Transferability of Empirical Potentials and the Knowledgebase of Interatomic Models (KIM)

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Daniel S. Karls

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Ryan S. Elliott, Adviser
Ellad B. Tadmor, Co-Adviser

April 2016

Acknowledgments

I oftentimes wonder how many would-be scientists and mathematicians were deprived a life of academic study by being born into poverty or slavery. How many of them could have made contributions akin to those of Gauss or Euler? Within this vein of thought, my principal acknowledgment for this work should, perhaps, be attributed to nothing more than fortuity. However, having been placed in an opportune situation, I must extend thanks to those around me who have supported me throughout my education. Specifically, my wife Wei, my parents Connie and Steve, and my sister Samantha, have provided me with life experience which is without substitute. I thank my dear friend, Stanley, for his sage counsel.

I am greatly indebted to my advisors, Profs. Ellad B. Tadmor and Ryan S. Elliott, for their unrelenting patience, mentorship, and motivation. Beyond their duties as advisors, they have provided unique comradery and a mature perspective of both scientific occupation and life, in general. The inclusive atmosphere which they maintain in their work will undoubtedly be enjoyed by many students to come. Furthermore, they have been directly responsible for much of the financial assistance I have received during my studies. In addition to my funding as a research assistant provided by the National Science Foundation,[1] they were instrumental in aiding to secure several scholarships which I received. I wish to express my sincere gratitude to the University of Minnesota for awarding me the Doctoral Dissertation Fellowship (DDF), and to the Department of Aerospace Engineering & Mechanics for their generous contributions. In particular, I gratefully recognize my financial support under the McCollum Memorial Scholarship, the Dunning Copper Fellowship, and the Goodman Fellowship in Theoretical and Applied Mechanics. I must also acknowledge the additional faculty members whose exceptional efforts made receiving these awards possible. I first offer thanks to Prof. Roger L. Fosdick, who inspired my interest in the subject of mechanics early in my education and encouraged me along my path to graduate school. I further thank Prof. Perry H. Leo, who was a pleasure to work with while serving as a teaching assistant and was of great help in applying for the DDF. A prominent role in my graduate education was also played by Prof. Richard D. James, whom I credit for introducing me to the mathematical side of mechanics. Finally, special thanks go to Profs. J. Ilja Siepmann and Richard Linares for agreeing to serve on my examining committee and reviewing this manuscript.

The many people I have encountered in my academic endeavors have been an enduring reminder that humanity is not only the purpose of science, but its very fabric. It is with the

[1] Our project manager, Daryl Hess, was particularly supportive.

utmost respect that I humbly acknowledge Dr. Albert Bartók-Pártay, who was a kind source of guidance and insight while learning about the many aspects of Gaussian Approximation Potentials which played a large part in this work. His rare level of commitment to interaction with the scientific community and to academia as a whole has been an inspiration. Stimulating discussions with Prof. Talid Sinno regarding the philosophy of fitting empirical potentials and the nature of material properties are greatly appreciated. I also thank Dr. Edward M. Kober, who encouraged my pursuit of understanding representations of atomic environments and provided motivation to develop a general classification of them. It has been a privilege to learn from all of my officemates and the entire OpenKIM development team, and I look foward to our future undertakings together. Finally, I thank all of the KIM users and everyone who has participated in our workshops for sharing the vision of the project.

Computing resources used in this work were graciously provided by the Minnesota Supercomputing Institute (MSI).

This work is dedicated to you, the reader. Without you, this manuscript would simply be more (digital) ink on more (digital) paper.

Abstract

Empirical potentials have proven to be an indispensable tool in understanding complex material behavior at the atomic scale due to their unrivaled computational efficiency. However, as they are currently used in the materials community, the realization of their full utility is stifled by a number of implementational difficulties. An emerging project specifically aimed to address these problems is the Knowledgebase of Interatomic Models (KIM). The primary purpose of KIM is to serve as an open-source, publically accessible repository of standardized implementations of empirical potentials (Models), simulation codes which use them to compute material properties (Tests), and first-principles/experimental data corresponding to these properties (Reference Data). Aside from eliminating the redundant expenditure of scientific resources and the irreproducibility of results computed using empirical potentials, a unique benefit offered by KIM is the ability to gain a further understanding of a Model's *transferability*, i.e. its ability to make accurate predictions for material properties which it was not fitted to reproduce. In the present work, we begin by surveying the various classes of mathematical representations of atomic environments which are used to define empirical potentials. We then proceed to offer a broad characterization of empirical potentials in the context of machine learning which reveals three distinct categories with which any potential may be associated. Combining one of the aforementioned representations of atomic environments with a suitable regression technique, we define the Regression Algorithm for Transferability Estimation (RATE), which permits a quantitative estimation of the transferability of an arbitrary potential. Finally, we demonstrate the application of RATE to a specific training set consisting of bulk structures, clusters, surfaces, and nanostructures of silicon. A specific analysis of the underlying quantities inferred by RATE which are used to characterize transferability is provided.

Contents

List of Tables

List of Figures

Each piece, or part, of the whole of nature is always merely an approximation to the complete truth, or the complete truth so far as we know it. In fact, everything we know is only some kind of approximation, because we know that we do not know all the laws as yet.

THE FEYNMAN LECTURES ON PHYSICS, VOLUME I
CHAPTER I: "ATOMS IN MOTION"

Chapter 1

Introduction

Materials have long played a vital role in the evolution of technology. From the structural design of bridges to life-saving medical breakthroughs, the science of materials has aided the advancement of technologies which address a myriad of military and civilian needs. This has acted to decrease the manufacturing cost of many existing devices while simultaneously improving their efficiency, robustness, and overall performance. Moreover, entirely novel applications have been enabled by the introduction of unique materials which stretch the bounds of feasibility in a range of technologies [1–3].

One of the salient features of materials is the apparent discrepancy in their function at different length scales. In everyday life, we observe materials at the *macroscale*, where they may typically be accurately represented using models which treat matter as a continuous medium whose state is characterized by various field quantities such as stress, strain, and temperature [4]. However, all material behavior ultimately originates from the interactions of individual atoms at the *nanoscale* where these models are insufficient, and the manner in which physical entities behave is best understood in the context of the statistical language of quantum mechanics. The reconciliation of the models developed to study these contrasting depictions of reality has become a topic of interest in a number of scientific disciplines, giving rise to diverse speculation with respect to both the explanatory capacity of different physical theories and the most appropriate methods of computational simulation. The field of materials science and engineering, in particular, has become increasingly involved in this subject over the past twenty years, where it has become known by the moniker of *multiscale modeling* [5].

While macroscopic processes are unequivocally important for much of the technology we

use, a growing number of recent innovations in fields such as biotechnology and microprocessor design have begun reaching deeper into the nanoscale than was previously thought possible. At these scales, the capabilities of experimental observation are extremely sparse, and we must resort to numerical simulation in order to gain an adequate understanding of material behavior. The most accurate, *first-principles* methods for such simulations require the explicit consideration of the electrons present in an atomistic system. As a result of the complex behavior unique to electronic structure, calculations of this sort are prohibitively expensive for the length and time scales corresponding to the majority of microscopic interactions present in engineering applications, and are usually limited to a few thousand atoms over the course of several picoseconds. This impediment has accordingly prompted the creation of *empirical potentials* (EPs) as a means of ingress to a greater breadth of physical phenomena. The empirical models applied within this domain make use of the pronounced separation in the time scale pertinent to electrons and the time scale associated with the protons and neutrons which comprise the nuclei of atoms. Because electrons are significantly less massive than protons and neutrons, they tend to equilibrate rapidly in response to changes in the positions of the nuclei and, consequently, the electronic degrees of freedom may often be subsumed by the nuclear positions with a negligible sacrifice in accuracy. By exploiting this property, empirical potentials permit a dramatic escalation up the cascade of length and time scales, providing a bridge between quantum mechanics and statistical mechanics which serves as a valuable contribution to the larger space of multiscale modeling. For example, it is possible to use such models to simulate millions and even billions of atoms over time scales long enough to encompass a wealth of material behavior such as grain boundary motion and diffusion [6–9].

Despite the tremendous promise shown by empirical atomistic modeling, it has thus far been plagued by a host of implementational difficulties. These problems stem from the fact that, other than a small number of common potentials packaged natively alongside popular molecular simulation software, the vast majority of empirical atomistics is performed in an isolated fashion. Aside from the possibility of informal collaboration through personal communications, each research group is responsible for producing its own implementation of a given empirical model. There are several prominent technical deficiencies which exist in this method of operation. First, it is evident that the creation of the same empirical potential in parallel by different researchers constitutes a redundant use of development time and energy. An insidious consequence of this separation is that it also acts to preclude the inheritance of bug fixes and optimizations discovered for implementations of empirical potentials that would otherwise occur if a concerted effort were undertaken. Furthermore, the

empirical potential codes produced by different groups may be written in such a way that they are tailor-fitted to run in a specific simulation environment and lack significant portability. Altogether, these technical issues limit the number of empirical potentials which are investigated in individual research endeavors, as well as in the scientific community as a whole. Finally, it should not be overlooked that empirical potentials are only as useful as the simulations which use them to resolve material properties, and it should be noted that the codes which utilize empirical potentials are similarly fragmented and are currently subject to all of the same hindrances as the potentials themselves.

An emerging project specifically aimed to address the aforementioned problems is the *Open Knowledgebase of Interatomic Models* (OpenKIM, KIM) [10–12]. KIM provides an open-source, publically accessible repository of empirical potentials, simulation codes which use them to compute material properties, and first-principles/experimental data corresponding to these properties. The first notable feature of this system is that each item in the OpenKIM Repository (**https://openkim.org**) possesses a unique identifier which functions as a version control mechanism, enabling true data provenance. Next, all uploaded content is designed to conform to the KIM application programming interface (API), which provides a standardized method of communicating the data relevant to an atomistic simulation such as atomic positions, forces, etc., in a cross-language format. As a result, the empirical potentials in KIM can be used portably with a variety of molecular simulation software, with popular packages such as ASAP [13], ASE [14], DL_POLY [15], GULP [16], IMD [17], LAMMPS [18], and libAtoms+QUIP [19] already featuring support. Finally, the principal separation of KIM, a knowledgebase, from similar projects such as the Interatomic Potentials Repository Project created by the National Institute of Standards and Technology (NIST) [20] or the Materials Project [21] is that the content in KIM continually interacts with itself. Whenever a new potential is uploaded, it is automatically mated with all existing compatible simulations in the OpenKIM Repository and the resulting model-simulation combinations are executed. Conversely, whenever a new simulation is uploaded, it is automatically mated with all existing potentials and executed. This important distinction is only made possible with the aid of the KIM API.

