was calculated as $R = \Delta T/q''$ where R is the thermal resistance, ΔT is the temperature difference at the water-copper interface, and q'' is the heat flux. Each system was equilibrated for 100 ps in the NPT ensemble, followed by a 1 ns production run in the NVE ensemble. The thermal resistance results were obtained by averaging the results of 100 independent runs, and are presented in Table 2.4. A higher thermal resistance was found for the TIP3P model compared to the rigid models, and is consistent with our thermal conductivity results. The discrepancy between the relative difference in the thermal resistances (around 60%) and the thermal conductivities (more than 200%) could be due to the fact that the thermal resistances were calculated using the NEMD Müller-Plathe algorithm, whereas the thermal conductivities were calculated using the EMD Green-Kubo algorithm. A second contributing factor could be the difference in geometry: whereas the nanocolloid

Table 2.4. Thermal resistance R at water-Cu interface for different water potentials. σ_R is the associated error in the measurement.

Model	$R(Km^2/W \times 10^{-9})$	σ_R
TIP3P	2.73	0.06
SPC/E	1.94	0.06
TIP4P/2005	1.68	0.05

contains highly curved surfaces, the thermal resistances are calculated for flat surfaces.

The findings presented in this chapter pointed to the fact that the utility of molecular dynamics depends entirely on the accuracy of the potential energy surface and the ability to calculate it efficiently, and that accurate potentials are needed to be able to investigate complex physical phenomena involved in multi-species systems. One approach to combining high accuracy with computational efficiency using machine learning is presented in the following chapter.

3. A NOVEL APPROACH TO DESCRIBE CHEMICAL ENVIRONMENTS IN HIGH-DIMENSIONAL NEURAL NETWORK POTENTIALS

3.1. Introduction

Molecular dynamics (MD) simulations are frequently used in computational materials science to study the behaviour of both molecular and bulk systems. These simulations assume that the energy of an atom can be defined as a function of the local atomic environment, and the way this is done can dramatically affect the accuracy and performance of the simulation. The two main approaches in the literature are to calculate the atomic energies and forces using electronic structure calculations [129], resulting in *ab initio* MD, or to pre-define functions describing the atomic interactions [130], resulting in *classical* MD. Perhaps the most popular electronic structure method in the literature is density functional theory (DFT) [131] due to its relative accuracy for condensed matter states. However, the accuracy provided by DFT has a tremendous computational cost that strongly restricts the time and length scales of the simulation. In contrast to ab initio methods, the potentials used in classical MD are generally many orders of magnitude faster to execute. This makes them suitable for longer simulations containing many millions of atoms, allowing the study of more complex phenomena in larger domains. The drawback of such potentials is that they contain a limited number of fitting parameters that are generally calibrated to reproduce the properties of a particular bulk phase, resulting in inaccuracies when simulating complex phenomena such as phase transitions [132], dislocations [45] and interfacial dynamics [133].

Recent interest in machine learning (ML) has encouraged the development of machine learning potentials (MLPs) with the goal of achieving quantum mechanical accuracy while approaching the speed of analytical potentials [42, 43, 47, 67, 134, 135]. These effectively apply nonparametric function regression to some reference data set to

interpolate the potential energy surface (PES) of a local chemical environment. After a training process, the MLP is able to predict the energy of and force on a central atom from a description of the atomic neighborhood. Since ML algorithms can in principle reproduce even subtle many-body relationships, they provide higher flexibility than empirical potentials with a fixed functional form. Moreover, once they are trained on data collected by high-accuracy DFT simulations of a variety of configurations and phases, they can (given suitable coverage of the training data) maintain comparable accuracy during MD simulations with less computational expense than ab initio MD; they have been reported to be up to five orders of magnitude faster than quantum mechanical simulations with comparable accuracy [136–138].

Typical reference data to train a MLP consists of a central atom's local chemical information (relative positions of neighboring atoms) and potential energy. Similar to analytical potentials, the potential energy is assumed to depend only on the environment within some cutoff radius and neighbors outside this volume are ignored. This atom-centered approach was initially proposed by Behler–Parrinello (BP) [42], and enables one to calculate the total energy of a given system by summing over all the atoms. The preparation of the training data is crucial for an accurate representation of the PES, and DFT simulations are usually employed to calculate target energies and forces with high accuracy. Considering the cost of these DFT simulations, preparation of the training data is the most computationally demanding part of MLP development.

There are two key steps in the construction of a suitable reference set. First, since ML algorithms do not extrapolate as well as they interpolate, the space of local atomic environments should be widely sampled to increase the transferability of the potential. Several procedures have been proposed to construct the set of reference points used in training. Pukrittayakamee et al. [139] used an importance sampling technique that selects training configurations based on atomic accelerations in MD simulations. This increases the sampling frequency where the potential energy gradient is large, i.e., where the PES is rapidly changing and could otherwise be sparsely sampled. Behler et al. [55] attempted to equitably sample different regions of the configuration space by constructing a training set that included a mixture of crystal structures with different

lattice parameters, amorphous structures, and some structures derived from metadynamics simulations. Another sampling technique that increases both the validity and accuracy of neural network potentials (NNPs) is extending the training set iteratively in a self-consistent way by detecting regions on the PES where the NNP performs poorly. Raff et al. [140] employed a primitive NNP in MD simulations to produce new trajectories. Energies associated with these new trajectories were then calculated with an ab initio method. Configurations from trajectories where contradictions occurred were added to the reference set, and a new NNP was trained. The procedure was initialized with an empirical potential to obtain the first target energies, but the overall method was shown to be independent of the initial potential. They also claimed that most of the points in the configuration space are redundant and only a small subset of possible configurations needs to be sampled, and devised a novelty sampling algorithm to compute a set of possible trajectories in MD simulations. Behler suggested that multiple neural networks (NNs) could be used to identify poorly represented regions on the PES by identifying regions where they conflicted, and appending these configurations to the training set [137]. Both iterative approaches were found to enhance the performance of a given NNP and can be employed in conjunction.

The second major requirement for an MLP is some description of the local chemical environment as a set of real-valued numbers known as 'descriptors'. Machine learning algorithms are unaware of the physical properties of the data by design, and training can be dramatically simplified by appropriate preconditioning of the inputs. For an MLP, the description of the local chemical environment should be invariant with respect to fundamental physical symmetries including translations and rotations of the coordinate system and permutations of the atomic labels. If invariance to these symmetries is not enforced during the construction of the descriptors, the MLP could predict that physically identical configurations have different energies. Let $\left\{\bar{r}^{\,i1}, \bar{r}^{\,i2}, ..., \bar{r}^{\,iN}\right\}$ be the relative positions of the neighbors around the ith atom. These are usually converted into a vector of real-valued numbers $\left\{G^i_1, G^i_2, ..., G^i_{N_d}\right\}$ that are invariant to the physical symmetries. A recent review of MLPs and local structural descriptors by Behler [141] included an overview of the Behler–Parrinello (BP) symmetry functions [42], one of the first sets of descriptors proposed and widely used in the literature [44,55,142,143]. Sep-

arately, Bartók et al. [63] reviewed several descriptors commonly used in the literature, proposed the smooth overlap of atomic position (SOAP) descriptors to represent atomic environments, and quantitatively compared several variations using ad hoc tests. The SOAP descriptors have been increasingly used in the literature, both within [47, 144] and without [65, 145] the context of MLPs.

Since the literature on MLPs is relatively immature, there remains the possibility that local structural descriptors could be further improved. This chapter proposes a set of local structural descriptors that are found to contain considerably more information than the BP descriptors and to be considerably more efficient to evaluate than the SOAP descriptors. The proposed descriptors were integrated into a NNP for solid-state silicon which was implemented as a new pair-style for large-scale atomic/molecular massively parallel simulator (LAMMPS) [119] and validated in MD simulations. The neural nets were constructed and trained in Python using Keras with the TensorFlow backend [146, 147]. Since the main subject of this study is the descriptors rather than the potential, the energies of configurations in the reference set were calculated by means of an empirical potential [16] to reduce the computational cost. These usually would have been calculated with ab initio methods to achieve higher accuracy, but at the price of more uncertainty in the systematic error. Finally, the performance of the NNP using the proposed descriptors was compared with that of comparable NNPs using the BP and SOAP descriptors.

3.2. Method

3.2.1. Descriptors

The faithfulness of any MLP depends strongly on how accurately the local structural descriptors describe the atomic neighborhood. A robust description would ideally provide a one-to-one mapping (bijection) between atomic positions and descriptors up to the symmetries of the physical system. This section introduces a new set of local structural descriptors that are continuous, twice-differentiable, and invariant to the physical symmetries identified in Section 3.1.

Many steps in the construction resemble those for the SOAP descriptors [63]. The first step is to map the list of relative atomic coordinates to a neighbor density function

$$\rho^k(\bar{r}) = \sum_j \omega_j^k \delta(\bar{r} - \bar{r}^{\,ij}) \tag{3.1}$$

for a central atom i, thereby handling the permutation symmetries. The summation is over all neighbors j within a spherical region defined by the cutoff radius r_c, which realizes the physical assumption that atomic energies should depend only on the local environment. Since all configurations are atom-centered, the neighbor position vectors $\bar{r}^{\,ij}$ are defined relative to the central atom. The weight factor ω_j^k could be used to distinguish the kth species in a multi-component system, but for simplicity is set to one, and the superscript on $\rho^k(\bar{r})$ is dropped in the following.

The second step is to project $\rho(\bar{r})$ onto some set of orthonormal basis functions on the ball of radius r_c. Similar to Bartók et al. [63], this projection is carried out by expanding $\rho(\bar{r})$ as

$$\rho(\bar{r}) \approx \sum_{n=0}^{n_{max}} \sum_{l=0}^{l_{max}} \sum_{m=-l}^{l} c_{nlm} g_n(r) Y_l^m(\theta, \phi), \tag{3.2}$$

where $Y_l^m(\theta, \phi)$ is a spherical harmonic and n_{max} and l_{max} are hyperparameters specifying the respective radial and angular resolutions. Although orthogonal radial basis functions should be preferred to minimize redundant information, Bartók et al. [63] neglected the appropriate weight factor for the spherical coordinate system and did not select orthogonal radial basis functions for their SO(3) and bispectrum descriptors, perhaps explaining the poor performance of these descriptors in their numerical experiments. Subsequent publications involving SOAP descriptors [148] do use orthogonal radial basis functions, as discussed further in Section 3.3.2. Apart from orthogonality, the radial basis functions should be defined to have vanishing values and first and second derivatives at the cutoff to ensure continuity of forces and accelerations [149]. Motivated by these requirements, we propose a set of radial basis functions constructed

from linear combinations of the spherical Bessel functions.

Let $f_{nl}(r)$ be the linear combination of spherical Bessel functions

$$f_{nl}(r) = a_{nl} j_l \left(r \frac{u_{ln}}{r_c} \right) + b_{nl} j_l \left(r \frac{u_{l,n+1}}{r_c} \right) \tag{3.3}$$

where a_{nl} and b_{nl} are constants, $j_l(r)$ is the lth spherical Bessel function of the first kind, u_{ln} is the nth root of $j_l(r)$, and r_c is the cutoff. Since $f_{nl}(r_c) = 0$ by definition, the objective is to find a_{nl} and b_{nl} such that $f'_{nl}(r_c) = 0$ and $f''_{nl}(r_c) = 0$. Combining the two differentiation rules for spherical Bessel functions in Appendix A and solving for the roots of first and second derivatives indicates that both conditions can be satisfied if the coefficients in Equation 3.3 satisfy

$$a_{nl} = -\frac{u_{l,n+1}}{j_{l+1}(u_{ln})} c_{nl} \tag{3.4}$$

$$b_{nl} = \frac{u_{ln}}{j_{l+1}(u_{l,n+1})} c_{nl} \tag{3.5}$$

for an arbitrary multiplicative constant c_{nl}. The value of c_{nl} can be fixed by requiring that the $f_{nl}(r)$ be normalized with respect to the inner product, leading to

$$f_{nl}(r) = \left(\frac{1}{r_c^3} \frac{2}{u_{ln}^2 + u_{l,n+1}^2} \right) \left[-\frac{u_{l,n+1}}{j_{l+1}(u_{ln})} j_l \left(r \frac{u_{ln}}{r_c} \right) + \frac{u_{ln}}{j_{l+1}(u_{l,n+1})} j_l \left(r \frac{u_{l,n+1}}{r_c} \right) \right] \tag{3.6}$$

as an explicit equation for the $f_{nl}(r)$. A set of orthonormal radial basis functions $g_n(r)$ can then be defined by applying the Gram-Schmidt process to the $f_{nl}(r)$ for $0 \leq n \leq n_{max}$, with details provided in Appendix A.

Observing that $j_0(r) = \text{sinc}(r)$ and $u_{0n} = (n+1)\pi$, the evaluation of the radial

basis functions simplifies considerably for the $l = 0$ case. The equation for $f_n(r) = f_{n0}(r)$ reduces to

$$f_n(r) = (-1)^n \frac{\sqrt{2}\pi}{r_c^{3/2}} \frac{(n+1)(n+2)}{\sqrt{(n+1)^2 + (n+2)^2}} \left\{ \mathrm{sinc}\left[r \frac{(n+1)\pi}{r_c} \right] + \mathrm{sinc}\left[r \frac{(n+2)\pi}{r_c} \right] \right\}. \tag{3.7}$$

The radial basis functions $g_n(r) = g_{n0}(r)$ can then be defined by the recursion relations

$$e_n = \frac{n^2(n+2)^2}{4(n+1)^4 + 1} \tag{3.8}$$

$$d_n = 1 - \frac{e_n}{d_{n-1}} \tag{3.9}$$

$$g_n(r) = \frac{1}{\sqrt{d_n}} \left[f_n(r) + \sqrt{\frac{e_n}{d_{n-1}}} g_{n-1}(r) \right], \tag{3.10}$$

initialized with $d_0 = 1$ and $g_0(r) = f_0(r)$. By construction, they satisfy the orthonormality condition

$$\int_0^{r_c} g_{n'}(r) g_n(r) r^2 \mathrm{d}r = \delta_{n'n} \tag{3.11}$$

appropriate for functions on the ball of radius r_c. Several examples of the $g_n(r)$ and their first derivatives are shown in Figure 3.1.

With a suitable set of orthonormal basis functions defined on the ball of radius r_c, the expansion coefficients c_{nlm} in Equation 4.3 can be written in terms of the relative spherical coordinates $(r^{ij}, \theta^{ij}, \phi^{ij})$ of the neighbors of the ith atom:

$$c_{nlm} = \sum_j g_n(r^{ij}) Y_l^{m*}(\theta^{ij}, \phi^{ij}). \tag{3.12}$$

While the c_{nlm} depend on the orientation of the coordinate system, the power spectrum p_{nl} obtained from

$$p_{nl} = \sum_{m=-l}^{l} c_{nlm}^* c_{nlm} \tag{3.13}$$

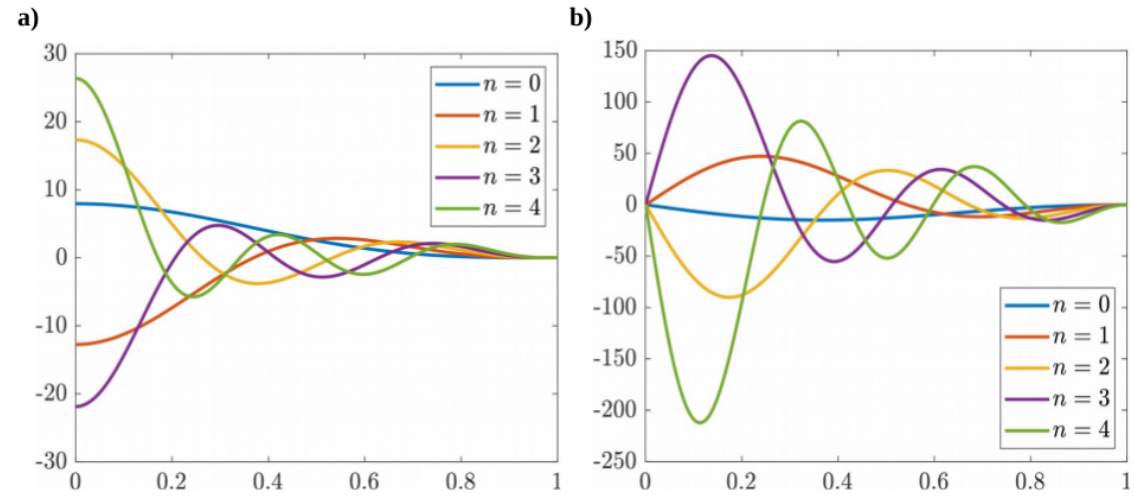

Figure 3.1. The values (a) and the first derivatives (b) of the radial basis functions $g_n(r)$ for $0 \leq n \leq 4$ and $r_c = 1$. The behavior of the functions close to $r = r_c$ indicates that the second derivatives vanish there as well.

is rotationally invariant [63]. We therefore propose to use the real-valued p_{nl} as local structural descriptors for neural networks. The number of descriptors is $(n_{\max} + 1)(l_{\max} + 1)$, and the accuracy of the expansion in Equation 3.2 increases with $n_{\max}$ and $l_{\max}$. That is, larger values of $n_{\max}$ and $l_{\max}$ include more terms in the approximation and more precisely specify the local environment. On the other hand, increasing the number of descriptors increases the cost of evaluating the NNP. While a local environment with $\nu > 1$ neighboring atoms requires precisely $3\nu - 3$ descriptors to describe the relative positions of all the atoms, we observe that more descriptors are often required in practice.

3.2.2. Neural Network Potential

Artificial Neural Networks (ANNs) have experienced a significant surge of interest in the last two decades after their success in various classification and regression problems [37, 150, 151]. In principle, NNs obey a universality theorem in that they are theoretically capable of reproducing any nonlinear functional relationship [152]. This encouraged their use for fitting PESs, where complex nonlinear relationships can exist

between atomic configurations and atomic energies. While several different procedures have been proposed to develop NNPs [137, 153–155], the Behler–Parinello construction [42] is followed here. The total energy E of the system is decomposed into a sum of atomic contributions E^i:

$$E = \sum_i E^i. \tag{3.14}$$

Each atomic energy is calculated from the local chemical environment by an atomic NN. This atom-centered approach enables the modeling of systems with a variable number of atoms, overcoming a limitation of early NNs in the chemistry literature [156–158].

The type of NNs that is generally used for fitting PESs is a *feed-forward neural network* (FFNN) [159] in which information only passes in a single direction towards the output layer. There is one input layer that feeds the relative atomic positions into the network and one output layer containing the atomic potential energy E^i. Some number of intervening hidden layers actually perform the regression, and the number of layers and the number of neurons in each layer are empirically optimized for the intended application. An example of a FFNN with one hidden layer is presented in Figure 3.2.