Apart from difficulty in implementation, there also exist obstacles related to how potentials are used that have not been rigorously accounted for. These problems pertain to what is known as the *transferability* of a potential, which refers to its ability to accurately predict material properties which it was not fitted to reproduce. Because of the complexity of the underlying quantum mechanical potential energy surface for any system containing more than a handful of atoms, it is non-trivial to approximate it by means of empirical

potentials composed of fixed functional forms. Hence, most empirical potentials will have limited transferability across the configuration space of an atomic system, yielding material properties which are generally inconsistent in terms of their accuracy compared to first-principles calculations and/or experiment. Currently, the decision of which potential to employ in a given application is guided largely by the intuition and personal experience of the researcher(s) involved. However, KIM presents a profitable opportunity in this regard: by compiling observations of the accuracy of an empirical potential at various atomic configurations from its predictions and the corresponding first-principles/experimental data stored in the OpenKIM Repository, it is possible to systematically predict its transferability across configuration space. The resulting algorithm, which serves as the primary focus of the present work, can then be used to select a potential from a pool of candidates for a specific application.

Before commencing our exposition, we pause to consider a broad overview of its contents. In Chapter 2, we begin by explaining the physical problem which interatomic potentials were created to solve. We then briefly summarize the most common first-principles method used to this end and introduce the abstract concept of an empirical potential, followed by a review of a sensible taxonomy for classifying empirical potentials. Chapter 3 delves further into the KIM framework, including its overall organization and operating infrastructure, as well as the specific types of content it contains and how they facilitate the study of transferability. Chapter 4 surveys the different classes of mathematical representations of atomic environments which form an essential component of any empirical potential. In Chapter 5, we present the most general possible classification of empirical potentials in the context of regression in machine learning, which reveals three distinct paradigms with which any potential can be associated; particular attention is paid to the prospective transferability of each type of potential. Combining one of the aforementioned representations of atomic environments with a suitable regression technique, we describe an algorithm which permits a quantitative estimation of the transferability of an arbitrary potential in Chapter 6 along with an analysis of numerical results. Finally, we conclude with a synopsis and prospects for future work.

Chapter 2

Background on Atomistic Modeling

Perhaps the most important contributions to human consciousness come from our immediate sensory perceptions. These observations occur at length and time scales which far exceed those germane to the atoms (and subatomic matter) which comprise the world around us. At the former set of scales, and any scales of greater extent, the rules used to predict physical phenomena assume that the matter they govern is *particle-like* in nature. That is to say, they rely on the assumption that matter has a definite position at any particular instant in time. This line of thought suffices to explain a large fraction of processes relevant to our daily lives insofar as the engineering of technology and understanding of natural events are concerned. In fact, many such models are still in active development and their significance, both practical and philosophical, is difficult to overstate.

Contrary to what may be tempting to infer from the name, particle-like explanations of matter may assume it to be either discrete or continuous. Consider, for example, one of the largest cosmological structures we know to exist: the Milky Way galaxy. Similar to a fluid, the internal structure of the Milky Way is most efficiently represented as a continuum, owing to the unfathomable number of constituents which it encompasses [22]. Proceeding further down the scales of length and time, we arrive at the notion of an individual solar system. The planetary motion within our own solar system was addressed with stunning accuracy by the formulation built upon Kepler's laws by Newton, which treats each planet as a point mass possessing a definite trajectory over time. Descending further still, we arrive at the geophysical laws which govern the planet we know as Earth, which once again assume matter is continuous and permit the precise prediction of complex tectonic evolution. All of these models, while differing in the dimensionality of their representation of the material degrees of freedom, assume a particle-like quality of their subjects.

Because our ability to perceive matter was confined to relatively large scales for so long, it comes as no surprise that when shrewdly designed experiments finally managed to effectively probe the nanoscale, there was a great deal of contention as to how to account for what was being observed [23]. Not only was it determined that all physical objects (including light) could display particle-like behavior, but moreover that the very same objects could display *wave-like* behavior, wherein their positions were apparently distributed across space at a single instance of time. While neither purely particle-like models nor purely wave-like models are independently capable of fully explaining the totality of experimental observations, it has become evident that both are apropos when dealing with matter which has mass on the scale of atoms. This profound realization has proven to be a metamorphic event in scientific history, and has resulted in a dichotomous set of explanations for nanoscopic physics. One can choose to construe matter either as a collection of delocalized waves which sometimes contract to behave like particles [24, 25], or as particulate entities which possess a definite position and momentum, but which are inextricably affixed to an oscillating wave-like medium [26–30]. The relative validity of these points of view continues to be the subject of intense scrutiny by the scientific community and, additionally, carries a fascinating philosophical debate as to whether physics is intrinsically aleatory or deterministic [31–33].

2.1 Schrödinger's formulation of quantum mechanics

Although the existential interpretation of matter is beyond the scope of interest of this manuscript, the aforementioned wave-particle duality does not impede our ultimate goal: the accurate prediction of atomistic processes pertinent to engineering applications. In the interest of pragmatism, we may choose to formally typify some matter as waves and other matter as particles in order to obtain a tractable overall solution. Because the mass of an individual proton or neutron is far greater than the mass of an individual electron, it is well-established that the former can be treated as classical particles, and the consideration of the wave-like behavior of atoms can be limited to their electrons. This can be accomplished by representing all of the N^{el} electrons in a system of atoms concurrently with a complex-valued *wave function* $\chi(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N^{\text{el}}}, t)$, where $\boldsymbol{x}$ denotes spatial position and t denotes time. The evolution of this wave function over time, subject to a potential exerted on the electrons, is governed by the celebrated *Schrödinger equation*. For brevity, we quickly summarize the derivation of this equation given in [5].

To maintain simplicity, we consider a single classical electron with mass m^{el}, position $\boldsymbol{r}(t)$, and momentum $\boldsymbol{p}(t)$, which moves in a time-varying potential field $\mathcal{V}(\boldsymbol{x}, t)$. The

Hamiltonian function for this single-particle system, which represents its total energy, is given by

$$\mathcal{H}_{\text{classical}}(\boldsymbol{r}(t), \boldsymbol{p}(t), t) = \mathcal{V}(\boldsymbol{r}(t), t) + \frac{||\boldsymbol{p}(t)||^2}{2m^{\text{el}}} \triangleq \mathcal{V}(\boldsymbol{r}(t), t) + \mathcal{T}_{\text{classical}}(\boldsymbol{p}(t)), \tag{2.1}$$

In order to analyze a quantum mechanical system in terms of a wave function, the concept of the Hamiltonian must be translated accordingly. In the quantum framework, the Hamiltonian manifests itself mathematically as an *operator* which acts on a wave function, rather than as a function which takes as input the positions and momenta of a set of discrete particles. In transitioning from the classical potential energy field $\mathcal{V}(\boldsymbol{x}, t)$, we write the quantum potential energy operator as $\mathcal{U} = \mathcal{U}(\boldsymbol{x}, t)$. The roles of $\mathcal{V}(\boldsymbol{x}, t)$ and $\mathcal{U}(\boldsymbol{x}, t)$ are quite similar, but they are distinguished in terms of their interpretation. Whereas $\mathcal{V}(\boldsymbol{x}, t)$ indicates the potential energy of our single electron when it has the definite position $\boldsymbol{x}$ at time t, the potential energy operator $\mathcal{U}(\boldsymbol{x}, t)$ is a function[1] which is also capable of rendering a specific total potential energy, but for the *single-electron wave function* $\chi(\boldsymbol{x}, t)$ which ascribes to the electron a *distribution* over space for each time t. Specifically, the way in which $\mathcal{U}$ yields a total potential energy from a wave function is through its *expectation value*. The expectation value of $\mathcal{U}(\boldsymbol{x}, t)$ with respect to wave function $\chi(\boldsymbol{x}, t)$ is written in Dirac notation [34] as

$$\langle \mathcal{U}(\boldsymbol{x}, t) \rangle = \langle \chi \,|\, \mathcal{U}(\boldsymbol{x}, t) \,|\, \chi \rangle = \int \frac{\overline{\chi(\boldsymbol{x}, t)} \mathcal{U}(\boldsymbol{x}, t) \chi(\boldsymbol{x}, t)}{\langle \chi | \chi \rangle} d\boldsymbol{x}, \tag{2.2}$$

where $\overline{(\,\cdot\,)}$ denotes complex conjugation and the integration is formally carried out over all of space. The term in the denominator is present in order to normalize $\chi(\boldsymbol{x}, t)$ if necessary so that, at any time t, we have

$$\langle \chi | \chi \rangle = \int \overline{\chi(\boldsymbol{x}, t)} \chi(\boldsymbol{x}, t) d\boldsymbol{x} = 1. \tag{2.3}$$

Note that even when the expectation value of a time-independent operator is computed, the

[1] In this context, we may regard a function as a multiplicative operator.

result of (2.2) is still generally a function of time due to the time dependence[2] of $\chi(\boldsymbol{x}, t)$. Meanwhile, the kinetic energy operator takes a markedly different form than the classical kinetic energy,

$$\mathcal{T} = -\frac{\hbar^2 \nabla^2}{2m^{\text{el}}}, \tag{2.4}$$

where $\hbar = h/2\pi \approx 1.054571726 \times 10^{-34} \text{J} \cdot \text{s}$ is the reduced Planck's constant and ∇^2 is the Laplacian operator. Substituting these expressions, we find that the quantum Hamiltonian operator for our single-electron system is given by

$$\mathcal{H} = \mathcal{U}(\boldsymbol{x}, t) - \frac{\hbar^2 \nabla^2}{2m^{\text{el}}}. \tag{2.5}$$

Applying the definition of the expectation value to the quantum Hamiltonian of (2.5), we have

$$\langle \mathcal{H} \rangle = \left\langle \chi \middle| \mathcal{U}(\boldsymbol{x}, t) - \frac{\hbar^2 \nabla^2}{2m^{\text{el}}} \middle| \chi \right\rangle. \tag{2.6}$$

At this point, we appeal to the relations of Planck and de Broglie, which explicitly couple the particle-like and wave-like properties of matter, stating that

$$\epsilon = \hbar\omega, \qquad \boldsymbol{p} = \hbar \boldsymbol{k}. \tag{2.7}$$

Here, ϵ and $\boldsymbol{p}$ are the energy and momentum of the particle description, while ω and $\boldsymbol{k}$ are the temporal frequency and wave vector in the complementary wave description. We will henceforth assume that any admissible wave function $\chi(\boldsymbol{x}, t)$ which describes our single-electron system can be expressed in a basis of plane waves:

$$\chi(\boldsymbol{x}, t) = \frac{1}{\sqrt{(2\pi)^3}} \int \widehat{\chi}(\boldsymbol{k}, t) \exp(i\boldsymbol{k} \cdot \boldsymbol{x} - \omega(\boldsymbol{k})t) d\boldsymbol{k}. \tag{2.8}$$