Let the hyperbolic tangent $h(x) = \tanh(x)$ be the transfer function for the hidden layers. The argument of the jth neuron in the first hidden layer is

$$a_j^1 = b_j^1 + \sum_{k=1}^{N_d} w_{kj}^1 G_k, \tag{3.15}$$

the argument of the jth neuron in the nth hidden layer is

$$a_j^n = b_j^n + \sum_k w_{kj}^n h(a_k^{n-1}), \tag{3.16}$$

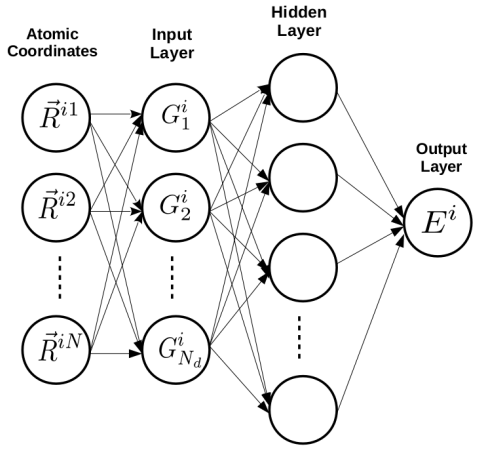

Figure 3.2. Feed-forward neural network scheme used in this study. E^i is the atomic potential energy of the ith atom, G^i are the descriptors of the local environment, $\bar{r}^{ij}$ are the relative position vectors of the neighbors, N is the number of neighbors and N_d is the number of descriptors.

and the value for the output neuron is

$$E^i = b_1^{N_L} + \sum_k w_{k1}^{N_L} h(a_k^{N_L-1}), \tag{3.17}$$

where N_L is the number of layers, w_{kj}^n is the weight that binds the jth neuron in the nth layer to the kth neuron in the $(n-1)$th layer, and b_j^n is the bias for the jth neuron of the nth layer. The weights and biases constitute the parameter space to be fitted during the training of the NN.

The number of hidden layers is of great importance and can dramatically affect both the accuracy and performance of MD simulations. Additional layers enhance the ability of the NN to fit complex functions, but have the drawback of increasing the number of weights and biases to optimize, possibly slowing down or even hindering the training process [160]. Redundant layers and neurons can also cause over-fitting, meaning that the NN becomes less capable of extrapolating to configurations outside of the training set. In regression problems such as PES fitting, this can be a severe problem and significantly reduce the reliability of the NNP in energy and force predictions. Therefore, it is preferable to use the smallest possible number of layers and neurons

that achieve the desired error when building the NN. We decided to use a single hidden layer after observing that additional layers did not substantially improve the fitting accuracy.

The NNs were trained using the standard back-propogation [161] and stochastic gradient descent [162] algorithms. The root mean square error (RMSE)

$$\Gamma = \left[\frac{1}{N_T}\sum_{i=1}^{N_T}(E^i_{\text{pre}} - E^i_{\text{act}})^2\right]^{1/2} \tag{3.18}$$

was used to quantify the error after each epoch, where N_T is the total number of training points and E^i_{pre} and E^i_{act} are the predicted and actual potential energies, respectively. The mini-batch size was 100 for all simulations. All of the training processes were performed in Python using Keras with the TensorFlow backend [146, 147].

Given the atomic energies, the forces acting on each atom can be computed from the gradient of E. This requires repeated application of the chain rule due to the dependence of the descriptors on the atomic positions. Let the jth component of the force on the ith atom be F^i_j. This is obtained by summing the contributions from all N atoms in the system by

$$F^i_j = -\sum_{k=1}^{N}\frac{\partial E^k}{\partial r^i_j} = -\sum_{k=1}^{N}\sum_{p=1}^{N_d}\frac{\partial E^k}{\partial G^k_p}\frac{\partial G^k_p}{\partial r^i_j}, \tag{3.19}$$

where r^i_j is the jth Cartesian coordinate of the ith atom, G^k_p is the pth descriptor for the kth atom, and N_d is the number of descriptors. The derivatives $\partial E^k/\partial G^k_p$ depend only on the NN architecture and can be calculated by back-propogation. The derivatives $\partial G^k_p/\partial r^i_j$ of the proposed descriptors with respect to the Cartesian coordinates and other details of the force calculation are provided in Appendix A. Note that the force includes contributions from the dependence of the neighboring atoms' energies on the position of the ith atom, and from the dependence of the energy of the ith atom on its own position—displacing the ith atom by $\Delta\bar{r}$ effectively displaces the surrounding atoms by -$\Delta\bar{r}$, contributing to the total force.

3.2.3. Training Data

The training data set should generally be prepared carefully, as the selection of configurations to include can significantly affect the performance and accuracy of the NN. If an atomic configuration that is not adequately represented in the reference set occurs during simulation, the error in the predicted potential energy could increase dramatically. This can be addressed by directly sampling points in diverse regions of the configuration space, e.g., by considering all possible structures represented on the phase diagram [55], but there is no guarantee that other configurations would not occur in simulation. A second option would be to employ an importance sampling method to enhance the flexibility and extrapolation capability of NNs. Since our main intention is to investigate the properties of the proposed descriptors rather than develop a general-purpose NNP, training configurations were only sampled from MD simulations of silicon within a limited temperature range. Sampling was performed using the algorithm proposed by Pukrittayakamee et al. [139] and modified by Stende [163], which was observed to reduce the fitting error. The algorithm consists of sampling the local environment around a given atom at a variable interval

$$\varpi = \begin{cases} 1 & |\bar{F}^i| > \beta \\ \lfloor \beta / |\bar{F}^i| \rfloor & |\bar{F}^i| \leq \beta \\ \varpi_{\max} & \lfloor \beta / |\bar{F}^i| \rfloor > \varpi_{\max} \end{cases} \tag{3.20}$$

where $\bar{F}^i$ is the total force acting on the ith atom and ϖ is measured in units of the MD timestep. We tracked ∼10 atoms throughout an MD simulation using the Stillinger–Weber potential [16], calcualted the corresponding forces, and sampled training set configurations at intervals specified by ϖ. The inverse relationship between $|\bar{F}^i|$ and ϖ ensures that high-gradient regions on the PES are more equitably represented in the training data, and is observed to reduce the fitting error. Note that $\varpi_{\max}$ and β are system-dependent parameters.

3.3. Results and Discussion

The performance of the descriptors proposed in Section 3.2.1 in an NNP for solid-state silicon is compared with that of the BP descriptors and the SOAP descriptors. The NNP is further validated by measuring the elastic constants of solid-state silicon. The Stillinger–Weber potential [16] is selected as the ground truth, and was used to calculate the energies of all configurations in the training set.

3.3.1. Behler–Parinello Descriptors

The BP descriptors were one of the earliest sets of descriptors used for MLPs, and are still used for this purpose. Following Behler [42], the radial symmetry functions G_i^r and angular symmetry functions G_i^a are defined as

$$G_i^r = \sum_{j=1}^{N} e^{-\eta(r^{ij}-r_s)^2} f_c(r^{ij}) \tag{3.21}$$

$$G_i^a = 2^{1-\zeta} \sum_{j\neq i} \sum_{k>j} [(1+\lambda \cos\gamma^{jik})^{\zeta} e^{-\eta((r^{ij})^2+(r^{ik})^2)} f_c(r^{ij}) f_c(r^{ik})] \tag{3.22}$$

where $f_c(r^{ij})$ is a cutoff function and η, λ, ζ and r_s are adjustable parameters. These functions are designed to create a set of real-valued numbers from the atomic distances r^{ij} and bond angles γ^{ijk}. A recent study [163] developed a single hidden layer NNP for silicon using the BP descriptors and a 24-10-1 architecture. The performance of this NNP is compared to one using our descriptors with a 25-10-1 architecture (the number of descriptors is not exactly the same because of indexing). The RMSE for the validation set was evaluated as a function of sampling temperature while keeping the training set and all other hyperparameters fixed. The results in Figure 3.3 suggest that our descriptors provide considerably more information about the local environment and result in a more accurate NNP than the BP descriptors. Alternatively, considerably fewer of our descriptors would be required to construct an NNP of a given accuracy, reducing the expense of force calculations in MD simulations. Moreover, the proposed descriptors contain few adjustable parameters ($n_{\max}$, $l_{\max}$ and r_c) and should therefore

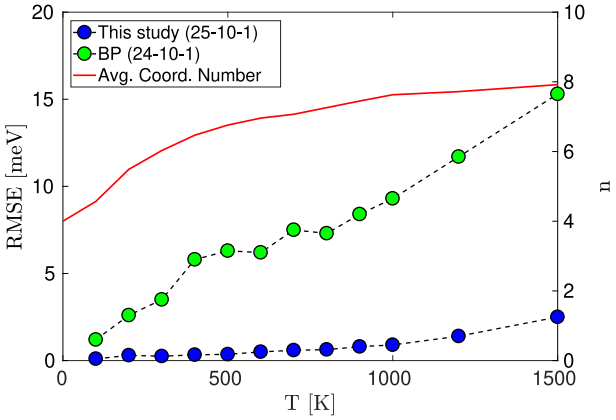

Figure 3.3. Performance of the BP descriptors and the proposed descriptors with increasing temperature. The left and right y-axis show the RMSE in meV and the average number of neighbors n, respectively.

be widely applicable with minimal calibration, whereas the parameters η, λ, ζ and r_s need to be adjusted for the BP descriptors.

We also observed that NNs using the BP descriptors were more difficult to train than ones using our descriptors. Similar to other ML algorithms, NNs often require detailed pre-processing of the input data to obtain reasonable results. One frequent problem is saturation of some hidden neurons during training, resulting in trapping around a local minimum that prevents further learning. This is related to the vanishing gradient problem, which is one of the most common issues with artificial neural networks and happens more frequently when some input values are much larger than others. The wide variation in the magnitudes of the BP descriptors, as indicated by Figure 3.4, likely caused the observed difficulties with training. While Behler suggested several pre-processing techniques to overcome this issue [137], we found that the problem could be solved by initializing the weight matrices with values from the Xavier normal distribution [164] with a variance of $\sqrt{6/(n_{l-1}+n_l)}$ where n_l is the number of neurons in the lth layer. By contrast, training with our descriptors progressed the same regardless of the weight initialization and without any additional pre-processing. The only advantage of the BP descriptors we observed was that they required roughly half the time to evaluate (with our naive implementations), but this seems to be strongly outweighted by the advantage in accuracy.

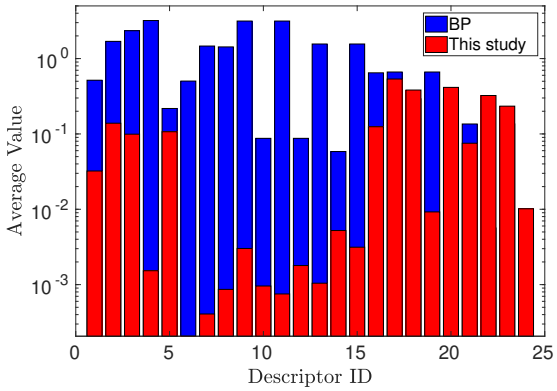

Figure 3.4. Average values of the proposed descriptors and the BP descriptors for a single training data set consisting of 10^4 silicon configurations at 300 K.

3.3.2. SOAP Descriptors

Bartók, Kondor, and Csányi initially introduced the Smooth Overlap of Atomic Positions (SOAP) descriptors [63] to support modeling the PES as a Gaussian process [43]. Rather than directly using the SOAP descriptors as inputs into an MLP though, an inner product of normalized descriptor vectors (the SOAP kernel) is generally employed to measure the similarity of a pair of atomic environments. If $\bar{G}^i$ is the vector of SOAP descriptors for the ith atom, the SOAP kernel κ^{ij} comparing the environments around the ith and jth atoms is defined by means of [148]

$$\hat{G}^i = \bar{G}^i / |\bar{G}^i| \tag{3.23}$$

$$\kappa^{ij} = \sigma_w^2 |\hat{G}^i \cdot \hat{G}^i|^\xi \tag{3.24}$$

where σ_w and ξ are adjustable parameters. The quantity $d^{ij} = \sqrt{1 - \kappa^{ij}}$ has been said to be a metric [65], though the identity property of a metric requires that the distance vanish if and only if the configurations around the ith and jth atoms are identical. Consider the case where, for every atom in the neighborhood of the ith atom, there is a corresponding pair of atoms separated by an arbitrarily small distance δ in the same position relative to the jth atom. The expansion coefficients c_{nlm} for the two configurations would then differ by roughly a factor of two, the SOAP descriptors $\bar{G}^i$ by roughly a factor of four, and the distance d^{ij} could be made arbitrarily close to

zero by adjusting the value of δ. That is, the quantity d^{ij} does not satisfy the identity property and is not a metric. The difficulty seems to be essential in that, if the vectors of SOAP descriptors were not normalized, the magnitude of κ^{ij} would not be bounded above, and the value for which environments are considered similar would no longer be unique. The existence of this counterexample does little to inspire confidence that there are not others, particularly since this is a function in a high-dimensional space where intuition is difficult to develop.

Instead of the SOAP kernel, this study uses the SOAP descriptors as inputs for an NNP. The derivation of the proposed descriptors is closely related to that of the SOAP descriptors in several respects; a neighbor density function is projected onto a set of orthogonal basis functions, and the descriptors are given by inner products of vectors of the expansion coefficients. That said, there are several significant differences. First, the neighbor density function for the proposed descriptors is a sum of Dirac delta functions, whereas that for the SOAP descriptors is a sum of Gaussians. This has the consequence that evaluating the SOAP descriptors involves a relatively expensive numerical integration, whereas the proposed descriptors can be found merely by evaluating the relevant basis functions at the neighboring atoms' positions. Using a sum of Gaussians is said to improve the stability of the SOAP descriptors with respect to perturbations of the atoms' positions [63], but the differentiability of the basis functions in Section 3.2.1 is sufficient to give the proposed descriptors the same property. Second, the SOAP descriptors are given by the inner products of vectors of expansion coefficient with different values of n:

$$p_{n'nl} = \sum_m c^*_{n'lm} c_{nlm}. \tag{3.25}$$

Depending on the radial basis functions this could help to couple information from different spherical shells within the domain, but increases the number of descriptors and the computational expense of evaluating an NNP for fixed n_{max} and l_{max}. The proposed descriptors instead depend on a set of orthonormal basis functions with strong radial overlap (visible in Figure 3.1), obviating the need for such explicit coupling.

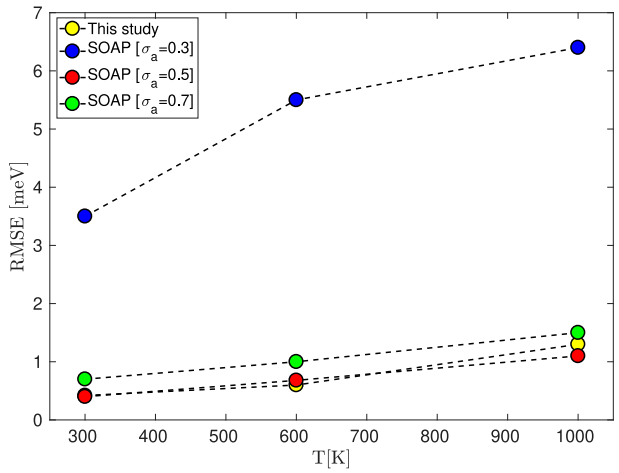

Figure 3.5. Performance of the SOAP descriptors and the proposed descriptors with increasing temperature, with NN architectures of (18-10-1) and (16-10-1), respectively. σ_a is the standard deviation of the Gaussians used to generate the neighbor density function in angstroms. All other fitting parameters for the SOAP descriptors were taken from the literature [148].

Evidence that these differences do not degrade the performance of the proposed descriptors relative to the SOAP descriptors is given in Figure 3.5. The performance of an NNP trained with 16 of the proposed descriptors is nearly identical to that of an NNP trained with 18 of the SOAP descriptors for the optimal value of the Gaussian width in the neighbor density function. Additionally, the proposed descriptors have several advantages that are not visible from this figure. First, computing 16 of the proposed descriptors for 100 training points requires ∼0.2 seconds whereas computing 18 of the SOAP descriptors requires ∼9.5 seconds (with our naive implementations). The special function evaluations and numerical integrations required for the SOAP descriptors would likely be expensive even in optimized code. Second, the second derviatives of the proposed descriptors are continuous to atoms passing through the domain boundary, whereas only the first derivatives are continuous for the SOAP descriptors [148]. This is significant because discontinuous second derivatives of the potential energy have been observed to lead to discontinuous elastic constants and anomalous thermal transport in MD simulations [149]. Third, the proposed descriptors do not require an arbitrary choice of cutoff function, and the number of adjustable parameters is smaller than for the SOAP descriptors. Specifically, the proposed descriptors only require that r_c,

Table 3.1. The minimum RMSE per atom for different temperatures T and NN architectures, where n is the average number of neighbors and N_d is the number of descriptors. All of the neural networks were trained on 8500 training points for 20000 epochs, and RMSE values were obtained on 1500 test configurations that are not included in the training set.

T [K]	n	NN	N_d	RMSE [meV]
300	6.03	16-8-1	16	0.22
300	6.03	16-16-1	16	0.23
300	6.03	25-8-1	25	0.35
600	6.96	16-8-1	16	0.56
600	6.96	16-16-1	16	0.64
600	6.96	25-8-1	25	0.51
1000	7.63	16-8-1	16	1.24
1000	7.63	16-16-1	16	1.98
1000	7.63	25-8-1	25	0.88
1500	7.92	16-8-1	16	2.62
1500	7.92	16-16-1	16	2.75
1500	7.92	25-8-1	25	2.3

n_{max}, and l_{max} be specified, whereas the SOAP descriptors have up to six adjustable parameters [148] if the Gaussian widths in the neighbor density function and in the raw radial basis functions are allowed to be independent. Setting these adjustable parameters introduces additional complexity, with Figure 3.5 showing the sensitivity of NNP performance to the value of one of them.

3.3.3. NNP Validations

We further investigated the performance of NNPs using our descriptors at a variety of sampling temperatures and NN architectures, with the results reported in Table

3.1. The number of accessible configurations in an MD simulation usually increases rapidly with temperature, meaning that any given accuracy would require more descriptors to encode the neighborhoods and training points to cover the configuration space. The parameters n_{max} and l_{max} in Equation 4.3 set the number of descriptors, with higher values resulting in more terms in the approximation of the neighbor density function and generally lower fitting errors. The average number of neighbors n varied from 6 to 8 within the selected temperature range, implying that a minimum of 15 to 21 descriptors were required. This is consistent with our observations that using more than 25 descriptors ($n_{max} = l_{max} = 4$) did not substantially decrease the RMSE, and is consistent with the number of descriptors used in other NNP studies [42, 44, 55, 140]. Moreover, when the temperature was elevated to 1500 K (the melting point of silicon is 1687 K), NNs using 25 descriptors consistently outperformed those using 16 descriptors; the higher average number of neighbors at these temperatures allowed more complex configurations that required more descriptors.