By $(2.7)_1$, each plane wave of the basis above has an energy of $\epsilon = \hbar\omega(\boldsymbol{k})$ and, consequently, the total energy associated with the wave function $\chi(\boldsymbol{x}, t)$ can be written as the expectation $\langle \epsilon \rangle = \hbar \langle \omega(\boldsymbol{k}) \rangle$. It can be shown that the quantum mechanical operator corre-

[2]The time dependence of the expectation values of quantum-level quantities may equivalently be understood by using the *matrix mechanics* formulation of Born, Heisenberg, and Jordan [35]. In this alternative derivation, the time dependence is generally subsumed by the quantum mechanical operators themselves, rather than by the quantum states as in Schrödinger's derivation. A specific case of interest is the time evolution of the expectation values of position and momentum: $\langle \boldsymbol{x} \rangle$ and $\langle \boldsymbol{p} \rangle$. This special case is subject to *Ehrenfest's theorem* [36], which states that these expectation values evolve over time in accordance with the classical Hamiltonian (or, equivalently, Newton's laws of motion).

sponding to the temporal frequency $\omega(\boldsymbol{k})$ is

$$\omega(\boldsymbol{k}) \to i\frac{\partial}{\partial t}. \tag{2.9}$$

Thus, in terms of the wave function, we can write the expectation of the energy as

$$\langle \epsilon \rangle = \left\langle \chi \left| i\hbar \frac{\partial}{\partial t} \right| \chi \right\rangle. \tag{2.10}$$

Finally, we invoke what is known as the *correspondence principle* [36], which states that the expectation values of quantum operators correspond to physical observables and must therefore agree in the macroscopic limit. Accordingly, we equate (2.6) and (2.10), deducing that the action of the corresponding operators on the wave function must coincide, and arrive at the *time-dependent Schrödinger equation*:

$$\left(\mathcal{U}(\boldsymbol{x}, t) - \frac{\hbar^2}{2m^{\text{el}}} \nabla^2 \right) \chi(\boldsymbol{x}, t) = i\hbar \frac{\partial}{\partial t} \chi(\boldsymbol{x}, t). \tag{2.11}$$

Although we derived this equation for a system consisting of a single electron moving in an external potential for simplicity, the same equation applies if we had instead considered a system consisting of multiple particles. For instance, if we had a system of N^{el} electrons with positions $\boldsymbol{r}^1, \ldots, \boldsymbol{r}^{N^{\text{el}}}$, we would represent all of them with a single *many-electron wave function* $\chi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\text{el}}}, t)$. In this case, the kinetic energy operator is summed over each electron and the potential energy operator would accordingly be a function of all of the positions, $\mathcal{U} = \mathcal{U}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\text{el}}}, t)$, so as to include the interactions between the electrons as well as any external potential which acts on them. Thus, for multiple electrons we write the time-dependent Schrödinger equation as

$$\left(\mathcal{U}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\text{el}}}, t) - \frac{\hbar^2}{2m^{\text{el}}} \sum_{\alpha=1}^{N^{\text{el}}} \nabla_\alpha^2 \right) \chi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\text{el}}}, t) = i\hbar \frac{\partial}{\partial t} \chi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\text{el}}}, t), \tag{2.12}$$

where ∇_α^2 represents the Laplacian with respect to the coordinates of the αth electron.

As mentioned at the beginning of this section, we will be particularly interested in treating atomic nuclei as classical particles and representing the electrons with a wave function which evolves according to (2.12). Our assertion is grounded in the fact that the mass[3] of a proton or neutron is roughly 1836 times greater than that of an electron. As an example of the effect this has, consider the Coulombic interaction between a classical proton and

[3] We refer here to the rest mass, as we are ignoring relativistic effects.

a single classical electron. The Coulomb force between the two particles has a magnitude proportional to the absolute value of the product of the individual charges (denoted e) divided by the square of the distance r separating them:

$$|\boldsymbol{f}_{\text{Coulomb}}| \propto \frac{e^2}{r^2}. \tag{2.13}$$

Since we are treating both particles as classical in this case, we use Newton's laws to write

$$|\boldsymbol{f}_{\text{Coulomb}}| = m^{\text{pr}}|\mathbf{a}^{\text{pr}}| = m^{\text{el}}|\mathbf{a}^{\text{el}}|, \tag{2.14}$$

where m^{pr} and m^{el} are the respective masses of the proton and the electron, with $\mathbf{a}^{\text{pr}}$ and $\mathbf{a}^{\text{el}}$ denoting the accelerations each experiences (which are oppositely directed). Dividing both sides of (2.14) by m^{pr} indicates that the magnitude of the acceleration of the proton due to the Coulombic interaction with the electron is only approximately (1/1836)th the acceleration imparted to the electron. As a result, the proton remains essentially stationary during the subsequent motion of the electron. Even when many electrons are present, and regardless of whether the electrons are represented classically or by wave functions, their time-evolution may generally be considered exceedingly faster than that of the nuclei.[4]

The fact that the time scale associated with the motion of the electrons is much smaller than the time scale associated with the motion of the nuclei is embodied in the so-called *Born–Oppenheimer approximation* (BOA). The principal consequence of the BOA is that we may regard the nuclear motion as quasi-static with respect to the electrons. Another way to state this is that the electrons are always capable of instantaneously equilibrating to their ground state as the nuclei evolve in time. The mathematical consequence of this disparity is that we need only encompass the influence of the nuclei on the electrons in an external potential term $\mathcal{U}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\text{el}}})$ which is independent of time. Returning to the single-electron case momentarily, we can see that if $\mathcal{U}(\boldsymbol{x}, t) = \mathcal{U}(\boldsymbol{x})$ has no dependence on time, then (2.11) becomes separable in $\boldsymbol{x}$ and t. We can correspondingly separate the wave function as

$$\chi(\boldsymbol{x}, t) = \psi(\boldsymbol{x})\tau(t), \tag{2.15}$$

where it is now assumed that the spatial component $\psi(\boldsymbol{x})$ of any permissible wave function

[4]While this assumption can be used in most contexts without incurring significant error, there are some important cases where it breaks down due to the rapid motion of the nuclei, as noted in [5]. Such cases include, for example, the fracture of solids or the behavior of molecules containing hydrogen atoms. We forego these cases here and refer to the reader to [37, 38] and references therein.

can be written in a basis of time-independent plane waves:

$$\psi(\boldsymbol{x}) = \frac{1}{\sqrt{(2\pi)^3}} \int \widehat{\psi}(\boldsymbol{k}) \exp(i\boldsymbol{k} \cdot \boldsymbol{x}) d\boldsymbol{k}. \tag{2.16}$$

We thus rewrite (2.11) as

$$\left(\mathcal{U}(\boldsymbol{x}) - \frac{\hbar^2}{2m^{\mathrm{el}}} \nabla^2 \right) \psi(\boldsymbol{x}) = \epsilon \psi(\boldsymbol{x}), \tag{2.17}$$

where we have additionally dropped the time dependence of the wave function. We term this equation the *time-independent Schrödinger equation*, which for the many-electron case similarly appears as

$$\left(\mathcal{U}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\mathrm{el}}}) - \frac{\hbar^2}{2m^{\mathrm{el}}} \sum_{\alpha=1}^{N^{\mathrm{el}}} \nabla_\alpha^2 \right) \psi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\mathrm{el}}}) = \epsilon \psi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\mathrm{el}}}). \tag{2.18}$$

Both (2.17) and (2.18) may readily be identified as eigenproblems[5] in which the wave function solutions $\psi(\boldsymbol{x})$ (or $\psi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\mathrm{el}}})$ in the latter case) are the eigenfunctions of the Hamiltonian operator which appears on the left-hand side. Physically speaking, the eigenfunctions are interpreted as a discrete set of states which the electrons may assume, with the associated eigenvalues ϵ indicating the energy of each state. Finally, in the way of terminology, we remark that computational schema which attempt to solve either of (2.17) or (2.18) are referred to as *first-principles* or *ab initio* methods.

2.2 Traditional empirical methods

In the previous sections, we saw that the physics of all matter is ultimately governed by the Schrödinger equation. While this equation is built upon the assumption that the protons and neutrons of an atomistic system can be treated classically, its most basic form still generally involves time-dependent potential energy operators and a time-dependent many-electron wave function. However, by leveraging the Born–Oppenheimer approximation, we managed to reduce the Schrödinger equation to a form in which the time and space dependence of the many-electron wave function could be separated. The equation subsequently obtained, the time-independent Schrödinger equation, is an eigenequation which relates the spatial many-electron wave function $\psi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\mathrm{el}}})$ to the time-independent

[5]The interested reader may note that this form is more generally referred to as a *Sturm–Liouville* equation, and may refer to [39] for a gentle introduction to the associated mathematical theory, which pertains to eigenvalue problems involving general self-adjoint differential operators.

quantum Hamiltonian operator.

Because the time-independent Schrödinger equation is still an insurmountably difficult equation to solve in practice, an approximate method of solution known as *density functional theory* (DFT) has become widely proliferated. The central idea of DFT is to recast the problem of solving for the many-electron wave function with an equivalent problem in which the electron density $\rho_{\text{el}}(\boldsymbol{x})$ is the fundamental quantity which is sought.[6] This reformulation then leads one to the greatly simplified problem of solving for a single-electron wave function $\psi(\boldsymbol{x})$, while still retaining accuracy. Regrettably, as it pertains to most material properties important to engineering and design, which are of a dynamical nature, this simplified problem still presents a computational expense which is unwieldy for modern computers even with highly efficient implementations. For this reason, we now turn to the subject of *empirical potentials* (EPs). Like all practical implementations of DFT, EPs are predicated on the BOA, and thus attempt to approximate the Born–Oppenheimer potential energy surface (BOPES). However, rather than attempting to solve an eigenequation of any sort in order to recover the solutions to the time-independent Schrödinger equation, these models forego the electronic degrees of freedom completely, instead basing themselves solely on heuristic prediction of the behavior of the internal potential energy in terms of the nuclear coordinates. Restricting ourselves to the monoatomic case, the motivation of EPs begins with the assertion that the total internal potential energy, which we denote hereafter by $\mathcal{V}$, of a system containing N nuclei with positions $\boldsymbol{r}^1, \ldots, \boldsymbol{r}^N$ can be written as

$$\mathcal{V} = \mathcal{V}(\boldsymbol{r}^1, \ldots, \boldsymbol{r}^N). \tag{2.19}$$

The foundation of EPs is to then decompose the energy into a sum of terms involving a sequence of interactions of increasing order between the nuclei, similar to a Taylor expansion:

$$\mathcal{V} = \mathcal{V}_0 + \frac{1}{2}\sum_{\substack{\alpha,\beta \\ \alpha\neq\beta}} \hat{\mathcal{V}}_2(\boldsymbol{r}^\alpha, \boldsymbol{r}^\beta) + \frac{1}{3!}\sum_{\substack{\alpha,\beta,\gamma \\ \alpha\neq\beta\neq\gamma}} \hat{\mathcal{V}}_3(\boldsymbol{r}^\alpha, \boldsymbol{r}^\beta, \boldsymbol{r}^\gamma) + \frac{1}{4!}\sum_{\substack{\alpha,\beta,\gamma,\delta \\ \alpha\neq\beta\neq\gamma\neq\delta}} \hat{\mathcal{V}}_4(\boldsymbol{r}^\alpha, \boldsymbol{r}^\beta, \boldsymbol{r}^\gamma, \boldsymbol{r}^\delta) + \ldots. \tag{2.20}$$

It can then be shown[7] that, if $\mathcal{V}$ is to remain invariant to changes of reference frame and permutation of the (arbitrary) indexing of the nuclei, it must further be true that it can be

[6] The justification for the use of $\rho_{\text{el}}(\boldsymbol{x})$ as a surrogate for the many-electron wave function $\psi(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N^{\text{el}}})$ is embodied in the *Hohenberg–Kohn theorems* [40].