Using more than one hidden layer or more than ten hidden neurons did not substantially improve the accuracy of the NNP. Table 3.1 indicates that using more hidden neurons actually *decreased* the accuracy, perhaps as a consequence of the increased complexity of the training process. This differs from previous studies that used the BP descriptors in two-layer NNPs for single-species systems [42, 44, 55]. Artrith and Behler [44] further mentioned that monocomponent systems typically require 40 to 60 symmetry functions to achieve a complete description, and used 51 in their study. The reason for the difference in behavior with that observed here is not known, but is conjectured to be related to our descriptors deriving from an efficient expansion of the neighbor density function using orthogonal basis functions, and to our radial basis functions effectively coupling information in multiple spherical shells.

Finally, the NNP developed here was added as a new pair-style to LAMMPS. The bulk modulus, shear modulus and Poisson's ratio of solid-state silicon were calculated from an MD simulation using an NNP with our descriptors and a 25-10-1 architecture at 300 K, and compared with those reported for the SW potential. The results in Table 3.2 offer additional evidence that our NNP is able to reproduce features of the

Table 3.2. Bulk modulus (K), shear modulus (S) and Poisson's ratio (PR) of solid-state silicon at 300 K as measured in MD simulations using the analytic SW potential and our NNP.

	K [GPa]	S [GPa]	PR
SW [16]	101.4	56.4	0.335
NNP	101.7 $\pm$0.3	56.3 $\pm$0.1	0.337 $\pm$0.2

potential energy surface with excellent accuracy.

4. CONTINUOUS AND OPTIMALLY COMPLETE DESCRIPTION OF CHEMICAL ENVIRONMENTS USING SPHERICAL BESSEL DESCRIPTORS

4.1. Introduction

Machine learning potentials (MLPs) have recently become a subject of interest in computational materials science [141], potentially offering the accuracy of electronic structure techniques like density functional theory without the associated computational cost. An MLP effectively learns to reproduce the potential energy surface (PES), i.e., the hypersurface that defines the potential energy of an atomic system as a function of the atomic positions. While the reliability of atomistic simulations including molecular dynamics (MD) depends on the accuracy of the PES, their usefulness to study complex phenomena is limited by the accessible time and length scales; in practice this makes the computational cost of an MD simulation nearly as much a concern as the accuracy. Recent studies [44, 45, 64] suggest that MLPs can achieve a favorable combination of performance and accuracy that is provided by neither classical force fields nor electronic structure calculations.

Machine learning (ML) algorithms that have been employed to construct MLPs include artificial neural networks [42,155], support vector machines [165] and Gaussian processes [43]. Regardless of the algorithm, MLPs rely on the reasonable assumption that the energy of an atom is a multidimensional function of the relative positions of the neighboring atoms. This atom-centered approach [42] enables the total energy E of a system to be calculated by summing over all individual atomic energies E^i as

$$E = \sum_i E^i \tag{4.1}$$

and reduces the problem to one involving a local atomic environment. This environment is usually encoded as a set of scalars, known as *descriptors*, that serve as

the inputs for the atom-centered MLPs. Faber et al. [166] carried out a systematic study of how the choice of descriptors and ML algorithm can affect the accuracy of an MLP by testing a variety of combinations. They found that the choice of descriptors could affect the accuracy more than the regression scheme, justifying the effort spent over the last decade in developing the many competing descriptors available in the literature [42, 43, 63, 68, 167–169]. Of these, the Behler-Parinello (BP) symmetry functions [42] and the Smooth Overlap of Atomic Position (SOAP) descriptors [63] are some of the most frequently used, and have been employed in MLPs that achieve the accuracy of electronic structure methods in a variety of applications [47, 60, 170, 171]. Afterwards, Khorshidi et al. proposed to use the Zernike polynomials [167, 172] and the neighbor density function of Bartok et al. [43] to construct the Zernike descriptors, and reported comparable results. Recently, Kocer et al. [76] proposed to use the spherical Bessel functions with a closely related procedure to construct the Spherical Bessel (SB) descriptors. These were found to allow construction of MLPs significantly more accurate than those using the BP symmetry functions, and of comparable accuracy to but an order of magnitude faster to evaluate than those using the SOAP descriptors.

Any set of descriptors should satisfy a number of mathematical properties to not constrain the ability of the ML algorithm to approximate the PES. First, it is desirable from a computational standpoint that they be invariant to the symmetries of the physical system (i.e., translations, rotations, inversions and permutation of atomic labels) to reduce the domain of the PES and the number of training examples required. More subtle but perhaps more important is that the descriptors be similar but distinct for similar but distinct atomic environments; if the descriptors are not similar, the MLP would not likely be continuous, and if the descriptors are not distinct, the MLP would not be able to reproduce potentially significant features of some physical systems. This is closely related to the concept of *completeness*, here defined as the condition that the space of physically-distinct atomic environments is smoothly embedded into the space of descriptors. This is desireable because, e.g., a set of descriptors that is complete allows the atomic environment to be reconstructed up to symmetry. The stronger condition of *optimal completeness* requires that the embedding into the space of descriptors always be achieved with the minimum number of descriptors, and is highly desireable

for computational reasons. Finally, the descriptors should be twice-differentiable to allow for continuity of forces and elastic constants, contain few adjustable parameters to help with transferrability of the potentials, and be numerically efficient to evaluate. To the extent of our knowledge, none of the descriptors available in the literature fulfills all of these requirements.

This chapter presents an updated version of the SB descriptors (requiring only a change in indexing) that makes them continuous with respect to atomic displacements. A necessary condition for optimal completeness is then formulated using the rank theorem [173]. The SB descriptors are found to satisfy this condition, whereas the power spectrum coefficients used in the construction of the SOAP descriptors do not. Finally, the accuracy and efficiency of the SB descriptors in a proof-of-concept MLP are compared to several of the alternatives available in the literature.

4.2. Spherical Bessel Descriptors

Following a similar procedure to our recent study [76], an atomic neighbor density function

$$\rho^k(\bar{r}) = \sum_j \omega_j^k \delta(\bar{r} - \bar{r}^{\,ij}) \tag{4.2}$$

is first defined for a central atom i, where $\bar{r}^{\,ij}$ are the relative position vectors of each neighbor j with respect to i. The weight factor ω_j^k could be used to specify the species of atoms i and j in a multi-component system, but is assumed to be one in this study. The neighbor density function $\rho(\bar{r})$ is projected onto a set of orthonormal basis functions on the ball of radius r_c, giving an expansion of the form

$$\rho(\bar{r}) \approx \sum_{n=0}^{n_{\max}} \sum_{l=0}^{n} \sum_{m=-l}^{l} c_{nlm} g_{n-l,l}(r) Y_l^m(\theta, \phi) \tag{4.3}$$

where $g_{nl}(r)$ is a radial basis function, $Y_l^m(\theta, \phi)$ is a spherical harmonic, and n_{max} specifies the order of the approximation. While many functions could be used for the

$g_{nl}(r)$, the one for the SB descriptors begins with the linear combination

$$f_{nl}(r) = a_{nl} j_l \left(r \frac{u_{ln}}{r_c} \right) + b_{nl} j_l \left(r \frac{u_{l,n+1}}{r_c} \right) \tag{4.4}$$

where a_{nl} and b_{nl} are constants, $j_l(r)$ is the lth spherical Bessel function of the first kind, u_{ln} is the $(n+1)$th nonzero root of $j_l(r)$, and r_c is the cutoff radius. The condition $f_{nl}(r_c) = 0$ is satisfied by definition, and a_{nl} and b_{nl} can surprisingly be chosen to simultaneously satisfy the conditions $f'_{nl}(r_c) = 0$ and $f''_{nl}(r_c) = 0$, i.e., to make the radial basis functions twice differentiable at the cutoff radius. Along with normalization, this leads to

$$f_{nl}(r) = \left(\frac{1}{r_c^3} \frac{2}{u_{ln}^2 + u_{l,n+1}^2} \right)^{1/2} \left[\frac{u_{l,n+1}}{j_{l+1}(u_{ln})} j_l \left(r \frac{u_{ln}}{r_c} \right) - \frac{u_{ln}}{j_{l+1}(u_{l,n+1})} j_l \left(r \frac{u_{l,n+1}}{r_c} \right) \right]. \tag{4.5}$$

The radial basis functions $g_{nl}(r)$ are then obtained by applying a Gram-Schmidt process to the $f_{nl}(r)$ for $0 \leq n \leq n_{max}$. A detailed derivation of the $g_{nl}(r)$ and explicit recursion relations that allow them to be efficiently evaluated are provided in Appendix B. Given the radial and angular basis functions in Equation 4.3, the expansion coefficients c_{nlm} for the ith atom are calculated from the relative spherical coordinates $(r^{ij}, \theta^{ij}, \phi^{ij})$ of the neighboring atoms as

$$c_{nlm} = \sum_j g_{n-l,l}(r^{ij}) Y_l^{m*}(\theta^{ij}, \phi^{ij}) \tag{4.6}$$

by means of the standard orthogonality relations. The power spectrum p_{nl} obtained from

$$p_{nl} = \sum_{m=-l}^{l} c_{nlm}^* c_{nlm} \tag{4.7}$$

then comprises an infinite set of real-valued numbers that are used as the local structural descriptors. They are invariant to translations by the use of relative spherical coordinates, and to permutations of atomic labels by the construction of the neighbor density function in Equation 4.2. Invariance to rotations and inversions can be seen

by substituting Equation 4.6 into Equation 4.7 and reordering the summations to find

$$p_{nl} = \sum_j \sum_k g_{n-l,l}(r^j) g_{n-l,l}(r^k) \sum_{m=-l}^{l} \left[Y_l^m(\theta^j, \phi^j) Y_l^{m*}(\theta^k, \phi^k) \right] \tag{4.8}$$

where the subscript i is suppressed for clarity. The spherical harmonic addition theorem [174] allows this to be reduced to

$$p_{nl} = \frac{2l+1}{4\pi} \sum_j \sum_k g_{n-l,l}(r^j) g_{n-l,l}(r^k) P_l(\cos \gamma^{jk}) \tag{4.9}$$

where P_l is the Legendre polynomial of order l and γ^{jk} is the triplet angle between atoms i, j and k. Since the radial distances and triplet angles that constitute the independent variables in Equation 4.9 are invariant to rotations and inversions, the p_{nl} necessarily have the same property. A second reason to consider Equation 4.9 as defining the p_{nl} is that Equation 4.9 is much more efficient to evaluate than Equations 4.6 and 4.7.

The original definition [76] of the radial basis functions $g_{nl}(r)$ included the constraint $l = 0$, removing the coupling between the angular and radial parts in Equation 4.3. While this significantly simplified the evaluation without an observable effect on the accuracy of the MLP for that specific set of training data, this also introduced discontinuities around the origin (near the central atom) in the basis functions in Equation 4.3 for odd l (Figure 4.1.a). Precisely the same issue occurs for the basis functions used in the SOAP descriptors (Figure 4.1.b), and is likely the motivation for using a superposition of Gaussians in the neighbor density function [63] to smooth over the discontinuity. This approach comes at the price of expensive numerical integrations when evaluating the descriptors though, and introduces additional adjustable parameters. The proposed SB descriptors instead use basis functions that are twice differentiable everywhere (Figure 4.1.c), allowing the neighbor density function to be written as a superposition of Dirac delta functions and the descriptors to be calculated at least an order of magnitude faster. Specifically, the MATLAB implementations of the SB descriptors and the SOAP descriptors provided in the supplementary material

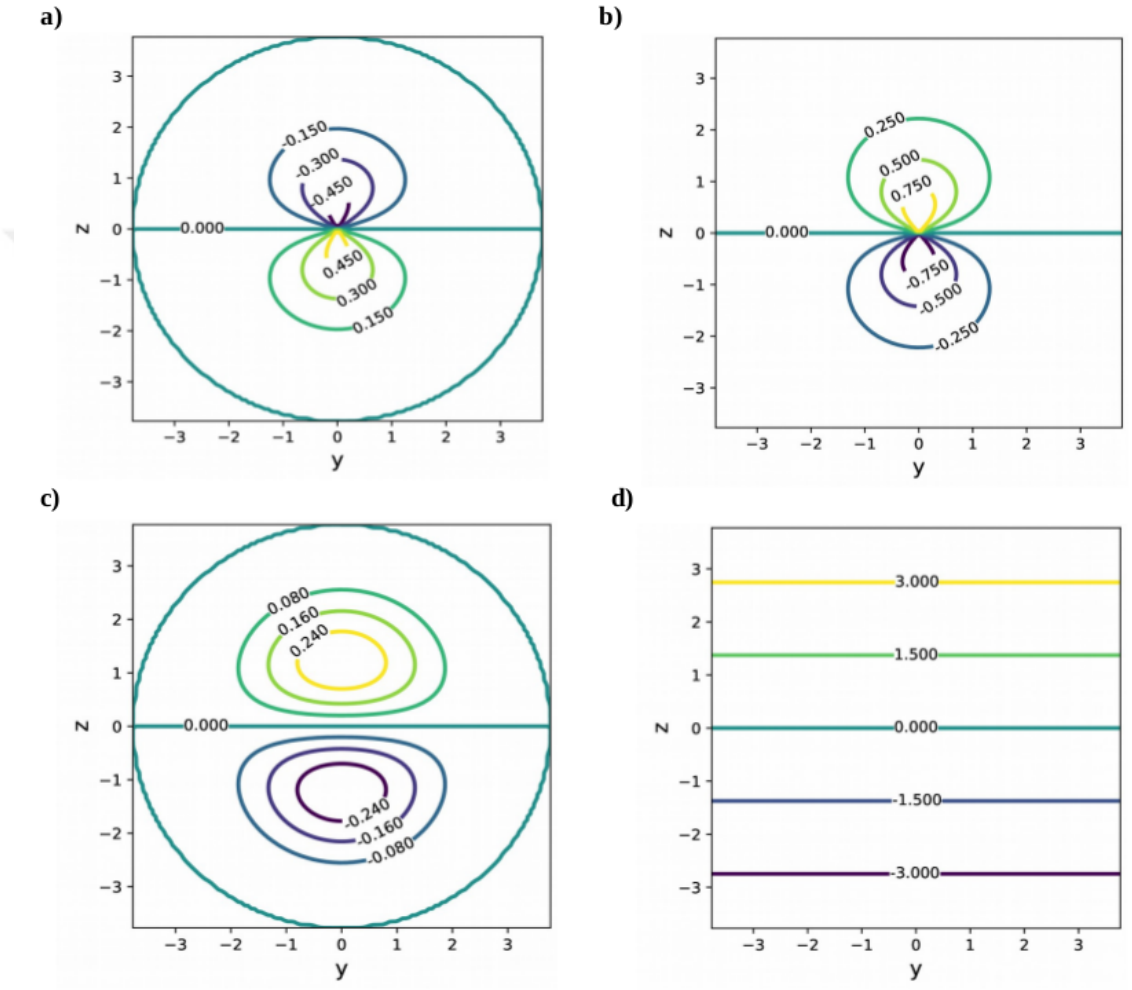

Figure 4.1. Contour plots on the yz plane of the basis functions used to construct (a) the previous SB descriptors [76] for $n = 0$, $l = 1$ and $m = 0$ ($g_{00}Y_1^0$), (b) the SOAP descriptors [148] for $n = 0$, $l = 1$ and $m = 0$, (c) the current SB descriptors for $n = 1$, $l = 1$ and $m = 0$ ($g_{01}Y_1^0$), and (d) the Zernike descriptors [172] for $n = 1$, $l = 1$ and $m = 0$. There is a visible discontinuity at the origin in (a) and (b).

respectively required 1.15 and 16.1 seconds on a 2.60GHz CPU to calculate a comparable number of descriptors for 1000 atomic environments. The implementation of the SOAP descriptors follows that of standard references [63,148], and employed a custom implementation of the double exponential integration technique [175] to accelerate the numerical integration.

Part of the appeal of the SOAP descriptors is that they leave a number of choices up to the practitioner. With specific regard to computational efficiency, a recent publication uses several approximations and a particular choice of radial basis functions to calculate the SOAP descriptors without numerical integration [176]. While this approach is indeed more efficient, the effect of the required approximations is unclear, all of the basis functions with odd values of l contain discontinuities at the origin of the type in Figure 4.1.b, and the orthogonalization of the radial basis functions does not include the appropriate weight factor for the spherical coordinate system, propagating a mistake made in the prior literature [63]. For these reasons, this particular version of the SOAP descriptors will not be considered further.

The basis functions of the Zernike descriptors [172,177] are known as the Zernike polynomials, are rotationally-invariant orthogonal polynomials in x, y and z, and do not contain any discontinuity within the cutoff sphere (Figure 4.1.d). While the Zernike polynomials have several other desireable properties, they do not vanish at the cutoff radius and actually oscillate most rapidly there for higher n and l. This effectively concentrates their ability to resolve atomic positions in the regions furthest from the central atom, where intuition suggests that the dependence of the potential energy on atomic position should be weakest. While using a cutoff function in the definition of the neighbor density function does make the descriptors differentiable as atoms leave the environment, this also discards much of the information from the boundary region precisely where the Zernike polynomials are most sensitive and introduces additional adjustable parameters. The relative performance of the Zernike descriptors is considered further in Section 4.4.

4.3. Completeness

Of the desirable mathematical properties of a set of descriptors identified in Section 4.1, the most difficult one to establish is completeness. This word means different things for the basis functions and the descriptors though. For the basis functions, completeness indicates that the expansion in Equation 4.3 is over a complete orthonormal basis, i.e., that the expansion converges for any piecewise-continuous square-integrable function on the ball of radius r_c. The functions

$$\psi_{nlm}(r,\theta,\phi) = N_n^{(l)} j_l\left(r\frac{u_{ln}}{r_c}\right) Y_l^m(\theta,\phi) \tag{4.10}$$

where $N_n^{(l)}$ is a normalizing constant are known to form a complete orthonormal basis for square-integrable functions on this domain [178, 179]. Since the proposed radial basis functions $g_{nl}(r)$ are derived by projecting the $j_l(r\frac{u_{ln}}{r_c})$ onto the space of functions with vanishing first and second derivatives at the boundary and constructing an orthonormal basis from the result, the basis functions in Equation 4.3 constitute a complete orthonormal basis for square-integrable functions with the given boundary conditions as well.