[7] See the text by Tadmor and Miller [5] for a proof.

written in terms of the pairwise distances between the nuclei:

$$\begin{aligned}\mathcal{V} =& \frac{1}{2}\sum_{\substack{\alpha,\beta \\ \alpha\neq\beta}} \mathcal{V}_2(r^{\alpha\beta}) + \frac{1}{3!}\sum_{\substack{\alpha,\beta,\gamma \\ \alpha\neq\beta\neq\gamma}} \mathcal{V}_3(r^{\alpha\beta}, r^{\alpha\gamma}, r^{\beta\gamma}) \\ &+ \frac{1}{4!}\sum_{\substack{\alpha,\beta,\gamma,\delta \\ \alpha\neq\beta\neq\gamma\neq\delta}} \mathcal{V}_4(r^{\alpha\beta}, r^{\alpha\gamma}, r^{\alpha\delta}, r^{\beta\gamma}, r^{\beta\delta}, r^{\gamma\delta}) + \ldots, \end{aligned} \tag{2.21}$$

where we have dropped the arbitrary constant term $\mathcal{V}_0$ as a matter of convenience, as it is inconsequential for our purposes. This brings us to the coarsest classification of EPs which is commonly used. Those EPs which consider only the first term of the expansion of (2.21) are known as *pair potentials*, while all EPs which make use of more than only the first term are collectively known as *many-body potentials* or *cluster potentials*. Many-body potentials are more specifically categorized according to precisely how many terms in the expansion they include: EPs which include the two-body and three-body terms are known as *three-body potentials*, those which include up to four-body terms are known as *four-body potentials*, etc.[8] A broader characterization of EPs due to Carlsson [41] is obtained by considering the analytical stratification of their functional forms, which we summarize next.

2.2.1 Pair potentials

As aforementioned, all EPs assume that the total internal potential energy $\mathcal{V}$ can be written in terms of the pairwise distances of the nuclei. However, pair potentials assert that only functions which consider at most two nuclei concurrently must be used to render a reasonable approximation to the true $\mathcal{V}$ which would be obtained from first principles. That is, a pair potential is written in the form

$$\mathcal{V} = \frac{1}{2}\sum_{\substack{\alpha,\beta \\ \alpha\neq\beta}} \mathcal{V}_2(r^{\alpha\beta}). \tag{2.22}$$

This simple form has been found to be ideally suited to model gaseous phases of materials, and remains the standard when modeling noble gases. However, because only two nuclei are ever considered simultaneously, the pair potential formalism only associates energies with the lengths of individual bonds, with no notion of the angular bond dependence which

[8] While three-body potentials have demonstrated exceptional capability to describe metallic and semi-metallic materials and are used in abundance, defining accurate many-body potentials which include terms higher than the three-body summation is notoriously difficult and is thus seldom done.

forms the hallmark of covalent solids.

2.2.2 Pair functionals

In order to more accurately capture the influence which chemical bonds have on one another, one can introduce a functional U to (2.22). As a functional, this quantity is a scalar-valued function which itself depends on another function which corresponds to an effective density experienced by a given atom. The defining feature of *pair functionals* is that they assume that the latter density is a function ϱ^{PF} which can be computed as a sum of pairwise terms, again abandoning any consideration of the relative orientation of bonds. The effective density, which holds a similar interpretation to the electronic density, is thus written in pair functionals in the form

$$\varrho_\alpha^{\mathrm{PF}} = \sum_{\substack{\beta \\ \beta \neq \alpha}} g_2(r^{\alpha\beta}), \tag{2.23}$$

where g_2 depends only on a single bond length at a time. Consequently, the energy $\mathcal{V}$ is given by

$$\mathcal{V} = \frac{1}{2} \sum_{\substack{\alpha,\beta \\ \alpha \neq \beta}} \mathcal{V}_2(r^{\alpha\beta}) + \sum_\alpha U_\alpha[\varrho_\alpha^{\mathrm{PF}}]. \tag{2.24}$$

2.2.3 Cluster potentials

Another option to incorporate interactions between bonds is simply to include higher-order contributions to the energy in (2.21). As a specific example, a three-body cluster potential takes the form

$$\mathcal{V} = \frac{1}{2} \sum_{\substack{\alpha,\beta \\ \alpha \neq \beta}} \mathcal{V}_2(r^{\alpha\beta}) + \frac{1}{3!} \sum_{\substack{\alpha,\beta,\gamma \\ \alpha \neq \beta \neq \gamma}} \mathcal{V}_3(r^{\alpha\beta}, r^{\alpha\gamma}, r^{\beta\gamma}). \tag{2.25}$$

With the presence of the function $\mathcal{V}_3$, which possesses greater arity than the pair function $\mathcal{V}_2$, cluster potentials include a dependence on the angles formed between bonds, which are completely geometrically determined by the triplet of distances which includes the length of each bond and the length which separates their ends.

2.2.4 Cluster functionals

Cluster potentials may be generalized to *cluster functionals* in the same way that pair potentials are generalized to pair functionals. The final stratum of EPs we consider again incorporate an embedding energy functional, but now rather than confining it to depend only on interatomic distances in a pairwise manner, we again utilize a multibody expan-

sion. Again considering the specific case involving up to three-body contributions, we write

$$\mathcal{V} = \frac{1}{2} \sum_{\substack{\alpha,\beta \\ \alpha \neq \beta}} \mathcal{V}_2(r^{\alpha\beta}) + \frac{1}{3!} \sum_{\substack{\alpha,\beta,\gamma \\ \alpha \neq \beta \neq \gamma}} \mathcal{V}_3(r^{\alpha\beta}, r^{\alpha\gamma}, r^{\beta\gamma}) + \sum_{\alpha} U_\alpha[\varrho_\alpha^{\mathrm{CF}}], \tag{2.26}$$

where the effective density ϱ^{CF} is now given by

$$\varrho_\alpha^{\mathrm{CF}} = \sum_{\substack{\beta \\ \alpha \neq \beta}} g_2(r^{\alpha\beta}) + \sum_{\substack{\beta,\gamma \\ \alpha \neq \beta \neq \gamma}} g_3(r^{\alpha\beta}, r^{\alpha\gamma}, r^{\beta\gamma}). \tag{2.27}$$

In later chapters, we will consider examples of a specific class of cluster functionals known as *bond order potentials*. A caveat to be mentioned in this regard is that, while these EPs may be written in the form above, doing so results in a sacrifice of their physical interpretability. However, for simplicity, the form of (2.26) and (2.27) is nevertheless useful as a general means of identifying EPs which cannot be described by one of the preceding categories just discussed.

2.2.5 Partitions of the energy

A critical feature which sets EPs apart from first-principles methods is that they posit locality of the energy, which is borne out in the functions which are subject to the summations performed above. In EPs, the functions $\mathcal{V}_2$, $\mathcal{V}_3$, ..., g_2, g_3, ... are always assigned forms which decay for large values of the bond lengths which constitute their arguments. The actual summations are accordingly carried out on subsets of the bonds of the system by forming *neighbor lists*. For example, rather than summing over every possible pair of nuclei α and β in (2.22), a neighbor list is formed for each atom α which consists only of those atoms β for which $r^{\alpha\beta}$ does not exceed a *cutoff distance* r_{cut}. Beyond r_{cut}, the function $\mathcal{V}_2(r)$ is approximately zero, yielding any such contribution negligible in measure compared to the final value of $\mathcal{V}$ calculated.[9] This locality is, in fact, the most appealing aspect of EPs because it offers great computational expediency compared to methods such as DFT which perform iterative self-consistent optimizations on quantities which involve all of the nuclear positions of a system simulataneously.

Given the notion of neighbor lists, it is possible to derive from the summations over individual bond pairs, bond triplets, etc., a partial *atomic energy* which is assigned to each atom based on the bonds which it forms as well as the other bonds nearby. We thus rewrite

[9] A caveat involved in this line of thought is that pair functionals and cluster functionals give rise to forces with an effective range of twice their cutoff distance. See [5, Section 5.8.5] for details.

EP class	Atomic energy function ε_α
Pair potential	$\frac{1}{2}\sum_{\substack{\beta \\ \beta\neq\alpha}} \mathcal{V}_2(r_{\alpha\beta})$
Pair functional	$\frac{1}{2}\sum_{\substack{\beta \\ \beta\neq\alpha}} \mathcal{V}_2(r_{\alpha\beta}) + U_\alpha[\varrho_\alpha^{\mathrm{PF}}]$
Three-body cluster potential	$\frac{1}{2}\sum_{\substack{\beta \\ \beta\neq\alpha}} \mathcal{V}_2(r^{\alpha\beta}) + \frac{1}{3!}\sum_{\substack{\beta,\gamma \\ \beta\neq\gamma\neq\alpha}} \mathcal{V}_3(r^{\alpha\beta}, r^{\alpha\gamma}, r^{\beta\gamma})$
Three-body cluster functional	$\frac{1}{2}\sum_{\substack{\beta \\ \beta\neq\alpha}} \mathcal{V}_2(r^{\alpha\beta}) + \frac{1}{3!}\sum_{\substack{\beta,\gamma \\ \beta\neq\gamma\neq\alpha}} \mathcal{V}_3(r^{\alpha\beta}, r^{\alpha\gamma}, r^{\beta\gamma}) + U_\alpha[\varrho_\alpha^{\mathrm{CF}}]$

Table 2.1: One possible set of atomic energy functions ε_α which can be defined for the different general categories of EPs, taking three-body examples for cluster potentials and cluster functionals. The atomic energies of cluster potentials and cluster functionals shown above will generally not be invariant to permuting the indices of the nuclei, and thus are not even natively defined. Furthermore, even for EPs which are natively written in terms of permutationally invariant atomic energies, they are fundamentally non-unique.

the energy $\mathcal{V}$ of a general EP as

$$\mathcal{V} = \sum_\alpha \varepsilon_\alpha. \tag{2.28}$$

One possible set of forms taken by the energy ε_α of atom α across the different strata of EPs outlined in Carlsson's taxonomy is summarized in Table 2.1.