With regard to the descriptors, completeness is usually considered to indicate whether the descriptors can be used to faithfully reconstruct a given local atomic environment up to symmetry. In this thesis, completeness is defined instead by whether the space of physically-distinct atomic environments is smoothly embedded by a map into the space of descriptors. This necessarily implies that there is an inverse map that allows the atomic environment to be reconstructed up to symmetry, and moreover that the map and its inverse are both continuous and differentiable. Observe that a local atomic environment with ν neighboring atoms is specified by 3ν distinct relative spherical coordinates, but only $3(\nu-1)$ quantities (e.g., the ν radial coordinates and $2\nu-3$ triplet angles) are required to specify the environment up to rotations. This means that the space of physically-distinct atomic environments is $3(\nu-1)$-dimensional, and a complete set of descriptors maps this space to a $3(\nu-1)$-dimensional submanifold

in the space of descriptors. Additionally, if the embedding is achieved using only the first $3(\nu - 1)$ of the descriptors for any $\nu \geq 2$, then the descriptors are said to be optimally complete. Intuitively, an optimally complete set of descriptors encodes all relevant information (and just this information) about the atomic environment as concisely as possible.

There is limited discussion of completeness in the literature. One exception is the proof by Shapeev [72] that any rotation- and permutation-invariant polynomial can be written as a linear combination of the moment tensor descriptors, implying that these descriptors are complete (though probably not optimally complete). A recent dimensionality-reduction study [180] also touches on this question, attempting to optimize an MLP by reducing the dimension of the feature space. The possibility of such a reduction indicates that the BP and SOAP descriptors considered there contain substantial redundant information.

While proving that a set of descriptors is complete using the definition above is quite difficult, there is a necessary (but not necessarily sufficient) condition for completeness and optimal completeness that can be readily evaulated for any set of descriptors. This involves using the rank theorem [173] (a generalization of the implicit function theorem) to establish that the map from the space of physically-distinct atomic environments into the space of descriptors is locally invertible. More precisely, the condition establishes that for any particular atomic environment there is a set of closely-related and physically-distinct atomic environments over which the map into the space of descriptors is invertible, and that the inverse map is continuously differentiable. Practically speaking, this involves finding the rank r of the Jacobian matrix J of the function that transforms the relative atomic coordinates into the vector of descriptors. Assume for the moment that r is constant. If $r < 3(\nu - 1)$, then the descriptors discard relevant information and cannot be complete. If $r > 3(\nu - 1)$, then the numerical calculation is faulty and should be checked. If $r = 3(\nu - 1)$, then the descriptors satisfy the necessary condition to be complete (though this local property does not necessarily extend to a global one). With regard to optimal completeness, let $J^{[q]}$ be the matrix formed by taking the first q rows of J. If the rank of $J^{[3(\nu-1)]}$

is $3(\nu-1)$ for any $\nu \geq 2$, then the descriptors satisfy the necessary condition to be optimally complete.

Consider the p_{nl} for the local atomic environment around the ith atom. From Equation 4.9, the p_{nl} can be written as a function of the relative spherical coordinates of the neighboring atoms as

$$p_{nl} = \frac{2l+1}{4\pi} \sum_j \sum_k g_{n-l,l}(r_j) g_{n-l,l}(r_k) P_l \left[(\cos\theta_j \cos\theta_k + \sin\theta_j \sin\theta_k \cos(\phi_j - \phi_k) \right] \tag{4.11}$$

using trigonometric identities. This defines a map from the 3ν-dimensional space of relative spherical coordinates into the infinite-dimensional space of the descriptors p_{nl}. Let the Jacobian matrix of this map be constructed with rows labelled by the pairs (n,l) in lexicographic order, where the (n,l)th row contains the 3ν partial derivatives of p_{nl} with respect to the relative spherical coordinates (provided in the supplementary material). In practice, $J^{[q]}$ is constructed to consider the information content of only the first q descriptors, and the rank of $J^{[q]}$ is found by performing singular value decomposition and counting the singular values that are substantially larger than the machine precision.

As a specific example, we generated an atomic environment with six randomly-positioned neighbors around a central atom and constructed $J^{[q]}$ for $13 \leq q \leq 17$. The significant and insignificant singular values s_i are distinguished by plotting $\log(s_i)$ in Figure 4.2 and observing where the decay to machine precision occurs. The sharp drop after $q = 15$ clearly indicates that the rank of J is $3(\nu-1) = 15$, and similar results are obtained for different environments and different numbers of neighbors. This strongly suggests that the p_{nl} satisfy the necessary conditions developed above for completeness and optimal completeness.

The above analysis is somewhat complicated by the differentiability of the p_{nl} as atoms pass through the boundary at $r = r_c$; the differentiability of the p_{nl} implies that the s_i are continuous, and the number of significant s_i should be reduced by three as an

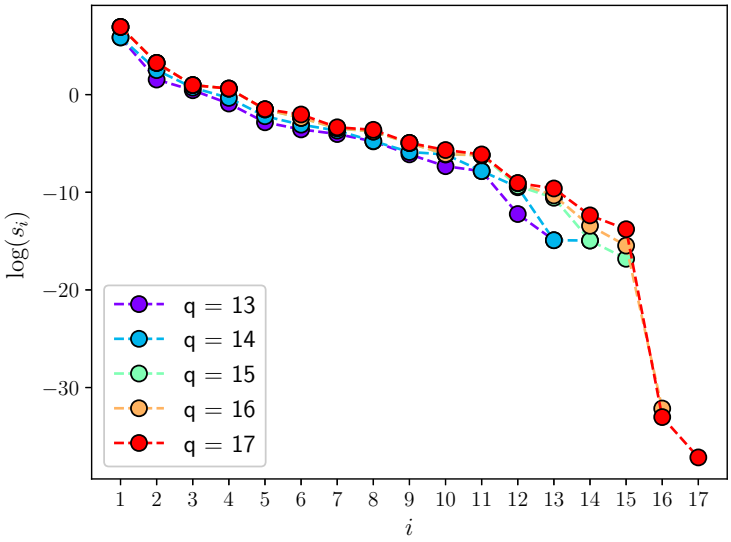

Figure 4.2. The logarithmic singular values $\log(s_i)$ of $J^{[q]}$ for the p_{nl} descriptors as a function of q. The decay of the s_i to the machine precision for $i > 15$ indicates that the rank of J is 15.

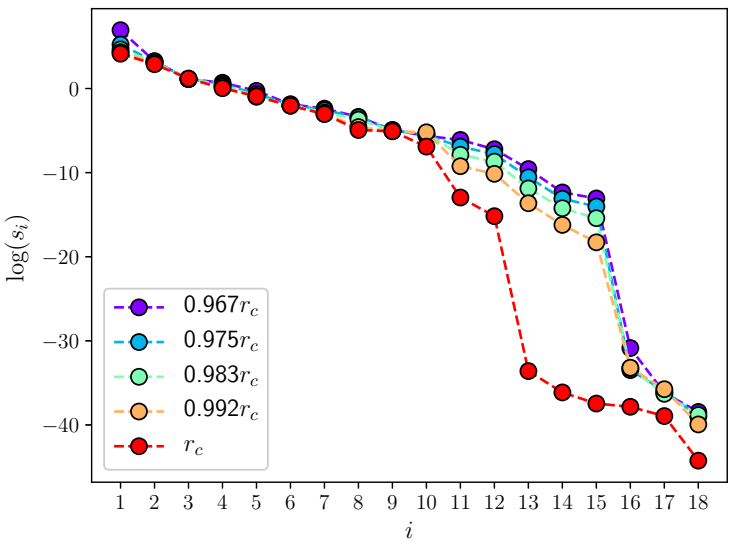

Figure 4.3. The logarithmic singular values $\log(s_i)$ of $J^{[18]}$ for the p_{nl} descriptors as a function of q. Five configurations were generated by scaling the initial configuration such that the radial coordinate of the most distant atom ranged from $0.967r_c$ to r_c.

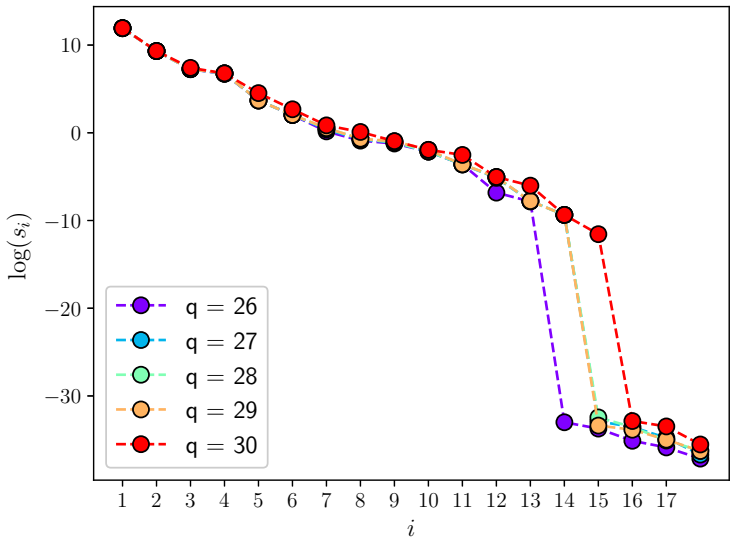

Figure 4.4. The logarithmic singular values $\log(s_i)$ of $J^{[q]}$ for the SOAP descriptors as a function of q indexed by $0 \leq n_1 \leq n_{\max}$, $0 \leq n_2 \leq n_{\max}$ and $0 \leq l \leq l_{\max}$ with the ordering described in the text.

atom leaves the environment. This situation is considered in Figure 4.3, where q is fixed at 18 and the $\log(s_i)$ are plotted as the most distant atom approaches the boundary. This shows that the rank behaves as expected, and moreover that the descriptors contain significant information even about atoms very close to the boundary.

While a comprehensive review of other descriptors in the literature is beyond the scope of this thesis, it is certainly not the case that they all satisfy the necessary conditions for completeness and optimal completeness. For example, the Coulomb matrix [68] for an atomic environment with ν neighbors around a central atom only has $\nu + 1$ eigenvalues, meaning that this descriptor could not possibly be complete. The Behler-Parinello descriptors [42] instead map the space of physically-distinct atomic environments into an infinite-dimensional feature space, but since the dimensions of the feature space do not have a canonical ordering the question of whether they are complete or not is difficult to answer. The SOAP descriptors (i.e., the descriptor vector in Szlachta et al. [148]) are not subject to either of these limitations and therefore provide a suitable basis for comparison with the SB descriptors. The analysis of the SOAP descriptors performed below is intended to provide practical evidence that the

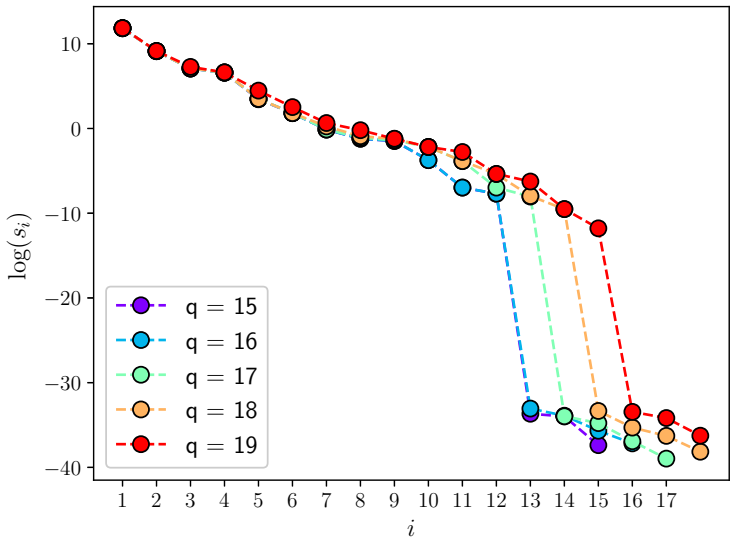

Figure 4.5. The logarithmic singular values $\log(s_i)$ of $J^{[q]}$ for the SOAP descriptors as a function of q indexed by $0 \leq n_1 \leq n_{\max}$, $0 \leq n_2 \leq n_{\max}$ and $0 \leq l \leq l_{\max}$ with the additional constraint that $n_1 \geq n_2$.

necessary conditions for completeness and optimal completeness developed in this thesis are not trivially satisfied.

The SOAP descriptors $p_{n_1 n_2 l}$ depend on two indices $0 \leq n_1 \leq n_{\max}$ and $0 \leq n_2 \leq n_{\max}$ that relate to radial information and one $0 \leq l \leq l_{\max}$ that relates to angular information. This article orders the SOAP descriptors by increasing $p = n_1 + n_2 + l$, increasing l for a given p, and lexicographically in (n_1, n_2) for given p and l. The various adjustable parameters are set to the same values as in Szlachta et al. [148], and the Jacobian matrix of the map from the space of relative spherical coordinates into the space of the descriptors $p_{n_1 n_2 l}$ is constructed using the partial derivatives provided in the supplementary material. Figure 4.4 shows the logarithmic singular values of $J^{[q]}$ for an environment with six randomly-positioned neighbors, and shows that the completeness condition ($r = 15$) is only satisfied for $q \geq 30$. That is, the SOAP descriptors as initially proposed are likely to be complete, but not optimally complete.

Although some of the literature does not place further restrictions on n_1 and n_2 [180], inspection reveals that $p_{n_1 n_2 l}$ and $p_{n_2 n_1 l}$ encode identical information. That

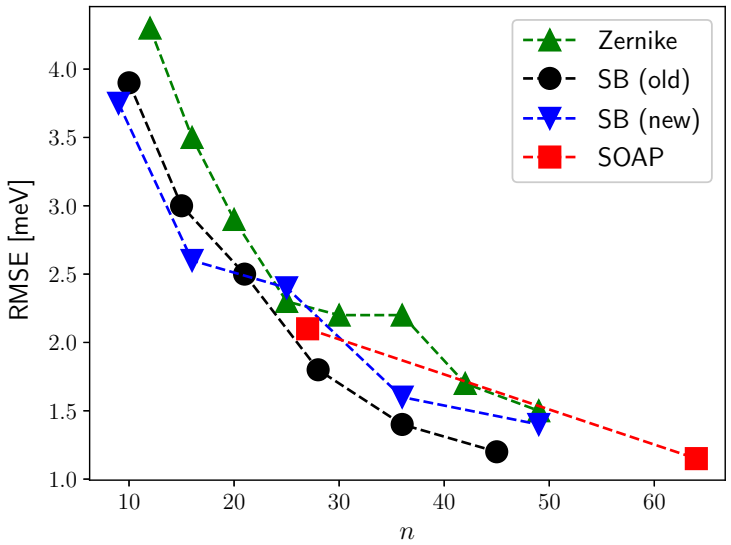

Figure 4.6. Comparison of NNP performance for (n-10-1) architectures, where n is the number of descriptors, for the four specified sets of descriptors. RMSE values are presented as a function of n for 1500 test points after 20,000 training cycles. The values of n do not coincide because of differing indexing schemes.

is, nearly half of the SOAP descriptors are trivially redundant and could be excluded by enforcing the constraint $n_1 \geq n_2$. Other references [47] do not explicitly specify whether this constraint is used, though the QUIP package[1] uses this property when calculating the SOAP kernel. Figure 4.5 shows that enforcing this constraint improves the situation considerably ($r = 15$ for $q \geq 19$), though there are still dependent terms. The difference in the behavior of the SOAP descriptors in Figs. 4.4 and 4.5 and the SB descriptors in Fig. 4.2 is likely a consequence of the differences in radial basis functions and indexing schemes, and is consistent with the redundancy in the SOAP descriptors reported elsewhere [180].

4.4. Performance and Efficiency

Since the intention is for the SB descriptors to be used as inputs for an MLP, a high-dimensional neural network potential (NNP) was constructed for solid-state silicon using the procedure described in Sections IIB and IIC of Kocer et al. [76]. The

[1] https://github.com/libAtoms/QUIP

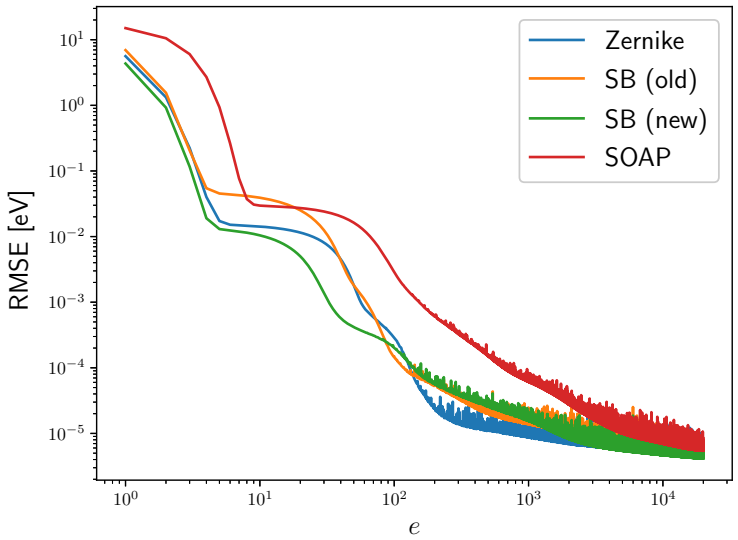

Figure 4.7. RMSE values are presented as a function of training cycles e for NNPs with a fixed (25-10-1) architecture and 8500 training points. While all four converged by $20,000$ training cycles, the SOAP descriptors required the most cycles.

training data consisted of 8500 environments with $r_c = 3.7711\,\text{Å}$ sampled from multiple MD simulations of 1000 Si atoms using the Stillinger–Weber potential [16] equilibrated at 0 Pa and 1500 K. There were an average of 7.9 neighboring atoms per environment for the specified cutoff radius. The NNP performance was evaluated by means of the root mean square deviation of the predicted energies from the calulated ones for an additional 1500 environments. Corresponding NNPs were constructed using the same data and training procedure for the prior SB descriptors [76] ($r_c = 3.7711\,\text{Å}, n_{\text{max}} = l_{\text{max}}$), the SOAP descriptors [148] ($\sigma_{\text{atom}} = 0.5\,\text{Å}$, $r_\Delta = 1.0\,\text{Å}, n_{\text{max}} = l_{\text{max}}$), and the Zernike descriptors [172] ($r_c = 3.7711\,\text{Å}$) (the BP descriptors were considered previously [76]).