Most EPs which may be found in the literature do not natively define atomic energies as they are formulated. Rather, to ensure that each unique bond energy and bond interaction energy is considered only once, only unique combinations of the nuclear indices are realized in the summations of a typical EP. For two-body terms, this is done by reducing the symmetric full summation to a half summation according to

$$\sum_{\substack{\alpha,\beta \\ \beta\neq\alpha}} \rightarrow \sum_{\substack{\alpha,\beta \\ \beta>\alpha}}, \tag{2.29}$$

while for three-body and higher-order terms, a similar rule is followed. For example,

$$\sum_{\substack{\alpha,\beta,\gamma \\ \alpha\neq\beta\neq\gamma}} \rightarrow \sum_{\substack{\alpha,\beta,\gamma \\ \alpha<\beta<\gamma}} . \tag{2.30}$$

Because of this change in summation, the precise order in which the atoms of a system are indexed is important for cluster potentials and cluster functionals, and the atomic energies ε_α shown in Table 2.1 will not generally be permutationally invariant. Furthermore, even if an EP is written natively in terms of permutationally invariant atomic energies, it is always fundamentally possible to derive alternative sets of atomic energies which are consistent with the total energies predicted by the potential for any atomic configuration, and this indeterminacy will become central to the study we undertake in Chapter 6.

For a general review of empirical potentials, the reader may refer to [42], while specific examples of eight EPs for silicon which will be used in the case study of Chapter 6 may be found in Appendix C. The Stillinger–Weber EP [43] and its parametrization for silicene [44] are three-body cluster potentials, while the remainder are three-body cluster functionals.

Chapter 3

Open Knowledgebase of Interatomic Models (KIM)

In the previous chapter, empirical potentials were introduced as devices which lend computational tractability to the study of dynamical material behavior, a subject which is largely impassable to first-principles calculations at the present juncture. It was noted in Chapter 1, however, that despite the wide adoption of these methods by researchers over the course of decades, their full ability to reveal scientific insight remains unrealized owing to their dissonant instantiation in practice. In response, the Open Knowledgebase of Interatomic Models (OpenKIM, KIM) [10–12] has been established through funding by the National Science Foundation (NSF). OpenKIM is an open-source, community-driven project led by principal investigators Ellad B. Tadmor and Ryan S. Elliott of the University of Minnesota and James P. Sethna of Cornell University. The purpose of OpenKIM is to create an organized framework for the controlled application of empirical potentials to a vast array of materials science problems in such a way as to yield publically accessible, reproducible results. Below, we briefly outline the content which is housed in KIM and its computational infrastructure. Having commented on the ways in which KIM can address the technical problems associated with empirical atomistic modeling, we conclude by identifying its role as a catalyst for studying the precision and transferability of potentials.

3.1 Content in KIM

In this section, we introduce the various algorithms and data structures which are stored as part of the KIM project, collectively referred to as the *OpenKIM Repository* (freely avail-

able at **https://openkim.org**). Every item submitted to the OpenKIM Repository is assigned a unique permanent identifier, enabling thorough data provenance and reproducibility of all results produced therefrom.

3.1.1 Models

As the name implies, a distinguishing feature of the KIM framework is that the empirical potentials it contains are as essential a component of the system as the results they produce. To this end, the most fundamental concept of KIM is that of a *Model*, formally defined in the KIM Requirements Document [45] as "... a computer implementation representing a specific interaction between atoms, e.g. an interatomic potential or force field." The availability of Models for use by researchers spanning academia, national laboratories, and industry settings is beneficial for two reasons. First, it avoids the simultaneous creation of identical empirical potentials by different developers—an obvious waste of scientific resources. Concentrating development efforts to a single source code further acts to efface the inhomogeneous propagation of programming errors, optimizations, and general modifications. Second, because the role of a KIM Model pertains only to an abstract scheme of the workflow of a generic empirical potential, they are designed to function as self-contained modules which can be used portably in a variety of external software environments, as well as within KIM. This conceptualization also allows Models to be written in different languages, with currently supported languages including FORTRAN77/90/95/2003, C, and C++.

3.1.2 Model Drivers

A distinction is made in KIM between an explicit empirical potential which may be used to perform calculations and the programmatic implementation of its functional form. Rather than indiscriminately combining the functional form of a potential with the specific values of its parameters into a single monolithic entity, KIM allows these two complementary aspects to remain independent. KIM Models as defined above are thus permitted to exist either as standalone codes or as individual parametrizations which can be used in conjunction with the code of a *Model Driver*. As an example of the motivation behind this separation, consider the Embedded-Atom Method (EAM) class of empirical potentials [46–48]. Potentials belonging to this family are commonly given in the form of a large table of parameters defining spline functions which are used as part of its overall form. Creating the primary potential code which appropriately handles these parameters and implements the corresponding potential, including efficiently processing neighbor lists and computing the

atomic forces, is laborious and prone to human error. However, because the parameters are always given in a consistent format, this code is, in principle, uniform across the majority of the various EAM potentials found in practice. As such, KIM avoids unnecessary code duplication by providing potential developers the option to reuse the same primary code with each parameter set. Maintaining this separation reaps the same benefits afore described for Models, averting redundant use of development efforts in favor of a single source code which follows a catalogued revision history. Furthermore, the inclusion of new Models in KIM is facilitated by requiring only the submission of a parameter file which meets the proper specifications of the relevant Model Driver.

3.1.3 Tests

By virtue of complementarity, a necessary consequence of establishing an abstract workflow for empirical potentials in the definition of KIM Models is the concurrent definition of an abstract workflow for the simulations which might use them. We correspondingly define a *Test* as "a specific program which when coupled with a suitable Model, possibly including additional input, calculates and returns a specific prediction about a particular configuration." That is, a Test is taken to include all information other than the internal potential function specified by the Model which is necessary to fully define an atomistic simulation. This generally includes the atomic species, positions, and charges, as well as temperature, boundary conditions, and any external potential field present. A KIM Test may consist of custom software written in a number of languages (including compiled and interpreted languages such as Python, C, C++, FORTRAN77/90/95/2003, and common GNU shells) by an individual research group to study specific problems in materials science. A convenient alternative for Test creation is that they are also allowed to take the form of a simple input script which can be used with a supported simulation package. Currently supported simulation software includes ASAP [13], ASE [14], DL_POLY [15], GULP [16], IMD [17], LAMMPS [18], and libAtoms+QUIP [19].

3.1.4 Test Drivers

Tests can be generalized in much the same as way as Models are generalized to Model Drivers. Albeit Tests may consist of autonomous code, they are also permitted to exist as parametrizations which are used by a more general *Test Driver*. For instance, a Test Driver may be a source code which when supplied with a parameter file containing an atomic species and lattice geometry by a Test, calculates the corresponding phonon dispersion curves using a given Model.

3.1.5 Property Definitions

A requisite aspect of pairing a Model and Test to perform a simulation is a well-defined concept of what the simulation output is, i.e. the definition of one or more material properties. In the KIM ecosystem, these are termed *Property Definitions*, and exist as data structures containing member elements which each serve to specify a physically meaningful quantity. An example of a Property Definition might be one intended to indicate the bulk modulus of an arbitrary crystal lattice at a specific temperature. Such a definition would thus contain data members which specify the lattice geometry and atomic species, the temperature, and the relevant boundary conditions alongside the isothermal bulk modulus itself. For simplicity, Property Definitions are defined on a case-by-case basis without any stringent requirements imposed on their structure. However, in creating Property Definitions, users are encouraged to reuse any individual data members which coincide with existing Property Definitions. For example, if all existing Property Definitions related to cubic crystal lattices specify a data member named '`a`' to indicate the conventional lattice constant, any appurtenant Property Definitions added to the OpenKIM Repository should adhere to the same convention.

3.1.6 Predictions

The result of a Test-Model coupling itself is known as a *Prediction* in the terminology of KIM. A Prediction thus corresponds to an instance of a Property Definition (a *Property Instance*) which contains specific numerical values as output by a Test when utilizing a particular Model. For more on Property Definitions and Predictions, the reader is referred to **https://openkim.org/properties-framework**.

3.1.7 Reference Data

Reference Data is comprised of "...experimental or first principles results (e.g. density functional theory (DFT) results) for a material property to which Predictions can be compared" [45]. In order to facilitate comparison with Predictions, Reference Data is also stored in the form of Property Instances.

3.2 KIM API

The precise way in which Models and Tests are abstracted in KIM is defined by the KIM *Application Programming Interface* (KIM API). As alluded to above, the KIM API is designed to treat Models as encapsulated objects which explicitly indicate the input they

require from a Test and the output they transmit back to a Test in an obligatory *KIM descriptor file*. At a minimum, a typical Model will specify as its required input its supported element(s) and the atomic positions, as well as the types of neighbor lists and boundary conditions it supports. As output, it will provide the total energy and the atomic forces, and in some cases additional information such as the virial stress. In turn, a Test will indicate the quantities which it expects to be returned by a Model and what information it consistently provides to a Model in its own KIM descriptor file. Altogether, if the inputs and outputs specified in the descriptor files of both a Model and a Test are mutually compatible, the two are capable of being used together to perform a calculation. This is true regardless of whether or not they were written in the same programming language because the KIM API transmits all information in a generic data object and makes all necessary interlanguage adjustments automatically.

The vast majority of Tests in KIM encompass calculations such as static minimization of the energy with respect to the atomic positions or their time integration under a suitable thermodynamical ensemble. Situations such as these require the ongoing exchange of the atomic positions contained within a Test and the energy and atomic forces computed by a Model for each collection of atomic positions subject to the presiding boundary conditions. Thus, the role of the KIM API in most cases is to supply a communication channel over which the inputs and outputs of a Test are iteratively reciprocated with those of a Model. While this mediation incurs its own computational cost, preliminary benchmarking of Lennard-Jones [49–52] and EAM [53, 54] Models within LAMMPS indicates that the reduction in parallel efficiency compared to its natively implemented versions of these potentials does not exceed ten percent when using 256 cores or less [55]. This overhead, which may be reduced over time with additional optimization of the KIM API and the Models held within KIM, is sufficiently mitigated in most cases by the advantages offered through the use of the KIM framework outlined above.

The KIM API, as well as further information about it, can be found at
https://openkim.org/kim-api.

3.3 Processing Pipeline

The process of running Test-Model couplings is orchestrated by the *OpenKIM Processing Pipeline*, an extensible cloud-based computing resource. When a new Model (Test) is uploaded to the OpenKIM Repository, the OpenKIM Pipeline compares its descriptor file to those of all existing Tests (Models). Every compatible pairing is then executed using a

distributed set of computing agents which provide suitably normalized benchmarks which can be used to compare the run times of different Test-Model pairs. When execution has completed, the Pipeline ensures that the Prediction(s) supplied by the Test at the end of the calculation conform to one or more existing Property Definition(s). If each Prediction rendered by the Test is found to be compliant, they are inserted into the publically queryable KIM database (**https://query.openkim.org**), as well as provided with their own page in the OpenKIM Repository which additionally includes links to any auxiliary files created by the Test, e.g. atomic trajectories which can be used for visualization.

In keeping with the principles of encapsulation and modularity which pervade the KIM philosophy, the calculation of material properties which can foreseeably be used in a variety of contexts is typically relegated to dedicated Tests. For example, the zero-temperature equilibrium lattice constant predicted by a Model for a given material and crystal structure is a primitive prerequisite to the computation of many other properties a Model is used to predict. A Test which uses a Model for aluminum to compute the relaxed formation energy of a point defect, e.g. a monovacancy in a face-centered cubic (fcc) lattice, requires that the associated equilibrium lattice constant be known. However, the assignment of unique, permanent identifiers to Predictions in KIM enables an efficient inheritance of information in this regard. Rather than estimating the lattice constant itself, it is more prudent for such a Test to query the Prediction of the fcc lattice constant for the relevant Model which has already been computed by a separate Test. However, this introduces a complication from the perspective of the Pipeline because it establishes a definite order in which Test-Model pairs must be run. This correlation is handled by allowing each KIM Test to provide a *dependencies file*, which is utilized by the Pipeline to resolve the sequence of execution which must be followed. Tests which have no dependencies file are run first, whereas Tests which require as input the Predictions of other Test-Model pairs are run only after these are completed and made available for reference in the KIM database.

3.4 Application

The foremost purpose of KIM is to provide ready access to standardized implementations of empirical potentials and simulation codes whose provenance is precisely recorded. This endeavor presents a means to obviate the irreproducibility of the results of scientific investigations in the domain of empirical atomistics while simultaneously facilitating their conduct by eliminating the redundant expenditure of resources. However, a concomitant of this pursuit is the generation of a large mass of data which can be used to learn about the Models which reside in KIM. The fact that every compatible Model and Test are paired

and executed by the Pipeline begets a sizeable number of Predictions which populate the OpenKIM Repository. Because of the conjugate relationship between Models and Tests, this information can be leveraged in two unique ways.

In the first mode of analysis, the Predictions of a single Test can be examined for a large number of Models. This allows trends among different classes of Models, e.g. the categories of potentials discussed in Chapter 2 or those fit to specific material properties, to be exposed. The same methodology can also be applied to an ensemble of Models which correspond to a single base potential but differ by a perturbation which is applied to its parameters. The range of variation observed in the Predictions of such an ensemble then indicates the *precision* of the base Model. The guiding principle of the notion of precision is that a high degree of sensitivity in a Prediction (corresponding to a material property) to the parameters of a Model disputes its validity in computing that specific property. That is, Models which demonstrate low precision in computing a particular material property are less plausible than Models whose corresponding Predictions are highly precise. This vein of thought, first communicated by Frederiksen *et al.* [56], thus adopts a distinctly Bayesian perspective of empirical potentials. Rather than thinking of a Model as a functional form accompanied by a single set of parameters deemed optimal in some sense, a purportedly more meaningful characterization is to consider it as a functional form which is complemented by a distribution of its various parameters. Admitting this interpretation reveals new prospects in the analysis of empirical potentials, including how fitting databases may be optimally constructed [57].

Conversely, in the second mode of analysis, the Predictions of a single Model can be inspected for a multitude of Tests. Comparing the results to Reference Data indicates the accuracy of the Model in computing an array of material properties. While a Model may be expected to sufficiently reproduce the properties to which it was fit, its accuracy in computing other properties gives rise to the concept of its *transferability*. Creating transferable potentials is, at least pragmatically, the ultimate goal of empirical atomistics. Moreover, it is not unreasonable to assert that a Model which is highly transferable across the configuration space of a specific material is likely to contain essential features of its governing mechanics. However, despite the fact that the transferability of empirical potentials is perhaps the most important feature of their development and application, there remains no rigorous, quantitative method of its assessment. The remainder of this manuscript seeks to explicate the ways in which the abundance of Predictions of a Model which exist in KIM can be used alongside comparable Reference Data in order to produce an empirical quantitative measure of its transferability.

Chapter 4

Representation of Atomic Environments

The basic operational units of an empirical interatomic potential are the potential energies it predicts for individual atomic environments, which subsequently yield the individual atomic forces and the total energy of the system. As such, selecting a proper representation of these environments is critical in designing an empirical potential. Moreover, atomic neighborhood representations are useful in other applications, including structure identification and prediction. The simplest class of representations is directly based on quantities of the real-space positions of the atoms which are invariant to global translations and rotations. If the line segments joining pairs of atoms are interpreted as chemical bonds, these quantities correspond to the lengths of atomic bonds and the angles between them. After first exploring several important descriptors of atomic environments which are carried out in real space, we move on to consider representations which are derived from a spectral analysis of the real-space positions by means of Fourier transforms. The chapter concludes with remarks on possible courses of defining the similarity between two arbitrary atomic environments. It should be noted that, throughout this chapter, we limit our study to monoatomic systems for simplicity, but the representations of atomic environments discussed within may be extended to multiatomic systems with modest difficulty.

4.1 Real-space representations

4.1.1 Bond lengths and bond angles

We begin from the most primitive possible representation of the geometry of an atomistic system. Excluding the possibility of unlike species of the atoms or the presence of charge states, this amounts to nothing more than the real-space coordinates of the atoms in some

coordinate system which, for simplicity, is assumed to be Cartesian. If the coordinates of atom α are contained in the vector $\boldsymbol{r}^\alpha$ and the system consists of N atoms labeled by indices $\alpha = 1, 2, \ldots, N$, this information is contained in the set of vectors $\{\boldsymbol{r}^1, \boldsymbol{r}^2, \ldots, \boldsymbol{r}^N\}$. We refer to this description of a system of atoms as an *atomic configuration*. Recall from Chapter 2 that the total internal potential energy $\mathcal{V}$ of the N-atom system can, in principle, be written to close approximation in terms of these vectors as $\mathcal{V} = \mathcal{V}(\boldsymbol{r}^1, \boldsymbol{r}^2, \ldots, \boldsymbol{r}^N)$. An important feature inherent to $\mathcal{V}$ which was briefly mentioned in this discussion is the *principle of material frame-indifference* [5]. This principle, to which all physical models we consider must adhere, states that the total internal potential energy $\mathcal{V}$ (and, hence, all quantities derived from it) must be invariant with respect to proper global rotations and translations of the system. That is, we require

$$\mathcal{V}(\boldsymbol{Q}\boldsymbol{r}^1 + \boldsymbol{c}, \boldsymbol{Q}\boldsymbol{r}^2 + \boldsymbol{c}, \ldots, \boldsymbol{Q}\boldsymbol{r}^N + \boldsymbol{c}) = \mathcal{V}(\boldsymbol{r}^1, \boldsymbol{r}^2, \ldots, \boldsymbol{r}^N) \tag{4.1}$$

for all $\boldsymbol{Q} \in \mathrm{SO}(3)$ and $\boldsymbol{c} \in \mathbb{R}^3$. The principal consequence of (4.1), originally given in [58] and discussed further in [59], is that $\mathcal{V}$ can be written solely in terms of the distances between the atoms. If the pairwise distances between the N atoms of the system are collected into a symmetric $N \times N$ matrix $\boldsymbol{\mathcal{D}}$ of the form

$$\boldsymbol{\mathcal{D}} = \begin{bmatrix} \mathbf{0} & \|\boldsymbol{r}^1 - \boldsymbol{r}^2\| & \cdots & \|\boldsymbol{r}^1 - \boldsymbol{r}^N\| \\ \|\boldsymbol{r}^2 - \boldsymbol{r}^1\| & \mathbf{0} & \cdots & \|\boldsymbol{r}^2 - \boldsymbol{r}^N\| \\ \vdots & \vdots & \ddots & \vdots \\ \|\boldsymbol{r}^N - \boldsymbol{r}^1\| & \|\boldsymbol{r}^N - \boldsymbol{r}^2\| & \cdots & \mathbf{0} \end{bmatrix}, \tag{4.2}$$

then we may write $\mathcal{V} = \hat{\mathcal{V}}(\boldsymbol{\mathcal{D}})$ where $\hat{\mathcal{V}}$ is, formally, a new function which takes as input a matrix as is defined in (4.2) rather than a set of vectors. If the elements of $\boldsymbol{\mathcal{D}}$ are interpreted as chemical bonds between the atoms, (4.1) can be expressed by saying that $\mathcal{V}$ depends only on the *bond lengths*. Thus, in practice, we may write $\mathcal{V}$ in terms of the bond lengths of the system and be guaranteed to satisfy (4.1).

As previously mentioned in Chapter 2, empirical potentials are based on the notion of locality in the sense that it is assumed that $\mathcal{V}$ can be written as

$$\mathcal{V} = \sum_{\alpha=1}^{N} \varepsilon_\alpha, \tag{4.3}$$

where the quantity ε_α can be interpreted as an atomic energy that is assigned to atom α,[1] and

[1] We emphasize that the atomic energies ε_α will not generally be unique due to the arbitrary indexing of

depends only on the positions of atom α and those atoms which lay nearby in space. This notion gives rise to the important concept of an *atomic environment*, which will serve as a fundamental building block for the remainder of this chapter. An atomic environment (also referred to as an *atomic neighborhood*) consists of a *target atom* along with its *neighbors*, which are defined to be those atoms which fall within some cutoff distance of the target atom. Because $\mathcal{V}$ is a function only of $\boldsymbol{\mathcal{D}}$, the form of (4.3) implies that ε_α is itself a function only of the subset of $\boldsymbol{\mathcal{D}}$ which includes information about atom α and its neighbors. While bond lengths have historically been used with empirical potentials in many instances[2] they are not always the most intuitive choice and, in practice, another set of features is derived directly from the bond lengths: the *bond angles*. Although the bond angles contain no information beyond that which is already contained in the bond lengths, they provide a natural medium within which an atomic neighborhood may be defined. For instance, it is well known that bulk silicon favors tetrahedral bond angles at standard temperature and pressure.

When we speak of "the bond lengths and bond angles" of the neighborhood of a target atom α, we refer to the distances between atom α and its neighbors, as well as the angles formed between the line segments which connect atom α to its neighbors. To provide clarity in notation, Figure 4.1 shows the relative position vectors and bond lengths/angles of an arbitrary two-dimensional atomic neighborhood. Note that we now transition to using subscripts of the form $\boldsymbol{r}_\beta$ to indicate the position of neighbor β relative to the target atom, and conventionally label the target atom with an index of 0 and its N_{neigh} neighbors with indices $\beta = 1, \ldots, N_{\text{neigh}}$.[3] Regular typeface of the form r_β is used to indicate the magnitude of these vectors, i.e. the bond lengths, while bond angles are given in the form $\theta_{\beta\gamma}$ (and it is assumed that $0 \leq |\theta_{\beta\gamma}| \leq \pi$). Not pictured in the Figure 4.1, the distance between neighbors β and γ is denoted $r_{\beta\gamma}$.