The convergence of the NNPs with training cycles is shown in terms of the RMSE for the training set in Fig. 4.7, where the number of descriptors is fixed at 25. The SOAP descriptors required more training cycles to reach convergence than the others; this is perhaps related to the larger variances of the SOAP descriptor values. After ensuring that all NNPs converged by $20,000$ training cycles, the performance of the NNPs on the test set was evaluated as a function of the number of descriptors. The results are

given in Fig. 4.6, though they should not be construed as ranking the effectiveness of the various descriptors; that would require both a more complex underlying PES and a thorough characterization of the random error involved in the training procedure. Instead, our purpose is to show that the SB descriptors would likely perform at least as well as other descriptors already used in the literature to construct MLPs. That said, the slight improvement in the performance of the current SB descriptors is attributed to the elimination of the discontinuity at the origin of the basis functions and to the change in the order of summation in Eq. 4.3.

One of the advantages of the SB descriptors is that while they allow for the construction of MLPs of comparable accuracy to those using the SOAP descriptors, the SB descriptors are much faster to evaluate. Apart from the MATLAB implementations used for the timing experiments reported in Sec. 4.2, a highly optimized C library with MATLAB and Python interfaces has been developed to calculate the SB descriptors and is publicly available[2] .

[2]https://github.com/harharkh/sb_desc

5. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

5.1. Conclusions

This thesis provides a detailed presentation of the author's MSc studies, mainly concentrated on describing local chemical environments in machine learning potentials.

First, the thermal conductivity of a water–Cu nanocolloid system was studied using the Green–Kubo method in Chapter 2, with the intention of identifying the source of the anomalously high nanofluid thermal enhancements in the literature. This is crucial for the validity of the conclusions drawn on the basis of such simulation results. A detailed error analysis identified different sources of statistical errors, denoted as short-time errors and long-time errors. The magnitude of the error was estimated for both, and the long-time error associated with differences in velocity seeding was found to be larger than the short-time error associated with the fluctuations of a single autocorrelation function.

The anomalous thermal enhancement in the literature was reproduced using rigid SPC/E and TIP4P/2005 water models, but was not observed for the flexible TIP3P water model. Although the rigidity of the water model appeared to be a possible reason for the observed anomalies, further simulations revealed that the discrepancy is related to the difference in the interfacial potential, pointing to the unsuitability of the LB mixing rules when calculating thermal conductivity with the Green–Kubo formulation. It was concluded that the interfacial potential parameters should be carefully optimized to correctly simulate heat transport for solid-liquid systems when using the Green-Kubo method.

The findings regarding the two main topics of the thesis, structural descriptors and machine learning potentials, were then presented in the following chapters. Be-

longing to the family of machine learning force-fields, high-dimensional neural network potentials have been found to be viable alternatives to electronic structure calculations by providing similar levels of accuracy at a lower computational cost. One crucial requirement for developing a robust neural network potential is a description of the local atomic neighborhood as a set of symmetrically-invariant real-valued numbers. Referred to as descriptors, different constructions have been proposed in the literature, but there is as yet no established canonical choice. Motivated by this immaturity, a new set of orthogonal descriptors is introduced in Chapter 3 that are invariant to the physical symmetries and can more efficiently represent structural environments than two of the frequent alternatives [42, 63].

The performance of the proposed descriptors in a neural network potential was compared to that of the Behler–Parinello descriptors and the SOAP descriptors, both commonly employed in machine learning potentials. For a given training set and comparable hyperparameters, our descriptors were found to give substantially smaller fitting errors than the Behler–Parinello descriptors, and similar fitting errors to the SOAP descriptors but at an order of magnitude lower computational cost. The superior performance of the proposed descriptors as compared to the Behler–Parinello descriptors is conjectured to be a consequence of the proposed descriptors deriving from a function expansion over orthogonal basis functions that efficiently encodes configurational information. As for the SOAP descriptors, the improved computational efficiency is a consequence of avoiding special function evaluations and numerical integration. The suitability of the proposed descriptors for machine learning potentials was verified by preliminary molecular dynamics simulations of solid-state silicon.

Finally in Chapter 4, a discontinuity was identified in the basis functions used to construct the recently presented Spherical Bessel descriptors [76] and the SOAP descriptors [63], and an updated version of the Spherical Bessel descriptors was introduced. Moreover, the Spherical Bessel descriptors were shown to satisfy a necessary condition for optimal completeness on the basis of the rank theorem [173], establishing their ability to encode all relevant physical information about a local atomic environment using the fewest possible descriptors. At present, the Spherical Bessel descriptors

are the only descriptors known to satisfy this condition. Moreover, they have been shown to be more than an order of magnitude faster to evaluate than the SOAP descriptors, and an optimized code to calculate the Spherical descriptors has been made available. The performance of an NNP for solid-state silicon using the Spherical Bessel descriptors was compared to that of NNPs using the prior version of the Spherical Bessel descriptors, the SOAP descriptors, and the Zernike descriptors. A detailed derivation of the final form of Spherical Bessel descriptors, of the derivatives of the Spherical Bessel descriptors and the SOAP descriptors with respect to the relative spherical coordinates of the surrounding atoms are provided in Appendix B.

5.2. Recommendations For Future Research

Once a robust description scheme is established and validated on a single-species system, the natural next step is to extend the formulation to multi-species systems as most of chemical phenomena occur in the presence of multiple chemical species. This problem has challenges due to the lack of a canonical descriptor set that is commonly-accepted by the MLP community. Another reason might be the fact that the complexity of MLPs is likely to increase nonlinearly (perhaps quadratically) with the number of chemical species in the system. This rapid growth in complexity is a result of the need to adjust the regressor architecture for each species, and to describe each distinct pairwise atomic interaction separately.

A standard approach for labelling species in MLPs is assigning unique weight factors ω^k to each atomic species k in the system. Bartók et al. [63] proposed to use weights in the neighbor densities to distinguish between species, but then only investigated single-species systems. In this thesis, the weight factor ω_j^k was also included in Equations 3.1 and 4.2 as the species-dependent weight for atom j, but then assumed to be one in both derivations for simplicity. Recently Artrith et al. [59] proposed a new approach that claimed to avoid quadratic scaling with the number of chemical species. They assigned weight factors to distinguish species following the Ising-model [181], and reported results with consant complexity for transition-metal oxide compositions and biomolecules up to 11 chemical species. The inclusion of zero as a weight was not

justified in the text though, and likely results in identical descriptions for different configurations. Rostami et al. [61] also claimed to obtain a linear scaling computational complexity in multicomponent systems, and proposed the so-called optimized symmetry functions, a modified version of the BP descriptors [42]. Although they reported results in agreement with some experimantal findings and DFT results for some ionic systems, the weighting scheme they used was based on the electronic charges of atoms (cation or anion) and therefore seemed an adhoc system-dependent solution rather than an extensible solution for all multicomponent systems.

The author of the thesis believes that there is not yet a robust weighting scheme that is mathematically justified, scales linearly and can be applied in all kinds of multicomponent chemical environments. Considering the superiority of the SB descriptors demonstrated in Chapters 3 and 4, their extension to multi-species systems with a robust weighting scheme appears to be the next step.

REFERENCES

1. Fermi, E., P. Pasta, S. Ulam and M. Tsingou, *Studies of the nonlinear problems*, Tech. rep., Los Alamos Scientific Lab., N. Mex., 1955.

2. Alder, B. J. and T. E. Wainwright, "Studies in molecular dynamics. I. General method", *The Journal of Chemical Physics*, Vol. 31, No. 2, pp. 459–466, 1959.

3. Rahman, A., "Correlations in the motion of atoms in liquid argon", *Physical Review*, Vol. 136, No. 2A, p. A405, 1964.

4. Rapaport, D. C. and D. C. R. Rapaport, *The art of molecular dynamics simulation*, Cambridge university press, 2004.

5. Haile, J. M., *Molecular dynamics simulation: elementary methods*, Vol. 1, Wiley New York, 1992.

6. van Gunsteren, W. F. and H. J. Berendsen, "Computer simulation of molecular dynamics: Methodology, applications, and perspectives in chemistry", *Angewandte Chemie International Edition in English*, Vol. 29, No. 9, pp. 992–1023, 1990.

7. Flory, P. J. and M. Volkenstein, "Statistical mechanics of chain molecules", , 1969.

8. Karplus, M. and G. A. Petsko, "Molecular dynamics simulations in biology", *Nature*, Vol. 347, No. 6294, p. 631, 1990.

9. Bao, G. and S. Suresh, "Cell and molecular mechanics of biological materials", *Nature materials*, Vol. 2, No. 11, p. 715, 2003.

10. Sarikaya, M., C. Tamerler, A. K.-Y. Jen, K. Schulten and F. Baneyx, "Molecular biomimetics: nanotechnology through biology", *Nature materials*, Vol. 2, No. 9,

p. 577, 2003.

11. Cheong, W. and L. Zhang, "Molecular dynamics simulation of phase transformations in silicon monocrystals due to nano-indentation", *Nanotechnology*, Vol. 11, No. 3, p. 173, 2000.

12. Curtin, W. A. and R. E. Miller, "Atomistic/continuum coupling in computational materials science", *Modelling and simulation in materials science and engineering*, Vol. 11, No. 3, p. R33, 2003.

13. Born, M. and R. Oppenheimer, "Zur quantentheorie der molekeln", *Annalen der Physik*, Vol. 389, No. 20, pp. 457–484, 1927.

14. Lennard-Jones, J. E., "On the determination of molecular fields. II. From the equation of state of gas", *Proc. Roy. Soc. A*, Vol. 106, pp. 463–477, 1924.

15. Tersoff, J., "New empirical approach for the structure and energy of covalent systems", *Physical Review B*, Vol. 37, No. 12, p. 6991, 1988.

16. Stillinger, F. H. and T. A. Weber, "Computer simulation of local order in condensed phases of silicon", *Physical review B*, Vol. 31, No. 8, p. 5262, 1985.

17. Van Duin, A. C., S. Dasgupta, F. Lorant and W. A. Goddard, "ReaxFF: a reactive force field for hydrocarbons", *The Journal of Physical Chemistry A*, Vol. 105, No. 41, pp. 9396–9409, 2001.

18. Daw, M. S. and M. I. Baskes, "Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals", *Physical Review B*, Vol. 29, No. 12, p. 6443, 1984.

19. Marx, D. and J. Hutter, *Ab initio molecular dynamics: basic theory and advanced methods*, Cambridge University Press, 2009.

20. Schrödinger, E., "An undulatory theory of the mechanics of atoms and molecules",

Physical review, Vol. 28, No. 6, p. 1049, 1926.

21. Yang, W. and P. W. Ayers, "Density-functional theory", *Computational Medicinal Chemistry for Drug Discovery*, pp. 103–132, CRC Press, 2003.

22. Kohn, W., A. D. Becke and R. G. Parr, "Density functional theory of electronic structure", *The Journal of Physical Chemistry*, Vol. 100, No. 31, pp. 12974–12980, 1996.

23. Jones, R. O., "Density functional theory: Its origins, rise to prominence, and future", *Reviews of modern physics*, Vol. 87, No. 3, p. 897, 2015.

24. Gross, E. K. and R. M. Dreizler, *Density functional theory*, Vol. 337, Springer Science & Business Media, 2013.

25. Parr, R. G., "Density functional theory", *Annual Review of Physical Chemistry*, Vol. 34, No. 1, pp. 631–656, 1983.

26. Ziegler, T., "Approximate density functional theory as a practical tool in molecular energetics and dynamics", *Chemical Reviews*, Vol. 91, No. 5, pp. 651–667, 1991.

27. Kresse, G. and J. Hafner, "Ab initio molecular-dynamics simulation of the liquid-metal–amorphous-semiconductor transition in germanium", *Physical Review B*, Vol. 49, No. 20, p. 14251, 1994.

28. Marzari, N., D. Vanderbilt and M. C. Payne, "Ensemble density-functional theory for ab initio molecular dynamics of metals and finite-temperature insulators", *Physical review letters*, Vol. 79, No. 7, p. 1337, 1997.

29. Fattebert, J.-L. and F. Gygi, "Density functional theory for efficient ab initio molecular dynamics simulations in solution", *Journal of computational chemistry*, Vol. 23, No. 6, pp. 662–666, 2002.

30. VandeVondele, J., F. Mohamed, M. Krack, J. Hutter, M. Sprik and M. Parrinello, "The influence of temperature and density functional models in ab initio molecular dynamics simulation of liquid water", *The Journal of chemical physics*, Vol. 122, No. 1, p. 014515, 2005.

31. Tse, J. S., "Ab initio molecular dynamics with density functional theory", *Annual review of physical chemistry*, Vol. 53, No. 1, pp. 249–290, 2002.

32. Kresse, G. and J. Hafner, "Ab initio molecular dynamics for open-shell transition metals", *Physical Review B*, Vol. 48, No. 17, p. 13115, 1993.

33. Leung, K. and J. L. Budzien, "Ab initio molecular dynamics simulations of the initial stages of solid–electrolyte interphase formation on lithium ion battery graphitic anodes", *Physical Chemistry Chemical Physics*, Vol. 12, No. 25, pp. 6583–6586, 2010.

34. Bishop, C. M., *Pattern recognition and machine learning*, springer, 2006.

35. Michalski, R. S., J. G. Carbonell and T. M. Mitchell, *Machine learning: An artificial intelligence approach*, Springer Science & Business Media, 2013.

36. Michie, D., D. J. Spiegelhalter, C. Taylor *et al.*, "Machine learning", *Neural and Statistical Classification*, Vol. 13, 1994.

37. Hinton, G., L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups", *IEEE Signal processing magazine*, Vol. 29, No. 6, pp. 82–97, 2012.

38. Sonka, M., V. Hlavac and R. Boyle, *Image processing, analysis, and machine vision*, Cengage Learning, 2014.

39. Holden, M. K., "Virtual environments for motor rehabilitation", *Cyberpsychology*

& behavior, Vol. 8, No. 3, pp. 187–211, 2005.

40. Russell, S. J. and P. Norvig, *Artificial intelligence: a modern approach*, Malaysia; Pearson Education Limited,, 2016.

41. Behler, J., "Atom-centered symmetry functions for constructing high-dimensional neural network potentials", *The Journal of chemical physics*, Vol. 134, No. 7, p. 074106, 2011.

42. Behler, J. and M. Parrinello, "Generalized neural-network representation of high-dimensional potential-energy surfaces", *Physical review letters*, Vol. 98, No. 14, p. 146401, 2007.

43. Bartók, A. P., M. C. Payne, R. Kondor and G. Csányi, "Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons", *Physical review letters*, Vol. 104, No. 13, p. 136403, 2010.

44. Artrith, N. and J. Behler, "High-dimensional neural network potentials for metal surfaces: A prototype study for copper", *Physical Review B*, Vol. 85, No. 4, p. 045439, 2012.

45. Botu, V., R. Batra, J. Chapman and R. Ramprasad, "Machine learning force fields: construction, validation, and outlook", *The Journal of Physical Chemistry C*, Vol. 121, No. 1, pp. 511–522, 2016.

46. Cooper, A. M., P. P. Hallmen and J. Kästner, "Potential energy surface interpolation with neural networks for instanton rate calculations", *The Journal of Chemical Physics*, Vol. 148, No. 9, p. 094106, 2018.

47. Deringer, V. L. and G. Csányi, "Machine learning based interatomic potential for amorphous carbon", *Physical Review B*, Vol. 95, No. 9, p. 094203, 2017.

48. Devillers, J. and A. T. Balaban, *Topological indices and related descriptors in*

QSAR and QSPAR, CRC Press, 2000.

49. Valle, M. and A. R. Oganov, "Crystal fingerprint space–a novel paradigm for studying crystal-structure sets", *Acta Crystallographica Section A: Foundations of Crystallography*, Vol. 66, No. 5, pp. 507–517, 2010.

50. Todeschini, R. and V. Consonni, *Handbook of molecular descriptors*, Vol. 11, John Wiley & Sons, 2008.

51. Jacob, L. and J.-P. Vert, "Protein-ligand interaction prediction: an improved chemogenomics approach", *Bioinformatics*, Vol. 24, No. 19, pp. 2149–2156, 2008.

52. Consonni, V., R. Todeschini and M. Pavan, "Structure/response correlations and similarity/diversity analysis by GETAWAY descriptors. 1. Theory of the novel 3D molecular descriptors", *Journal of Chemical Information and Computer Sciences*, Vol. 42, No. 3, pp. 682–692, 2002.

53. Grisoni, F., V. Consonni, M. Vighi, S. Villa and R. Todeschini, "Expert QSAR system for predicting the bioconcentration factor under the REACH regulation", *Environmental research*, Vol. 148, pp. 507–512, 2016.

54. Hobday, S., R. Smith and J. Belbruno, "Applications of neural networks to fitting interatomic potential functions", *Modelling and Simulation in Materials Science and Engineering*, Vol. 7, No. 3, p. 397, 1999.

55. Behler, J., R. Martoňák, D. Donadio and M. Parrinello, "Pressure-induced phase transitions in silicon studied by neural network-based metadynamics simulations", *physica status solidi (b)*, Vol. 245, No. 12, pp. 2618–2629, 2008.

56. Handley, C. M. and P. L. Popelier, "Dynamically polarizable water potential based on multipole moments trained by machine learning", *Journal of chemical theory and computation*, Vol. 5, No. 6, pp. 1474–1489, 2009.

57. Gastegger, M., L. Schwiedrzik, M. Bittermann, F. Berzsenyi and P. Marquetand, "wACSF—Weighted atom-centered symmetry functions as descriptors in machine learning potentials", *The Journal of chemical physics*, Vol. 148, No. 24, p. 241709, 2018.

58. Singraber, A., J. Behler and C. Dellago, "Library-Based LAMMPS Implementation of High-Dimensional Neural Network Potentials", *Journal of chemical theory and computation*, Vol. 15, No. 3, pp. 1827–1840, 2019.

59. Artrith, N., A. Urban and G. Ceder, "Efficient and accurate machine-learning interpolation of atomic energies in compositions with many species", *Physical Review B*, Vol. 96, No. 1, p. 014112, 2017.

60. Liu, Q., X. Zhou, L. Zhou, Y. Zhang, X. Luo, H. Guo and B. Jiang, "Constructing high-dimensional neural network potential energy surfaces for gas–surface scattering and reactions", *The Journal of Physical Chemistry C*, Vol. 122, No. 3, pp. 1761–1769, 2018.

61. Rostami, S., M. Amsler and S. A. Ghasemi, "Optimized symmetry functions for machine-learning interatomic potentials of multicomponent systems", *The Journal of chemical physics*, Vol. 149, No. 12, p. 124106, 2018.

62. Rasmussen, C. E., "Gaussian processes in machine learning", *Summer School on Machine Learning*, pp. 63–71, Springer, 2003.

63. Bartók, A. P., R. Kondor and G. Csányi, "On representing chemical environments", *Physical Review B*, Vol. 87, No. 18, p. 184115, 2013.