The cumulative geometric information of an atomic environment used to determine a cor-

---

the atoms in a configuration.

[2]Aside from the obvious case of the pair potentials discussed in Chapter 2, which depend only on bond lengths, the *Coulomb Matrix* $\mathbf{M}$ defined by

$$[\mathbf{M}]_{\alpha\beta} = \begin{cases} 0.5 Z_\alpha^{2.4} & \text{for } \alpha = \beta \\ \frac{Z_\alpha Z_\beta}{||\boldsymbol{r}^\alpha - \boldsymbol{r}^\beta||} & \text{for } \alpha \neq \beta \end{cases} \tag{4.4}$$

(where Z_α represents the atomic number of atom α) has also been successfully used to define empirical potentials [60].

[3]In the sequel, where a plethora of indices is required to delineate the mathematical entities which we will use to describe atomic environments, we will continue to refer to the actual *atoms* of the system by the Greek letters α, β, γ.

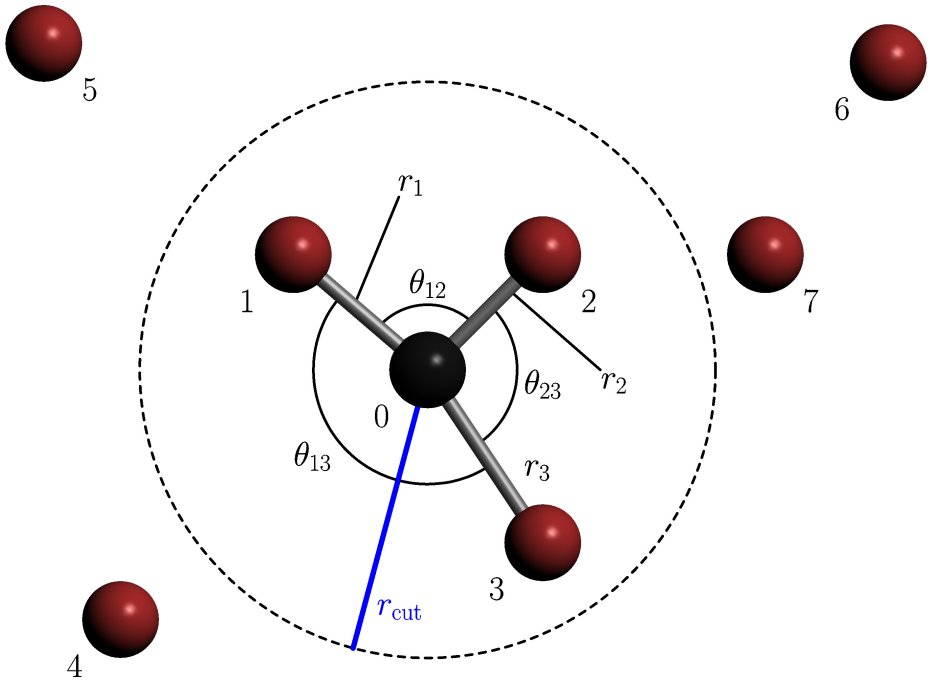


Figure 4.1: The bond lengths and bond angles of an arbitrary two-dimensional atomic neighborhood of a target atom with $N_{\text{neigh}} = 3$ neighbors. The target atom is taken to have an index of 0, while its neighbors are indexed 1–3; to guide the eye, grey "bonds" have been drawn connecting the target atom to each of its neighbors. The bond lengths shown are denoted r_1, r_2, and r_3, while the bond angles are denoted θ_{12}, θ_{23}, and θ_{13}. Note that atoms 4–7 are outside of the cutoff distance r_{cut}, and are thus not considered neighbors of the target atom.

responding atomic energy ε_α can be encompassed by simultaneously considering all of its individual geometric parameters, e.g. the bond lengths and bond angles between the target atom and its neighbors. This more global definition of an atomic environment is henceforth referred to as a *descriptor*. Our first example of a descriptor is due to a construction similar to (4.2) which was discussed in relation to invariants of a collection of vectors (over the field of real numbers) to the full orthogonal group by Weyl in his well-known text, *The Classical Groups: Their Invariants and Representations* [61, Ch. II, Sec. 9]. This descriptor, which we term a *Weyl matrix* and denote by $\boldsymbol{\Sigma}$, following the convention of Bartók *et al.* [62], is defined as a Gram matrix of the form

$$\boldsymbol{\Sigma} \triangleq \begin{bmatrix} \boldsymbol{r}_1 \cdot \boldsymbol{r}_1 & \boldsymbol{r}_1 \cdot \boldsymbol{r}_2 & \cdots & \boldsymbol{r}_1 \cdot \boldsymbol{r}_{N_{\text{neigh}}} \\ \boldsymbol{r}_2 \cdot \boldsymbol{r}_1 & \boldsymbol{r}_2 \cdot \boldsymbol{r}_2 & \cdots & \boldsymbol{r}_2 \cdot \boldsymbol{r}_{N_{\text{neigh}}} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{r}_{N_{\text{neigh}}} \cdot \boldsymbol{r}_1 & \boldsymbol{r}_{N_{\text{neigh}}} \cdot \boldsymbol{r}_2 & \cdots & \boldsymbol{r}_{N_{\text{neigh}}} \cdot \boldsymbol{r}_{N_{\text{neigh}}} \end{bmatrix}. \tag{4.5}$$

Evidently, the diagonal components of $\boldsymbol{\Sigma}$ contain the (squared) bond lengths $\boldsymbol{r}_\beta \cdot \boldsymbol{r}_\beta = ||\boldsymbol{r}_\beta||^2$ of the neighbors of the target atom, while the off-diagonal components contain the bond angles in the form $\boldsymbol{r}_\beta \cdot \boldsymbol{r}_\gamma = r_\beta r_\gamma \cos\theta_{\beta\gamma}$. One might initially think that this descriptor suffices to quantify atomic neighborhoods in a manner which is practically conducive to empirical potentials. Because the vectors $\boldsymbol{r}_\beta$ are expressed in a basis which is always centered at the target atom, global translations of the neighborhood have no bearing on $\boldsymbol{\Sigma}$. Moreover, $\boldsymbol{\Sigma}$ is invariant to orthogonal transformations since it consists of bond lengths and bond angles, and thus proper rotations also have no effect on it. However, there is one additional invariance which is lacking in this descriptor: *permutational invariance*. This invariance, which is entirely artificial in nature, relates to how the neighbors are indexed in order to compute a descriptor. Permuting this labeling has the result of interchanging rows and columns in $\boldsymbol{\Sigma}$. For instance, for the neighborhood shown in Figure 4.1, the Weyl matrix is written as

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{r}_1 \cdot \boldsymbol{r}_1 & \boldsymbol{r}_1 \cdot \boldsymbol{r}_2 & \boldsymbol{r}_1 \cdot \boldsymbol{r}_3 \\ \boldsymbol{r}_2 \cdot \boldsymbol{r}_1 & \boldsymbol{r}_2 \cdot \boldsymbol{r}_2 & \boldsymbol{r}_2 \cdot \boldsymbol{r}_3 \\ \boldsymbol{r}_3 \cdot \boldsymbol{r}_1 & \boldsymbol{r}_3 \cdot \boldsymbol{r}_2 & \boldsymbol{r}_3 \cdot \boldsymbol{r}_3 \end{bmatrix}. \tag{4.6}$$

Now, if the labels of atoms 1 and 2 are interchanged, $\boldsymbol{\Sigma}$ becomes (in terms of the original labeling notation)

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{r}_2 \cdot \boldsymbol{r}_2 & \boldsymbol{r}_2 \cdot \boldsymbol{r}_1 & \boldsymbol{r}_2 \cdot \boldsymbol{r}_3 \\ \boldsymbol{r}_1 \cdot \boldsymbol{r}_2 & \boldsymbol{r}_1 \cdot \boldsymbol{r}_1 & \boldsymbol{r}_1 \cdot \boldsymbol{r}_3 \\ \boldsymbol{r}_3 \cdot \boldsymbol{r}_2 & \boldsymbol{r}_3 \cdot \boldsymbol{r}_1 & \boldsymbol{r}_3 \cdot \boldsymbol{r}_3 \end{bmatrix}, \tag{4.7}$$

where the first two rows and first two columns have been interchanged. Because the indexing of the neighbors of an atomic environment is stochastic in practical implementation, it is essential for any descriptor to satisfy permutational invariance. This leads us to consider other classes of descriptors which are functions of the bond lengths and bond angles, but which are invariant to permutations of the neighbor indices, in addition to Euclidean motions.

4.1.2 Behler–Parrinello Symmetry Functions

The most elementary non-trivial mathematical operation which is invariant to permutation of its terms is a finite summation. This basic property gives rise to a new category of atomic descriptors which will comprise the remainder of those we consider. Being descriptors, these objects represent an atomic environment by means of a single set of quantities which are collected into an array, but they are distinguished by the fact that all of their terms are computed by summing over every neighboring atom. In this sense, these *additive*

descriptors may be understood as interpreting an atomic environment as a global distribution rather than treating its individual constituents separately. This distinction is precisely what determines the *order* of a descriptor. For example, the Weyl matrix of the previous section is a third-order descriptor of an atomic environment, considering at most three elements of the neighborhood simultaneously (including the target atom) in the bond angles contained in its off-diagonal terms. By contrast, an additive descriptor has no limit to its order; regardless of how many neighboring atoms are contained in an atomic environment, an additive descriptor is always calculated by summing over all of them. It may thus be helpful to think of additive descriptors as having infinite order, although this is manifestly never realized in practice.