64. Bartók, A. P., J. Kermode, N. Bernstein and G. Csányi, "Machine learning a general-purpose interatomic potential for silicon", *Physical Review X*, Vol. 8, No. 4, p. 041048, 2018.

65. De, S., A. P. Bartók, G. Csányi and M. Ceriotti, "Comparing molecules and solids

across structural and alchemical space", *Physical Chemistry Chemical Physics*, Vol. 18, No. 20, pp. 13754–13769, 2016.

66. Ghasemi, S. A., A. Hofstetter, S. Saha and S. Goedecker, "Interatomic potentials for ionic systems with density functional accuracy based on charge densities obtained by a neural network", *Physical Review B*, Vol. 92, No. 4, p. 045131, 2015.

67. Ceriotti, M., M. J. Willatt and G. Csányi, "Machine Learning of Atomic-Scale Properties Based on Physical Principles", *Handbook of Materials Modeling: Methods: Theory and Modeling*, pp. 1–27, 2018.

68. Rupp, M., A. Tkatchenko, K.-R. Müller and O. A. Von Lilienfeld, "Fast and accurate modeling of molecular atomization energies with machine learning", *Physical review letters*, Vol. 108, No. 5, p. 058301, 2012.

69. Hastie, T., R. Tibshirani, J. Friedman and J. Franklin, "The elements of statistical learning: data mining, inference and prediction", *The Mathematical Intelligencer*, Vol. 27, No. 2, pp. 83–85, 2005.

70. Chmiela, S., A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt and K.-R. Müller, "Machine learning of accurate energy-conserving molecular force fields", *Science advances*, Vol. 3, No. 5, p. e1603015, 2017.

71. Chmiela, S., H. E. Sauceda, K.-R. Müller and A. Tkatchenko, "Towards exact molecular dynamics simulations with machine-learned force fields", *Nature communications*, Vol. 9, No. 1, p. 3887, 2018.

72. Shapeev, A. V., "Moment tensor potentials: A class of systematically improvable interatomic potentials", *Multiscale Modeling & Simulation*, Vol. 14, No. 3, pp. 1153–1173, 2016.

73. Müller, C., *Spherical harmonics*, Vol. 17, Springer, 2006.

74. Andrews, L. C. and L. C. Andrews, *Special functions of mathematics for engineers*, McGraw-Hill New York, 1992.

75. Akiner, T., E. Kocer, J. K. Mason and H. Erturk, "Green–Kubo assessments of thermal transport in nanocolloids based on interfacial effects", *Materials Today Communications*, p. 100533, 2019.

76. Kocer, E., J. K. Mason and H. Erturk, "A novel approach to describe chemical environments in high-dimensional neural network potentials", *The Journal of Chemical Physics*, Vol. 150, No. 15, p. 154102, 2019.

77. Kocer, E., J. K. Mason and H. Erturk, "Continuous and optimally complete description of chemical environments using Spherical Bessel descriptors", *AIP Advances*, Vol. 10, No. 1, p. 015021, 2020.

78. Russel, W. B., W. Russel, D. A. Saville and W. R. Schowalter, *Colloidal dispersions*, Cambridge university press, 1991.

79. Nguyen, A. and H. J. Schulze, *Colloidal science of flotation*, CRC Press, 2003.

80. Tadros, T. F., *Handbook of Colloid and Interface Science: Industrial Applications*, Vol. 3, Walter de Gruyter GmbH & Co KG, 2017.

81. Prasher, R., P. Bhattacharya and P. E. Phelan, "Thermal conductivity of nanoscale colloidal solutions (nanofluids)", *Physical review letters*, Vol. 94, No. 2, p. 025901, 2005.

82. Müller-Plathe, F., "A simple nonequilibrium molecular dynamics method for calculating the thermal conductivity", *The Journal of chemical physics*, Vol. 106, No. 14, pp. 6082–6085, 1997.

83. Kubo, R., "The fluctuation-dissipation theorem", *Reports on progress in physics*, Vol. 29, No. 1, p. 255, 1966.

84. Schelling, P. K., S. R. Phillpot and P. Keblinski, "Comparison of atomic-level simulation methods for computing thermal conductivity", *Physical Review B*, Vol. 65, No. 14, p. 144306, 2002.

85. Vogelsang, R., C. Hoheisel and G. Ciccotti, "Thermal conductivity of the Lennard-Jones liquid by molecular dynamics calculations", *The Journal of chemical physics*, Vol. 86, No. 11, pp. 6371–6375, 1987.

86. Bresme, F., J. Biddle, J. Sengers and M. Anisimov, "Communication: Minimum in the thermal conductivity of supercooled water: A computer simulation study", , 2014.

87. Sirk, T. W., S. Moore and E. F. Brown, "Characteristics of thermal conductivity in classical water models", *The Journal of chemical physics*, Vol. 138, No. 6, p. 064505, 2013.

88. English, N. J. and J. S. Tse, "Thermal Conductivity of Supercooled Water: An Equilibrium Molecular Dynamics Exploration", *The journal of physical chemistry letters*, Vol. 5, No. 21, pp. 3819–3824, 2014.

89. Rosenbaum, E. J., N. J. English, J. K. Johnson, D. W. Shaw and R. P. Warzinski, "Thermal conductivity of methane hydrate from experiment and molecular simulation", *The Journal of Physical Chemistry B*, Vol. 111, No. 46, pp. 13194–13205, 2007.

90. Li, L., Y. Zhang, H. Ma and M. Yang, "Molecular dynamics simulation of effect of liquid layering around the nanoparticle on the enhanced thermal conductivity of nanofluids", *Journal of nanoparticle research*, Vol. 12, No. 3, pp. 811–821, 2010.

91. Teng, K.-L., P.-Y. Hsiao, S.-W. Hung, C.-C. Chieng, M.-S. Liu and M.-C. Lu, "Enhanced thermal conductivity of nanofluids diagnosis by molecular dynamics simulations", *Journal of nanoscience and nanotechnology*, Vol. 8, No. 7, pp. 3710–3718, 2008.

92. Sarkar, S. and R. P. Selvam, "Molecular dynamics simulation of effective thermal conductivity and study of enhanced thermal transport mechanism in nanofluids", *Journal of applied physics*, Vol. 102, No. 7, p. 074302, 2007.

93. Sankar, N., N. Mathew and C. Sobhan, "Molecular dynamics modeling of thermal conductivity enhancement in metal nanoparticle suspensions", *International Communications in Heat and Mass Transfer*, Vol. 35, No. 7, pp. 867–872, 2008.

94. Li, J., L. Porter and S. Yip, "Atomistic modeling of finite-temperature properties of crystalline β-SiC: II. Thermal conductivity and effects of point defects", *Journal of Nuclear Materials*, Vol. 255, No. 2-3, pp. 139–152, 1998.

95. Chen, J., G. Zhang and B. Li, "How to improve the accuracy of equilibrium molecular dynamics for computation of thermal conductivity?", *Physics Letters A*, Vol. 374, No. 23, pp. 2392–2396, 2010.

96. McGaughey, A. J. and M. Kaviany, "Phonon transport in molecular dynamics simulations: formulation and thermal conductivity prediction", *Advances in heat transfer*, Vol. 39, pp. 169–255, 2006.

97. Jones, R. E. and K. K. Mandadapu, "Adaptive Green-Kubo estimates of transport coefficients from molecular dynamics based on robust error analysis", *The Journal of chemical physics*, Vol. 136, No. 15, p. 154102, 2012.

98. de Sousa Oliveira, L. and P. A. Greaney, "Method to manage integration error in the Green-Kubo method", *Physical Review E*, Vol. 95, No. 2, p. 023308, 2017.

99. Ercole, L., A. Marcolongo and S. Baroni, "Accurate thermal conductivities from optimally short molecular dynamics simulations", *Scientific reports*, Vol. 7, No. 1, p. 15835, 2017.

100. Maxwell, J. C., W. Garnett and P. Pesic, *An elementary treatise on electricity*, Courier Corporation, 2005.

101. Keblinski, P., S. Phillpot, S. Choi and J. Eastman, "Mechanisms of heat flow in suspensions of nano-sized particles (nanofluids)", *International journal of heat and mass transfer*, Vol. 45, No. 4, pp. 855–863, 2002.

102. Muraleedharan, M. G., D. S. Sundaram, A. Henry and V. Yang, "Thermal conductivity calculation of nano-suspensions using Green–Kubo relations with reduced artificial correlations", *Journal of Physics: Condensed Matter*, Vol. 29, No. 15, p. 155302, 2017.

103. Sachdeva, P. and R. Kumar, "Effect of hydration layer and surface wettability in enhancing thermal conductivity of nanofluids", *Applied Physics Letters*, Vol. 95, No. 22, p. 223105, 2009.

104. Kang, H., Y. Zhang and M. Yang, "Molecular dynamics simulation of thermal conductivity of Cu–Ar nanofluid using EAM potential for Cu–Cu interactions", *Applied Physics A*, Vol. 103, No. 4, p. 1001, 2011.

105. Babaei, H., P. Keblinski and J. Khodadadi, "A proof for insignificant effect of Brownian motion-induced micro-convection on thermal conductivity of nanofluids by utilizing molecular dynamics simulations", *Journal of Applied Physics*, Vol. 113, No. 8, p. 084302, 2013.

106. Akıner, T., H. Ertürk and K. Atalık, "Prediction of Thermal Conductivity and Shear Viscosity of Water-Cu Nanofluids Using Equilibrium Molecular Dynamics", *ASME 2013 International Mechanical Engineering Congress and Exposition*, pp. V08CT09A012–V08CT09A012, American Society of Mechanical Engineers, 2013.

107. Yu, W. and S. Choi, "The role of interfacial layers in the enhanced thermal conductivity of nanofluids: a renovated Maxwell model", *Journal of nanoparticle research*, Vol. 5, No. 1-2, pp. 167–171, 2003.

108. Feng, Y., B. Yu, P. Xu and M. Zou, "The effective thermal conductivity of nanofluids based on the nanolayer and the aggregation of nanoparticles", *Journal of*

Physics D: Applied Physics, Vol. 40, No. 10, p. 3164, 2007.

109. Yu, C.-J., A. Richter, A. Datta, M. Durbin and P. Dutta, "Molecular layering in a liquid on a solid substrate: an X-ray reflectivity study", *Physica B: Condensed Matter*, Vol. 283, No. 1-3, pp. 27–31, 2000.

110. Akıner, T., J. K. Mason and H. Ertürk, "Nanolayering around and thermal resistivity of the water-hexagonal boron nitride interface", *The Journal of Chemical Physics*, Vol. 147, No. 4, p. 044709, 2017.

111. McQuarrie, D., "Statistical mechanics. 2000", *Sausalito, Calif.: University Science Books*, Vol. 12, p. 641, 2004.

112. Babaei, H., P. Keblinski and J. M. Khodadadi, "Equilibrium molecular dynamics determination of thermal conductivity for multi-component systems", *Journal of Applied Physics*, Vol. 112, No. 5, p. 054310, 2012.

113. Jorgensen, W. L., J. Chandrasekhar, J. D. Madura, R. W. Impey and M. L. Klein, "Comparison of simple potential functions for simulating liquid water", *The Journal of chemical physics*, Vol. 79, No. 2, pp. 926–935, 1983.

114. Abascal, J. L. and C. Vega, "A general purpose model for the condensed phases of water: TIP4P/2005", *The Journal of chemical physics*, Vol. 123, No. 23, p. 234505, 2005.

115. Berendsen, H., J. Grigera and T. Straatsma, "The missing term in effective pair potentials", *Journal of Physical Chemistry*, Vol. 91, No. 24, pp. 6269–6271, 1987.

116. Ryckaert, J.-P., G. Ciccotti and H. J. Berendsen, "Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes", *Journal of Computational Physics*, Vol. 23, No. 3, pp. 327–341, 1977.

117. Mark, P. and L. Nilsson, "Structure and dynamics of the TIP3P, SPC, and SPC/E water models at 298 K", *The Journal of Physical Chemistry A*, Vol. 105, No. 43, pp. 9954–9960, 2001.

118. Sanders, D. E. and A. E. DePristo, "Metal/metal homo-epitaxy on fcc (001) surfaces: Is there transient mobility of adsorbed atoms?", *Surface science*, Vol. 254, No. 1-3, pp. 341–353, 1991.

119. Plimpton, S., "Fast parallel algorithms for short-range molecular dynamics", *Journal of computational physics*, Vol. 117, No. 1, pp. 1–19, 1995.

120. Humphrey, W., A. Dalke and K. Schulten, "VMD: visual molecular dynamics", *Journal of molecular graphics*, Vol. 14, No. 1, pp. 33–38, 1996.

121. Khadem, M. H. and A. P. Wemhoff, "Comparison of Green–Kubo and NEMD heat flux formulations for thermal conductivity prediction using the Tersoff potential", *Computational Materials Science*, Vol. 69, pp. 428–434, 2013.

122. Turney, J., E. Landry, A. McGaughey and C. Amon, "Predicting phonon properties and thermal conductivity from anharmonic lattice dynamics calculations and molecular dynamics simulations", *Physical Review B*, Vol. 79, No. 6, p. 064301, 2009.

123. Mao, Y. and Y. Zhang, "Thermal conductivity, shear viscosity and specific heat of rigid water models", *Chemical Physics Letters*, Vol. 542, pp. 37–41, 2012.

124. Xuan, Y. and Q. Li, "Heat transfer enhancement of nanofluids", *International Journal of heat and fluid flow*, Vol. 21, No. 1, pp. 58–64, 2000.

125. Matsumoto, S. M. T. K. S. and Y. Y. T. Kimura, "Liquid droplet in contact with a solid surface", *Microscale Thermophysical Engineering*, Vol. 2, No. 1, pp. 49–62, 1998.

126. Leroy, F. and F. Müller-Plathe, "Solid-liquid surface free energy of Lennard-Jones liquid on smooth and rough surfaces computed by molecular dynamics using the phantom-wall method", *The Journal of chemical physics*, Vol. 133, No. 4, p. 044110, 2010.

127. Leroy, F., S. Liu and J. Zhang, "Parametrizing nonbonded interactions from wetting experiments via the work of adhesion: Example of water on graphene surfaces", *The Journal of Physical Chemistry C*, Vol. 119, No. 51, pp. 28470–28481, 2015.

128. Akıner, T., J. Mason and H. Ertürk, "Thermal characterization assessment of rigid and flexible water models in a nanogap using molecular dynamics", *Chemical Physics Letters*, Vol. 687, pp. 270–275, 2017.

129. Hautier, G., A. Jain and S. P. Ong, "From the computer to the laboratory: materials discovery and design using first-principles calculations", *Journal of Materials Science*, Vol. 47, No. 21, pp. 7317–7340, 2012.

130. Hill, J.-R., C. M. Freeman and L. Subramanian, "Use of force fields in materials modeling", *Reviews in computational chemistry*, Vol. 16, pp. 141–216, 2000.

131. Burke, K., "Perspective on density functional theory", *The Journal of chemical physics*, Vol. 136, No. 15, p. 150901, 2012.

132. Binks, D. J. and R. W. Grimes, "Incorporation of monovalent ions in ZnO and their influence on varistor degradation", *Journal of the American Ceramic Society*, Vol. 76, No. 9, pp. 2370–2372, 1993.

133. Süle, P. and M. Szendrő, "The classical molecular dynamics simulation of graphene on Ru (0001) using a fitted Tersoff interface potential", *Surface and Interface Analysis*, Vol. 46, No. 1, pp. 42–47, 2014.

134. Li, Z., J. R. Kermode and A. De Vita, "Molecular dynamics with on-the-fly

machine learning of quantum-mechanical forces", *Physical review letters*, Vol. 114, No. 9, p. 096405, 2015.

135. Bose, S., D. Dhawan, S. Nandi, R. R. Sarkar and D. Ghosh, "Machine learning prediction of interaction energies in rigid water clusters", *Physical Chemistry Chemical Physics*, Vol. 20, No. 35, pp. 22987–22996, 2018.

136. Snyder, J. C., M. Rupp, K. Hansen, K.-R. Müller and K. Burke, "Finding density functionals with machine learning", *Physical review letters*, Vol. 108, No. 25, p. 253002, 2012.

137. Behler, J., "Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations", *Physical Chemistry Chemical Physics*, Vol. 13, No. 40, pp. 17930–17955, 2011.

138. Smith, J. S., O. Isayev and A. E. Roitberg, "ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost", *Chemical science*, Vol. 8, No. 4, pp. 3192–3203, 2017.

139. Pukrittayakamee, A., M. Malshe, M. Hagan, L. Raff, R. Narulkar, S. Bukkapatnum and R. Komanduri, "Simultaneous fitting of a potential-energy surface and its corresponding force fields using feedforward neural networks", *The Journal of chemical physics*, Vol. 130, No. 13, p. 134101, 2009.

140. Raff, L., M. Malshe, M. Hagan, D. Doughan, M. Rockley and R. Komanduri, "Ab initio potential-energy surfaces for complex, multichannel systems using modified novelty sampling and feedforward neural networks", *The Journal of chemical physics*, Vol. 122, No. 8, p. 084104, 2005.

141. Behler, J., "Perspective: Machine learning potentials for atomistic simulations", *The Journal of chemical physics*, Vol. 145, No. 17, p. 170901, 2016.

142. Morawietz, T., A. Singraber, C. Dellago and J. Behler, "How van der Waals in-

teractions determine the unique properties of water", *Proceedings of the National Academy of Sciences*, Vol. 113, No. 30, pp. 8368–8373, 2016.

143. Natarajan, S. K. and J. Behler, "Neural network molecular dynamics simulations of solid–liquid interfaces: Water at low-index copper surfaces", *Physical Chemistry Chemical Physics*, Vol. 18, No. 41, pp. 28704–28725, 2016.

144. Dragoni, D., 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*, Vol. 2, No. 1, p. 013808, 2018.

145. Rosenbrock, C. W., E. R. Homer, G. Csányi and G. L. Hart, "Discovering the Building Blocks of Atomic Systems using Machine Learning", *arXiv preprint arXiv:1703.06236*, 2017.

146. Chollet, F., "Keras", `https://github.com/fchollet/keras`, 2015.

147. Abadi, M., P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: a system for large-scale machine learning.", *OSDI*, Vol. 16, pp. 265–283, 2016.

148. Szlachta, W. J., A. P. Bartók and G. Csányi, "Accuracy and transferability of Gaussian approximation potential models for tungsten", *Physical Review B*, Vol. 90, No. 10, p. 104108, 2014.

149. Zhou, X. and R. Jones, "Effects of cutoff functions of Tersoff potentials on molecular dynamics simulations of thermal transport", *Modelling and Simulation in Materials Science and Engineering*, Vol. 19, No. 2, p. 025004, 2011.

150. Yegnanarayana, B., *Artificial neural networks*, PHI Learning Pvt. Ltd., 2009.

151. LeCun, Y., Y. Bengio and G. Hinton, "Deep learning", *nature*, Vol. 521, No. 7553, p. 436, 2015.

152. Csáji, B. C., *Approximation with artificial neural networks*, Master's Thesis, Etvs Lornd University, Hungary, 2001.

153. Bholoa, A., S. D. Kenny and R. Smith, "A new approach to potential fitting using neural networks", *Nuclear instruments and methods in physics research section B: Beam interactions with materials and atoms*, Vol. 255, No. 1, pp. 1–7, 2007.

154. Lorenz, S., A. Groß and M. Scheffler, "Representing high-dimensional potential-energy surfaces for reactions at surfaces by neural networks", *Chemical Physics Letters*, Vol. 395, No. 4-6, pp. 210–215, 2004.

155. Blank, T. B., S. D. Brown, A. W. Calhoun and D. J. Doren, "Neural network models of potential energy surfaces", *The Journal of chemical physics*, Vol. 103, No. 10, pp. 4129–4137, 1995.

156. Prudente, F. V., P. H. Acioli and J. S. Neto, "The fitting of potential energy surfaces using neural networks: Application to the study of vibrational levels of H 3+", *The Journal of chemical physics*, Vol. 109, No. 20, pp. 8801–8808, 1998.

157. Bittencourt, A. C. P., F. V. Prudente and J. D. M. Vianna, "The fitting of potential energy and transition moment functions using neural networks: transition probabilities in OH (A2Σ+→ X2Π)", *Chemical physics*, Vol. 297, No. 1-3, pp. 153–161, 2004.

158. Brown, D. F., M. N. Gibbs and D. C. Clary, "Combining ab initio computations, neural networks, and diffusion Monte Carlo: An efficient method to treat weakly bound molecules", *The Journal of chemical physics*, Vol. 105, No. 17, pp. 7597–7604, 1996.

159. Bebis, G. and M. Georgiopoulos, "Feed-forward neural networks", *IEEE Potentials*, Vol. 13, No. 4, pp. 27–31, 1994.

160. Hochreiter, S., "The vanishing gradient problem during learning recurrent neural

nets and problem solutions", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 6, No. 02, pp. 107–116, 1998.

161. Rumelhart, D. E., G. E. Hinton and R. J. Williams, "Learning representations by back-propagating errors", *nature*, Vol. 323, No. 6088, p. 533, 1986.

162. Robbins, H. and S. Monro, "A stochastic approximation method", *Herbert Robbins Selected Papers*, pp. 102–109, Springer, 1985.

163. Stende, J.-A., *Constructing high-dimensional neural network potentials for molecular dynamics*, Master's Thesis, University of Oslo, Norway, 2017.

164. Glorot, X. and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010.

165. Balabin, R. M. and E. I. Lomakina, "Support vector machine regression (SVR/LS-SVM)–an alternative to neural networks (ANN) for analytical chemistry? Comparison of nonlinear methods on near infrared (NIR) spectroscopy data", *Analyst*, Vol. 136, No. 8, pp. 1703–1712, 2011.

166. Faber, F. A., L. Hutchison, B. Huang, J. Gilmer, S. S. Schoenholz, G. E. Dahl, O. Vinyals, S. Kearnes, P. F. Riley and O. A. Von Lilienfeld, "Prediction errors of molecular machine learning models lower than hybrid DFT error", *Journal of Chemical Theory and Computation*, Vol. 13, No. 11, pp. 5255–5264, 2017.

167. Novotni, M. and R. Klein, "3D Zernike descriptors for content based shape retrieval", *Proceedings of the* 8^{th} *ACM Symposium on Solid Modeling and Applications*, pp. 216–225, ACM, 2003.

168. Glielmo, A., P. Sollich and A. De Vita, "Accurate interatomic force fields via machine learning with covariant kernels", *Physical Review B*, Vol. 95, No. 21, p. 214302, 2017.

169. Huo, H. and M. Rupp, "Unified representation of molecules and crystals for machine learning", *arXiv preprint*, , No. 1704.06439, 2017.

170. Rowe, P., G. Csányi, D. Alfè and A. Michaelides, "Development of a machine learning potential for graphene", *Physical Review B*, Vol. 97, No. 5, p. 054303, 2018.

171. Artrith, N. and A. Urban, "An implementation of artificial neural-network potentials for atomistic materials simulations: Performance for TiO2", *Computational Materials Science*, Vol. 114, pp. 135–150, 2016.

172. Khorshidi, A. and A. A. Peterson, "Amp: A modular approach to machine learning in atomistic simulations", *Computer Physics Communications*, Vol. 207, pp. 310–324, 2016.

173. Krantz, S. G. and H. R. Parks, *The Implicit Function Theorem: History, Theory, and Applications*, Springer Science & Business Media, 2012.

174. Arfken, G., "Mathematical Methods for Physicists 3^{rd} Edition", , 1985.

175. Takahasi, H. and M. Mori, "Double exponential formulas for numerical integration", *Publications of the Research Institute for Mathematical Sciences*, Vol. 9, No. 3, pp. 721–741, 1974.

176. Caro, M. A., "Optimizing many-body atomic descriptors for enhanced computational performance of machine-learning-based interatomic potentials", *arXiv preprint*, , No. 1905.02142, 2019.

177. Canterakis, N., "3D Zernike moments and Zernike affine invariants for 3D image analysis and recognition", *In 11th Scandinavian Conf. on Image Analysis*, Citeseer, 1999.

178. Wang, Q., O. Ronneberger and H. Burkhardt, "Rotational invariance based on

Fourier analysis in polar and spherical coordinates", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31, No. 9, pp. 1715–1722, 2009.

179. Arvacheh, E. and H. Tizhoosh, "Pattern analysis using Zernike moments", *2005 IEEE Instrumentationand Measurement Technology Conference Proceedings*, Vol. 2, pp. 1574–1578, IEEE, 2005.

180. Imbalzano, G., A. Anelli, D. Giofré, S. Klees, J. Behler and M. Ceriotti, "Automatic selection of atomic fingerprints and reference configurations for machine-learning potentials", *The Journal of Chemical Physics*, Vol. 148, No. 24, p. 241730, 2018.

181. Sanchez, J. M., F. Ducastelle and D. Gratias, "Generalized cluster description of multicomponent systems", *Physica A: Statistical Mechanics and its Applications*, Vol. 128, No. 1-2, pp. 334–350, 1984.

APPENDIX A: CHAPTER 3

Sections A.1 and A.2 contain the derivation of the proposed radial basis functions in Chapter 3 and the associated force calculation procedure, respectively.

A.1. Derivation of the Radial Basis Functions

This section derives the proposed radial basis functions in Chapter 3 and provides all the required formula. Let $f_{nl}(r)$ be the linear combination of spherical Bessel functions

$$f_{nl}(r) = a_{nl} j_l\left(r\frac{u_{ln}}{r_c}\right) + b_{nl} j_l\left(r\frac{u_{l,n+1}}{r_c}\right) \tag{A.1}$$

where a_{nl} and b_{nl} are constants, $j_l(r)$ is the lth spherical Bessel function of the first kind, u_{ln} is the $(n+1)$th nonzero root of $j_l(r)$, and r_c is the cutoff radius. Since $f_{nl}(r_c) = 0$ by definition, the objective is to find a_{nl} and b_{nl} such that $f'_{nl}(r_c) = 0$ and $f''_{nl}(r_c) = 0$. Using the two differentiation rules for spherical Bessel functions [172]

$$j'_l(x) = j_{l-1}(x) - \frac{l+1}{x} j_l(x) \tag{A.2}$$

$$j'_l(x) = \frac{l}{x} j_l(x) - j_{l+1}(x), \tag{A.3}$$

the first and second derivatives of $f_{nl}(r)$ can be shown to vanish at $r = r_c$ if the coefficients in Equation A.1 satisfy

$$a_{nl} = \frac{u_{l,n+1}}{j_{l+1}(u_{ln})} c_{nl} \tag{A.4}$$

$$b_{nl} = -\frac{u_{ln}}{j_{l+1}(u_{l,n+1})} c_{nl} \tag{A.5}$$

for an arbitrary multiplicative constant c_{nl}. The value of c_{nl} is fixed by requiring that $f_{nl}(r)$ be appropriately normalized, or

$$\int_0^{r_c} f_{nl}(r) f_{nl}(r) r^2 \, \mathrm{d}r = 1. \tag{A.6}$$

This leads to

$$f_{nl}(r) = \left(\frac{1}{r_c^3} \frac{2}{u_{ln}^2 + u_{l,n+1}^2} \right)^{1/2} \left[\frac{u_{l,n+1}}{j_{l+1}(u_{ln})} j_l \left(r \frac{u_{ln}}{r_c} \right) - \frac{u_{ln}}{j_{l+1}(u_{l,n+1})} j_l \left(r \frac{u_{l,n+1}}{r_c} \right) \right] \tag{A.7}$$

as an explicit equation for the $f_{nl}(r)$.

Let the $g_{nl}(r)$ be a set of orthonormal radial basis functions with respect to the index n, derived by applying the Gram-Schmidt procedure to the $f_{nl}(r)$. For reference, the orthogonality relation for the spherical Bessel functions is [172]

$$\int_0^1 x^2 j_l(x u_{ln'}) j_l(x u_{ln}) \, \mathrm{d}x = \frac{\delta_{n'n}}{2} \left[j_{l+1}(u_{ln}) \right]^2 \tag{A.8}$$

where $\delta_{n'n}$ is the Kronecker delta. Using the substitution $x = r/r_c$, this is rewritten as

$$\int_0^{r_c} j_l \left(\frac{r}{r_c} u_{ln'} \right) j_l \left(\frac{r}{r_c} u_{ln} \right) r^2 \, \mathrm{d}r = \delta_{n'n} \frac{r_c^3}{2} \left[j_{l+1}(u_{ln}) \right]^2 . \tag{A.9}$$

For any integer $l \geq 0$, let $g_{ll}(r) = f_{ll}(r)$. The orthogonalization procedure involves constructing $g_{nl}(r)$ for $n > l$ given $f_{nl}(r)$ and $g_{ml}(r)$ for $l \leq m \leq n-1$. Notice that $g_{ml}(r)$ contains components of all $f_{pl}(r)$ for $l \leq p \leq m$, and therefore terms involving $j_l(\frac{r}{r_c} u_{lq})$ for all $l \leq q \leq m+1$. Since $f_{nl}(r)$ only contains terms involving $j_l(\frac{r}{r_c} u_{ln})$ and $j_l(\frac{r}{r_c} u_{l,n+1})$, $f_{nl}(r)$ is already orthogonal to all $g_{ml}(r)$ for $l \leq m \leq n-2$ by the orthogonality of the spherical Bessel functions. The only remaining step is to subtract the projection of $f_{nl}(r)$ onto $g_{n-1,l}(r)$, or

$$h_{nl}(r) = f_{nl}(r) - g_{n-1,l}(r) \int_0^{r_c} f_{nl}(r) g_{n-1,l}(r) r^2 \, \mathrm{d}r. \tag{A.10}$$

$h_{nl}(r)$ is orthogonal to all $g_{ml}(r)$ for $l \leq m \leq n-1$, but is not yet normalized. Let

$$d_{nl} = \int_0^{r_c} h_{nl}(r) h_{nl}(r) r^2 \, \mathrm{d}r \tag{A.11}$$

be the squared magnitude of $h_{nl}(r)$. Then the desired orthonormal $g_{nl}(r)$ is

$$g_{nl}(r) = d_{nl}^{-1/2} h_{nl}(r). \tag{A.12}$$

Explicitly evaluating the integral

$$\int_0^{r_c} f_{nl}(r) g_{n-1,l}(r) r^2 \, \mathrm{d}r = \frac{a_{nl} b_{n-1,l}}{\sqrt{d_{n-1,l}}} \frac{r_c^3}{2} \left[j_{l+1}(u_{ln}) \right]^2 \tag{A.13}$$

allows Equations A.10, A.11 and A.12 to be solved to derive the recursion relations

$$d_{nl} = 1 - \frac{e_{nl}}{d_{n-1,l}} \tag{A.14}$$

$$g_{nl}(r) = \frac{1}{\sqrt{d_{nl}}} \left[f_{nl}(r) + \sqrt{\frac{e_{nl}}{d_{n-1,l}}} g_{n-1,l}(r) \right] \tag{A.15}$$

where the constants e_{nl} are defined as

$$e_{nl} = \frac{u_{l,n-1}^2 u_{n+1,l}^2}{(u_{l,n-1}^2 + u_{ln}^2)(u_{ln}^2 + u_{l,n+1}^2)}. \tag{A.16}$$

These recursion relations can be initialized with $d_{ll} = 1$ and $g_{ll}(r) = f_{ll}(r)$ for any $0 \leq l$.

When $l = 0$, the equations for the spherical Bessel functions of the first kind and the corresponding roots are $j_0(r) = \mathrm{sinc}(r)$ and $u_{0n} = (n+1)\pi$. This allows $f_n(r) = f_{n0}(r)$ to be defined as

$$f_n(r) = (-1)^n \frac{\sqrt{2}\pi}{r_c^{3/2}} \frac{(n+1)(n+2)}{\sqrt{(n+1)^2 + (n+2)^2}} \left\{ \mathrm{sinc}\left[r \frac{(n+1)\pi}{r_c} \right] + \mathrm{sinc}\left[r \frac{(n+2)\pi}{r_c} \right] \right\}, \tag{A.17}$$

the evaluation of which does not involve any special functions. The recursion relations

that define $g_n(r) = g_{n0}(r)$ do not change:

$$d_n = 1 - \frac{e_n}{d_{n-1}} \tag{A.18}$$

$$g_n(r) = \frac{1}{\sqrt{d_n}} \left[f_n(r) + \sqrt{\frac{e_n}{d_{n-1}}} g_{n-1}(r) \right] \tag{A.19}$$

but the equation for $e_n = e_{n0}$ reduces to

$$e_n = \frac{n^2(n+2)^2}{4(n+1)^4 + 1}. \tag{A.20}$$

With these simplifications, the $g_n(r)$ can be evaluated with significantly less effort than the general $g_{nl}(r)$.

A.2. Force Calculation

Due to the dependence of the descriptors on the atomic positions, the calculation of the force vector $\bar{F} = -\nabla E$ requires the application of the chain rule. The jth component of the force on the ith atom is denoted by F_j^i, and is obtained by summing contributions from all N atoms:

$$F_j^i = -\sum_{k=1}^{N} \frac{\partial E^k}{\partial r_j^i} = -\sum_{k=1}^{N} \sum_{p=1}^{N_d} \frac{\partial E^k}{\partial G_p^k} \frac{\partial G_p^k}{\partial r_j^i}, \tag{A.21}$$

where r_j^i is the jth Cartesian coordinate of the ith atom, G_p^k is the pth descriptor for the kth atom, and N_d is the number of descriptors. The derivatives $\partial E^k / \partial G_p^k$ depend only on the NN architecture and can be calculated by back-propogation. The derivatives $\partial G_p^k / \partial r_j^i$ of the proposed descriptors with respect to the Cartesian coordinates require analytical differentiation. The descriptors for the kth atom are defined as

$$p_{nl}^k = \sum_{m=-l}^{l} c_{nlm}^* c_{nlm}. \tag{A.22}$$

Expanding this using the definition for the c_{nlm} from Chapter 3 gives

$$p_{nl}^{k} = \sum_{m=-l}^{l} \left(\sum_{q} g_n(r^{kq}) Y_l^m(\theta^{kq}, \phi^{kq}) \right) \left(\sum_{q} g_n(r^{kq}) Y_l^{m*}(\theta^{kq}, \phi^{kq}) \right). \tag{A.23}$$

Let $F_1(r) = g_n(r)$, $F_2(\theta) = \sqrt{\frac{2l+1}{2}\frac{(l-m)!}{(l+m)!}} P_{lm}(\cos\theta)$ and $F_3(\phi) = \frac{1}{\sqrt{2\pi}} e^{im\phi}$. Then p_{nl}^{k} can be redefined as a function of the relative spherical coordinates of the neighboring atoms as

$$p_{nl}^{k} = \sum_{m=-l}^{l} \left(\sum_{q} F_1(r^{kq}) F_2(\theta^{kq}) F_3(\phi^{kq}) \right) \left(\sum_{q} F_1(r^{kq}) F_2(\theta^{kq}) F_3^*(\phi^{kq}) \right). \tag{A.24}$$

Using Equation A.24 one can calculate the derivatives of p_{nl}^{k} with respect to the spherical coordinates of the qth atom relative to the kth atom as

$$\begin{aligned}
\frac{\partial p_{nl}^{k}}{\partial r^{kq}} &= \sum_{m=-l}^{l} \left[\frac{dF_1(r^{kq})}{dr^{kq}} F_2(\theta^{kq}) F_3(\phi^{kq}) c_{nlm} + c_{nlm}^{*} \frac{dF_1(r^{kq})}{dr^{kq}} F_2(\theta^{kq}) F_3^*(\phi^{kq}) \right] \\
\frac{\partial p_{nl}^{k}}{\partial \theta^{kq}} &= \sum_{m=-l}^{l} \left[F_1(r^{kq}) \frac{dF_2(\theta^{kq})}{d\theta^{kq}} F_3(\phi^{kq}) c_{nlm} + c_{nlm}^{*} F_1(r^{kq}) \frac{dF_2(\theta^{kq})}{d\theta^{kq}} F_3^*(\phi^{kq}) \right] \\
\frac{\partial p_{nl}^{k}}{\partial \phi^{kq}} &= \sum_{m=-l}^{l} \left[F_1(r^{kq}) F_2(\theta^{kq}) \frac{dF_3(\phi^{kq})}{d\phi^{kq}} c_{nlm} + c_{nlm}^{*} F_1(r^{kq}) F_2(\theta^{kq}) \frac{dF_3^*(\phi^{kq})}{d\phi^{kq}} \right]
\end{aligned} \tag{A.25}$$

The derivatives in Equation A.21 are with respect to absolute Cartesian coordinates, though the derivatives in Equation A.25 are with respect to relative spherical coordinates. The transformation from relative spherical to relative Cartesian is accomplished by multiplying by the Jacobian matrix of the transformation

$$\begin{bmatrix} \frac{\partial p_{nl}^{k}}{\partial x^{kq}} & \cdots \\ \frac{\partial p_{nl}^{k}}{\partial y^{kq}} & \cdots \\ \frac{\partial p_{nl}^{k}}{\partial z^{kq}} & \cdots \end{bmatrix}_{3\times N_d} = \begin{bmatrix} \frac{\partial r^{kq}}{\partial x^{kq}} & \frac{\partial \theta^{kq}}{\partial x^{kq}} & \frac{\partial \phi^{kq}}{\partial x^{kq}} \\ \frac{\partial r^{kq}}{\partial y^{kq}} & \frac{\partial \theta^{kq}}{\partial y^{kq}} & \frac{\partial \phi^{kq}}{\partial y^{kq}} \\ \frac{\partial r^{kq}}{\partial z^{kq}} & \frac{\partial \theta^{kq}}{\partial z^{kq}} & \frac{\partial \phi^{kq}}{\partial z^{kq}} \end{bmatrix} \begin{bmatrix} \frac{\partial p_{nl}^{k}}{\partial r^{kq}} & \cdots \\ \frac{\partial p_{nl}^{k}}{\partial \theta^{kq}} & \cdots \\ \frac{\partial p_{nl}^{k}}{\partial \phi^{kq}} & \cdots \end{bmatrix}_{3\times N_d} \tag{A.26}$$

where N_d is the number of descriptors. Finally, calculating the force vector on the ith atom requires differentiation of all atomic energies inside the cutoff with respect to the absolute Cartesian coordinates of the ith atom. This requires one final application of

the chain rule to convert relative coordinates to absolute coordinates by

$$\frac{\partial p^k_{nl}}{\partial x^i} = \frac{\partial p^k_{nl}}{\partial x^{kq}} \frac{\partial x^{kq}}{\partial x^i}$$
$$\frac{\partial p^k_{nl}}{\partial y^i} = \frac{\partial p^k_{nl}}{\partial y^{kq}} \frac{\partial y^{kq}}{\partial y^i}$$
$$\frac{\partial p^k_{nl}}{\partial z^i} = \frac{\partial p^k_{nl}}{\partial z^{kq}} \frac{\partial z^{kq}}{\partial z^i}$$

where the derivatives vanish unless k or q is equal to i.

APPENDIX B: CHAPTER 4

Sections B.1 and B.2 contain the derivation of the radial basis functions and the derivatives of the SB descriptors and the SOAP descriptors presented in Chapter 4, respectively.

B.1. Derivation of the Radial Basis Functions

This section contains the derivation of the radial basis functions in Chapter 4. Let $f_{nl}(r)$ be the linear combination of spherical Bessel functions

$$f_{nl}(r) = a_{nl} j_l\left(r\frac{u_{ln}}{r_c}\right) + b_{nl} j_l\left(r\frac{u_{l,n+1}}{r_c}\right) \tag{B.1}$$

where a_{nl} and b_{nl} are constants, $j_l(r)$ is the lth spherical Bessel function of the first kind, u_{ln} is the $(n+1)$th nonzero root of $j_l(r)$, and r_c is the cutoff radius. Since $f_{nl}(r_c) = 0$ by definition, the objective is to find a_{nl} and b_{nl} such that $f'_{nl}(r_c) = 0$ and $f''_{nl}(r_c) = 0$. Using the two differentiation rules for spherical Bessel functions [172]

$$j'_l(x) = j_{l-1}(x) - \frac{l+1}{x} j_l(x) \tag{B.2}$$

$$j'_l(x) = \frac{l}{x} j_l(x) - j_{l+1}(x), \tag{B.3}$$

the first and second derivatives of $f_{nl}(r)$ can be shown to vanish at $r = r_c$ if the coefficients in Equation B.1 satisfy

$$a_{nl} = \frac{u_{l,n+1}}{j_{l+1}(u_{ln})} c_{nl} \tag{B.4}$$

$$b_{nl} = -\frac{u_{ln}}{j_{l+1}(u_{l,n+1})} c_{nl} \tag{B.5}$$

for an arbitrary multiplicative constant c_{nl}. The value of c_{nl} is fixed by requiring that $f_{nl}(r)$ be appropriately normalized, or

$$\int_0^{r_c} f_{nl}(r) f_{nl}(r) r^2 \, \mathrm{d}r = 1. \tag{B.6}$$

This leads to

$$f_{nl}(r) = \left(\frac{1}{r_c^3} \frac{2}{u_{ln}^2 + u_{l,n+1}^2} \right)^{1/2} \left[\frac{u_{l,n+1}}{j_{l+1}(u_{ln})} j_l \left(r \frac{u_{ln}}{r_c} \right) - \frac{u_{ln}}{j_{l+1}(u_{l,n+1})} j_l \left(r \frac{u_{l,n+1}}{r_c} \right) \right] \tag{B.7}$$

as an explicit equation for the $f_{nl}(r)$.

Let the $g_{nl}(r)$ be the set of orthonormal radial basis functions derived by applying the Gram-Schmidt process to the $f_{nl}(r)$ for fixed l. For reference, the orthogonality relation for the spherical Bessel functions is [172]

$$\int_0^1 x^2 j_l(x u_{ln}) j_l(x u_{ln'}) \, \mathrm{d}x = \frac{\delta_{nn'}}{2} \left[j_{l+1}(u_{ln}) \right]^2 \tag{B.8}$$

where $\delta_{nn'}$ is the Kronecker delta. Using the substitution $x = r/r_c$, this is rewritten as

$$\int_0^{r_c} j_l \left(r \frac{u_{ln}}{r_c} \right) j_l \left(r \frac{u_{ln'}}{r_c} \right) r^2 \, \mathrm{d}r = \delta_{nn'} \frac{r_c^3}{2} \left[j_{l+1}(u_{ln}) \right]^2 . \tag{B.9}$$

For any integer $l \geq 0$, let $g_{0l}(r) = f_{0l}(r)$. The orthogonalization procedure involves constructing $g_{nl}(r)$ for $n > 0$ given $f_{nl}(r)$ and $g_{ml}(r)$ for $0 \leq m \leq n-1$. Notice that $g_{ml}(r)$ contains components of all $f_{pl}(r)$ for $0 \leq p \leq m$, and therefore terms involving $j_l(r \frac{u_{lq}}{r_c})$ for all $0 \leq q \leq m+1$. Since $f_{nl}(r)$ only contains terms involving $j_l(r \frac{u_{ln}}{r_c})$ and $j_l(r \frac{u_{l,n+1}}{r_c})$, $f_{nl}(r)$ is already orthogonal to all $g_{ml}(r)$ for $0 \leq m \leq n-2$ by the orthogonality of the spherical Bessel functions. The only remaining step is to subtract the projection of $f_{nl}(r)$ onto $g_{n-1,l}(r)$, or

$$h_{nl}(r) = f_{nl}(r) - g_{n-1,l}(r) \int_0^{r_c} f_{nl}(r) g_{n-1,l}(r) r^2 \, \mathrm{d}r. \tag{B.10}$$

$h_{nl}(r)$ is orthogonal to all $g_{ml}(r)$ for $l \leq m \leq n-1$, but is not yet normalized. Let

$$d_{nl} = \int_0^{r_c} h_{nl}(r) h_{nl}(r) r^2 \, \mathrm{d}r \tag{B.11}$$

be the squared magnitude of $h_{nl}(r)$. Then the desired orthonormal $g_{nl}(r)$ is

$$g_{nl}(r) = d_{nl}^{-1/2} h_{nl}(r). \tag{B.12}$$

Explicitly evaluating the integral

$$\int_0^{r_c} f_{nl}(r) g_{n-1,l}(r) r^2 \, \mathrm{d}r = \frac{a_{nl} b_{n-1,l}}{\sqrt{d_{n-1,l}}} \frac{r_c^3}{2} \left[j_{l+1}(u_{ln}) \right]^2 \tag{B.13}$$

allows Equations B.10, B.11 and B.12 to be solved to derive the recursion relations

$$d_{nl} = 1 - \frac{e_{nl}}{d_{n-1,l}} \tag{B.14}$$

$$g_{nl}(r) = \frac{1}{\sqrt{d_{nl}}} \left[f_{nl}(r) + \sqrt{\frac{e_{nl}}{d_{n-1,l}}} g_{n-1,l}(r) \right] \tag{B.15}$$

where the constants e_{nl} are defined as

$$e_{nl} = \frac{u_{l,n-1}^2 u_{l,n+1}^2}{(u_{l,n-1}^2 + u_{ln}^2)(u_{ln}^2 + u_{l,n+1}^2)}. \tag{B.16}$$

These recursion relations can be initialized with $d_{0l} = 1$ and $g_{0l}(r) = f_{0l}(r)$ for any $0 \leq l$.

B.2. Derivatives

B.2.1. Spherical Bessel Descriptors

The descriptors p_{nl} of atom i must be differentiated with respect to the relative spherical coordinates $(r^{ij}, \theta^{ij}, \phi^{ij})$ of the neighboring atoms to construct the Jacobian of the map to the space of descriptors. These derivatives are found using Equations

4.8 and 4.9 of the main text to be

$$\frac{\partial p_{nl}}{\partial r^j} = \frac{2l+1}{2\pi}\frac{\mathrm{d}g_{n-l,l}(r^j)}{\mathrm{d}r^j}\sum_k\left[g_{n-l,l}(r^k)P_l(\cos\gamma^{jk})\right]$$

$$\frac{\partial p_{nl}}{\partial \theta^j} = \frac{(2l+1)l}{2\pi}g_{n-l,l}(r^j)\sum_{k\neq j}\left\{g_{n-l,l}(r^k)\frac{1}{\cos^2\gamma^{jk}-1}[\cos\gamma^{jk}P_l(\cos\gamma^{jk}) - P_{l-1}(\cos\gamma^{jk})]\right.$$
$$\left.\times[\cos\theta^j\sin\theta^k\cos(\phi^j-\phi^k) - \sin\theta^j\cos\theta^k]\right\}$$

$$\frac{\partial p_{nl}}{\partial \phi^j} = \frac{(2l+1)l}{2\pi}g_{n-l,l}(r^j)\sum_{k\neq j}\left\{\frac{g_{n-l,l}(r^k)}{\cos^2\gamma^{jk}-1}[\cos\gamma^{jk}P_l(\cos\gamma^{jk}) - P_{l-1}(\cos\gamma^{jk})]\right.$$
$$\left.\times\sin\theta^j\sin\theta^k\sin(\phi^k-\phi^j)\right\}$$

where P_l is the Legendre polynomial of order l and γ^{jk} is the triplet angle between atoms i, j and k. The subscript i for the central atom is suppressed for clarity. The derivative of the radial basis functions can be calculated recursively using

$$\frac{\mathrm{d}g_{nl}(r)}{\mathrm{d}r} = \frac{1}{\sqrt{d_{nl}}}\left[\frac{\mathrm{d}f_{nl}(r)}{\mathrm{d}r} + \sqrt{e_{nl}d_{n-1,l}}\frac{\mathrm{d}g_{n-1,l}(r)}{\mathrm{d}r}\right]$$

$$\frac{\mathrm{d}f_{nl}(r)}{\mathrm{d}r} = \left(\frac{1}{r_c^3}\frac{2}{u_{ln}^2+u_{l,n+1}^2}\right)^{1/2}\left\{\frac{u_{l,n+1}}{j_{l+1}(u_{ln})}\left[\frac{l}{r}j_l\left(r\frac{u_{ln}}{r_c}\right) - \frac{u_{ln}}{r_c}j_{l+1}\left(r\frac{u_{ln}}{r_c}\right)\right]\right.$$
$$\left.-\frac{u_{ln}}{j_{l+1}(u_{l,n+1})}\left[\frac{l}{r}j_l\left(r\frac{u_{l,n+1}}{r_c}\right) - \frac{u_{l,n+1}}{r_c}j_{l+1}\left(r\frac{u_{l,n+1}}{r_c}\right)\right]\right\}.$$

The constants d_{nl} and e_{nl} are provided in Section B.1.

B.2.2. SOAP Descriptors

The SOAP descriptors [63,148] use a neighbor density function where the atomic densities of the surrounding atoms are given by unnormalized Gaussians multiplied by a cutoff function:

$$\rho(\bar{r}) = \sum_j \exp\left(-\frac{|\bar{r}-\bar{r}_j|^2}{2\sigma_a^2}\right)f_{\mathrm{cut}}(r_j). \tag{B.17}$$

As before, $\rho(\bar{r})$ is approximated by a truncated expansion over an orthogonal set of functions on the ball or radius r_c

$$\rho(\bar{r}) \approx \sum_{n=0}^{n_{\max}} \sum_{l=0}^{l_{\max}} \sum_{m=-l}^{l} c_{nlm} g_n(r) Y_l^m(\theta, \phi) \tag{B.18}$$

where the radial basis functions $g_n(r)$ are defined in Szlachta et al. [148] Using the coefficients of this expansion, the SOAP descriptors are defined as

$$p_{n_1 n_2 l} = \sum_m c_{n_1 l m} c^*_{n_2 l m}. \tag{B.19}$$

The equations in Bartok et al. [63] and Szlachta et al. [148] allow these descriptors to be written in a more extended form as

$$p_{n_1 n_2 l} = 4\pi(2l+1) \sum_{n_1' n_2'} \sum_{jk} (\mathbf{U}^{-1})_{n_1 n_1'} I'_{n_1' l}(r_j) f_{\text{cut}}(r_j) (\mathbf{U}^{-1})_{n_2 n_2'} I'_{n_2' l}(r_k) f_{\text{cut}}(r_k) P_l(\cos\gamma_{jk})$$

(B.20)

$$I'_{nl}(r_j) = \int_0^{r_c} \exp\left\{ -\left[\left(r - \frac{r_c n}{n_{\max}}\right)^2 + r^2 + r_j^2\right] \frac{1}{2\sigma_a^2} \right\} \iota_l\left(\frac{r r_j}{\sigma_a^2}\right) r^2 \mathrm{d}r \tag{B.21}$$

$$f_{\text{cut}} = \begin{cases} 1 & 0 < r \leq r_c - r_\Delta \\ \frac{1}{2}[1 + \cos(\pi \frac{r - r_c + r_\Delta}{r_\Delta})] & r_c - r_\Delta < r \leq r_c \\ 0 & r_c < r \end{cases} \tag{B.22}$$

where $\iota_l(r)$ is the modified spherical Bessel function of the first kind of order l, $\mathbf{U}$ is related to an overlap matrix [148], and r_Δ and σ_a are adjustable parameters. Repeated application of the chain rule then allows the partial derivatives of the SOAP descriptors with respect to the radial spherical coordinates of the neighboring atoms to be written

as

$$\frac{\partial p_{n_1 n_2 l}}{\partial r_j} = 4\pi(2l+1)\sum_k \Bigg(\sum_{n_1'} \left\{ (\mathbf{U}^{-1})_{n_1 n_1'} \left[\frac{\mathrm{d}I'_{n_1' l}(r_j)}{\mathrm{d}r_j} f_{\mathrm{cut}}(r_j) + I'_{n_1' l}(r_j) \frac{\mathrm{d}f_{\mathrm{cut}}(r_j)}{\mathrm{d}r_j} \right] \right\} \\ \times \sum_{n_2'} \left[(\mathbf{U}^{-1})_{n_2 n_2'} I'_{n_2' l}(r_j) f_{\mathrm{cut}}(r_j) \right] \\ + \sum_{n_2'} \left\{ (\mathbf{U}^{-1})_{n_2 n_2'} \left[\frac{\mathrm{d}I'_{n_2' l}(r_j)}{\mathrm{d}r_j} f_{\mathrm{cut}}(r_j) + I'_{n_2' l}(r_j) \frac{\mathrm{d}f_{\mathrm{cut}}(r_j)}{\mathrm{d}r_j} \right] \right\} \\ \times \sum_{n_1'} \left[(\mathbf{U}^{-1})_{n_1 n_1'} I'_{n_1' l}(r_j) f_{\mathrm{cut}}(r_j) \right] \Bigg) P_l(\cos\gamma_{jk})$$

where

$$\frac{\mathrm{d}I'_{nl}(r_j)}{\mathrm{d}r_j} = \int_0^{r_c} \exp\left\{ -\left[\left(r - \frac{r_c n}{n_{\max}} \right)^2 + r^2 + r_j^2 \right] \frac{1}{2\sigma_a^2} \right\} \\ \times \left[\left(\frac{r_j}{\sigma_a^2} + \frac{l}{r_j} \right) \iota_l \left(\frac{r r_j}{\sigma_a^2} \right) + \frac{r}{\sigma_a^2} \iota_{l+1} \left(\frac{r r_j}{\sigma_a^2} \right) \right] r^2 \mathrm{d}r$$

$$\frac{\mathrm{d}f_{\mathrm{cut}}}{\mathrm{d}r_j} = \begin{cases} 0 & 0 < r \leq r_c - r_\Delta \\ -\frac{\pi}{2r_\Delta} \sin(\pi \frac{r - r_c + r_\Delta}{r_\Delta}) & r_c - r_\Delta < r \leq r_c \\ 0 & r_c < r \end{cases}$$

At this point it is convenient to define the quantities $I_{nl}(r_j) = \sum_{n'} (\mathbf{U}^{-1})_{nn'} I'_{n'l}(r_j)$ to reduce the notational burden. The remaining partial derivatives can then be written as

$$\frac{\partial p_{n_1 n_2 l}}{\partial \theta_j} = 4\pi(2l+1) l \sum_{k \neq j} \{ [I_{n_1 l}(r_j) f_{\mathrm{cut}}(r_j) I_{n_2 l}(r_k) f_{\mathrm{cut}}(r_k) + I_{n_2 l}(r_j) f_{\mathrm{cut}}(r_j) I_{n_1 l}(r_k) f_{\mathrm{cut}}(r_k)] \\ \times \sin^{-2}(\gamma_{jk}) [P_{l-1}(\cos\gamma_{jk}) - \cos\gamma_{jk} P_l(\cos\gamma_{jk})] \\ \times [\cos\theta_j \sin\theta_k \cos(\phi_j - \phi_k) - \sin\theta_j \cos\theta_k] \},$$

$$\frac{\partial p_{n_1 n_2 l}}{\partial \phi_j} = 4\pi(2l+1) l \sum_{k \neq j} \{ [I_{n_1 l}(r_j) f_{\mathrm{cut}}(r_j) I_{n_2 l}(r_k) f_{\mathrm{cut}}(r_k) + I_{n_2 l}(r_j) f_{\mathrm{cut}}(r_j) I_{n_1 l}(r_k) f_{\mathrm{cut}}(r_k)] \\ \times \sin^{-2}(\gamma_{jk}) [P_{l-1}(\cos\gamma_{jk}) - \cos\gamma_{jk} P_l(\cos\gamma_{jk})] \\ \times [\sin\theta_j \sin\theta_k \sin(\phi_k - \phi_j)] \}.$$

As before, the partial derivatives with respect to the spherical angles vanish when $l = 0$, removing the need to evaluate Legendre polynomials with $l < 0$.