The first set of additive descriptors we consider is attributed to Behler and Parrinello [63], who refer to them as "atom-centered symmetry functions," and which we will refer to here as *Behler–Parrinello symmetry functions*. These descriptors, denoted G_α^k ($k = 1, \ldots, 5$) for target atom α, are further divided into two sets: radial symmetry functions and angular symmetry functions. We begin by defining a cutoff function $f_{\text{cut}}(r)$, shown in Figure 4.2:

$$f_{\text{cut}}(r) = \begin{cases} \frac{1}{2}\left[\cos\left(\frac{\pi r}{r_{\text{cut}}}\right) + 1\right] & \text{for } r \leq r_{\text{cut}} \\ 0 & \text{for } r > r_{\text{cut}}. \end{cases} \tag{4.8}$$

There are several important features of this cutoff function which may be readily observed: (1) it is monotonically decreasing, (2) it decays from unity to zero at the cutoff distance r_{cut}, and (3) this decay is smooth. The role of any cutoff function is to enforce the locality of a descriptor by serving as a modulating factor by which the various terms involved are multiplied. With this purpose in mind, a cutoff function is designed to satisfy (1) in order to assign higher influence to atoms which are close to the target atom than those which are far away. Property (2) ensures that atoms which are extremely close to the target atom contribute terms to the descriptor which are almost unaffected by the modulation of the cutoff function, while atoms which are beyond the cutoff distance have no contribution. Finally, property (3) is important because it promises the cutoff function is differentiable. For a generic function $\mathcal{F}$ which depends on the descriptor, this prevents the cutoff function from introducing unwanted discontinuities in either $\mathcal{F}$ or its derivatives. Furthermore, it is desirable for the derivatives of the descriptor to tend to zero at the cutoff radius. This implies that there are no sudden jumps in the value of $\mathcal{F}$ or its derivatives as atoms enter or leave the cutoff radius, which frequently occurs in simulation. The process of atoms entering or leaving the atomic environment of a target atom in the course of simulation

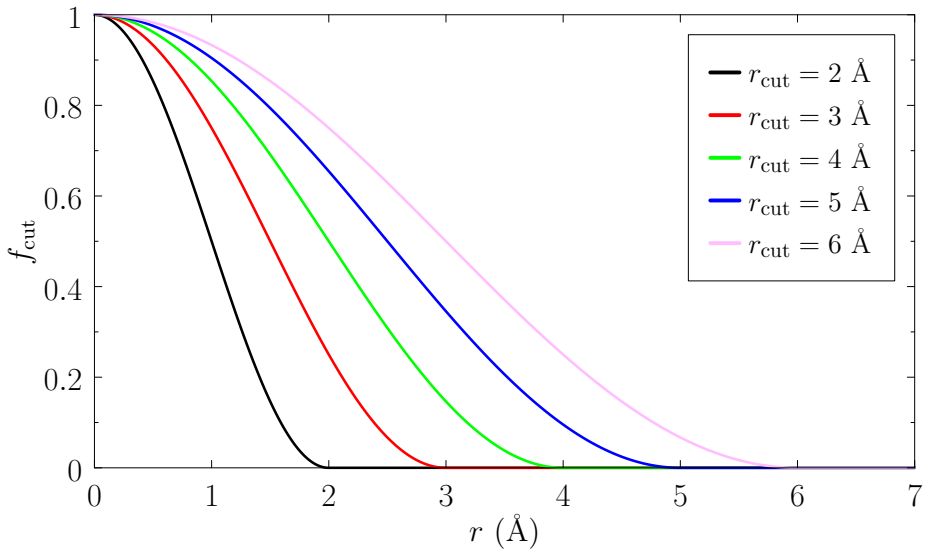


Figure 4.2: Cutoff function used in the Behler–Parrinello symmetry functions, displayed for different choices of the radial cutoff distance r_{cut}.

also underscores one final invariance whose importance will be revealed in Chapter 5. This *dimensional invariance* requires that the dimensionality of a descriptor remain unchanged regardless of the number of neighbors in the environment and, once more, the mathematical operation of summation proves to be efficacious in this regard.

Using this cutoff function, the radial symmetry functions are defined as

$$G_\alpha^1 \triangleq \sum_{\beta=1}^{N_{\text{neigh}}} f_{\text{cut}}(r_\beta), \tag{4.9}$$

$$G_\alpha^2(\eta, R_s) \triangleq \sum_{\beta=1}^{N_{\text{neigh}}} e^{-\eta(r_\beta - R_s)^2} f_{\text{cut}}(r_\beta), \tag{4.10}$$

$$G_\alpha^3(\kappa) \triangleq \sum_{\beta=1}^{N_{\text{neigh}}} \cos(\kappa r_\beta) f_{\text{cut}}(r_\beta). \tag{4.11}$$

At first, it may seem unnecessary to introduce the function G_α^2 since the exponential term decays in a fashion similar to the cutoff function. However, the introduction of the exponential term serves two purposes. First, the dependence on η presents a method to circumvent the sensitivity of the descriptor to the choice of cutoff radius, which can drastically alter its discriminative power. Rather than recomputing the descriptor for different choices of cutoff radii, it is more computationally efficient to instead compute only one set of neighbor lists and tailor the radial dependence of the descriptor by considering different choices

of η. Second, the parameter R_s allows the function G_α^2 to probe different radial "shells" which may be relevant to identifying bond formation or bond types. Figure 4.3 shows the dependence of G_α^2 on η and R_s for an atomic environment with a single neighbor as the distance r_β is varied. Finally, we remark that, as the author warns [63], the last of these functions must be used with great caution, since the cosine term can take either negative or positive values depending on the distance of neighbor β from the target atom. These contributions can conceivably cancel one another out, and G_α^3 is only suggested for potential use in combination with other radial or angular symmetry functions. Therefore, we will neglect this function from now on.

Finally, the angular symmetry functions are:

$$G_\alpha^4(\eta,\lambda,\zeta) \triangleq 2^{1-\zeta} \sum_{\beta=1}^{N_{\text{neigh}}} \sum_{\gamma>\beta}^{N_{\text{neigh}}} \Big[(1+\lambda\cos\theta_{\beta\gamma})^\zeta e^{-\eta(r_\beta^2+r_\gamma^2+r_{\beta\gamma}^2)} \times f_{\text{cut}}(r_\beta) f_{\text{cut}}(r_\gamma) f_{\text{cut}}(r_{\beta\gamma}) \Big], \tag{4.12}$$

$$G_\alpha^5(\eta,\lambda,\zeta) \triangleq 2^{1-\zeta} \sum_{\beta=1}^{N_{\text{neigh}}} \sum_{\gamma>\beta}^{N_{\text{neigh}}} \Big[(1+\lambda\cos\theta_{\beta\gamma})^\zeta e^{-\eta(r_\beta^2+r_\gamma^2)} f_{\text{cut}}(r_\beta) f_{\text{cut}}(r_\gamma) \Big]. \tag{4.13}$$

Here, λ takes a value of ± 1 and determines whether the minima of the angular terms in these functions occur at $\theta_{\beta\gamma} = 0$ or $\theta_{\beta\gamma} = \pi$, while ζ controls the angular sensitivity (see Figure 4.3). The only essential difference between G_α^4 and G_α^5 is that the latter includes dependence on $r_{\beta\gamma}$, the distance between the two neighbors which form the bond angle $\theta_{\beta\gamma}$ with the target atom. While in G_α^4, neighbors β and γ must fall within the distance r_{cut} of one another to even make a contribution, G_α^5 is indifferent to their separation.

4.1.3 Angular Fourier Series

As a final alternative, the bond lengths and bond angles may be directly expanded in a suitable basis of functions. It was suggested by Bartók *et al.* [62] that the angular information of the neighbors of atom α be represented in the basis of Chebyshev polynomials $T_l(\theta) = \cos(l\theta)$ (where $l = 0, 1, \ldots$), which form a complete orthogonal basis for $L^2([-1,1])$. The radial information can then be embodied by an orthonormal basis of radial functions $g_n(r)|_{n=1}^\infty$. In order to guarantee smoothness of the representation for neighbors near the cutoff distance, the authors use cubic and higher-order polynomials

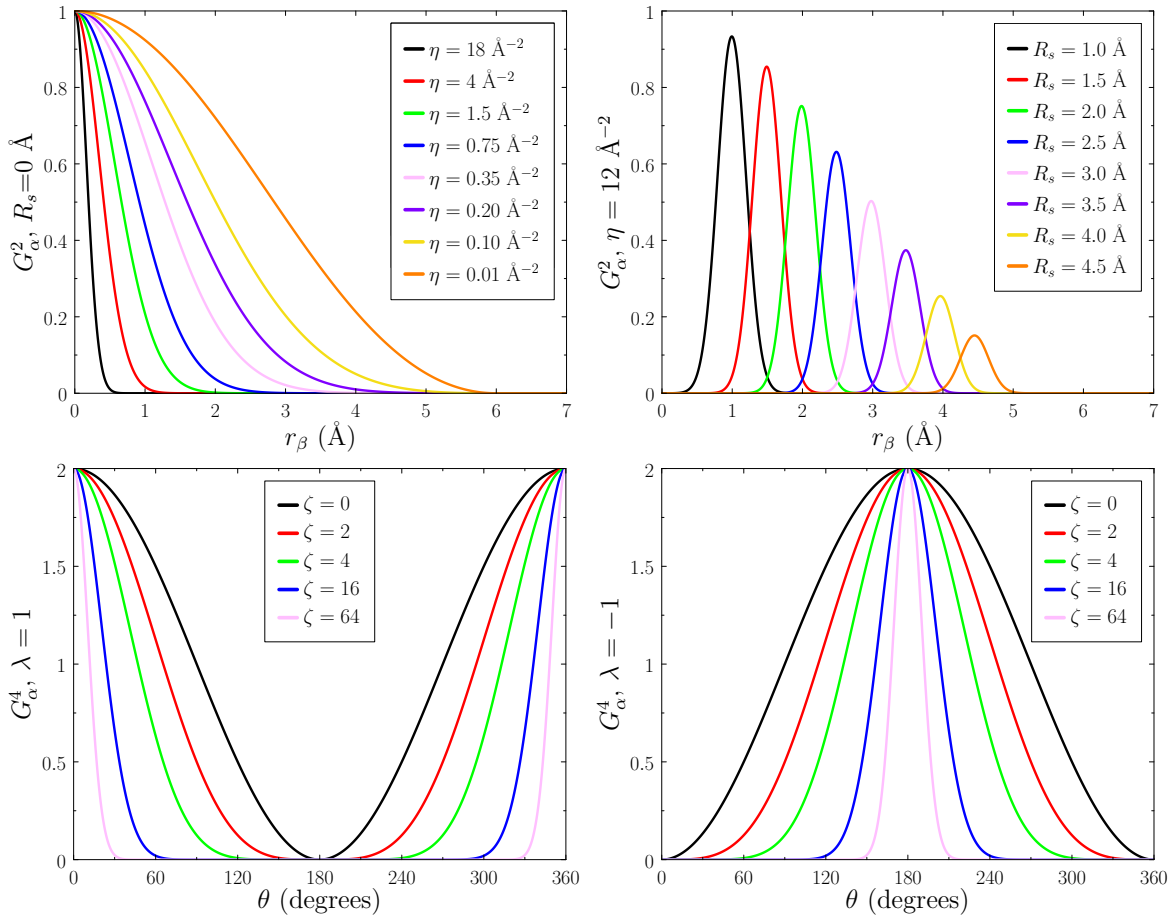


Figure 4.3: (Top) Parameter dependence of radial symmetry function G^2_α for a target atom with a single neighbor β whose distance r_β is varied. (Bottom) The angular portion $2^{1-\zeta}(1+\lambda\cos\theta)^\zeta$ of the symmetry functions G^4_α and G^5_α. Sensitivity of these functions is increased as ζ is increased, resulting in a narrower set of angles which result in non-zero contributions.

$\phi_p(r) = (r_{\text{cut}} - r)^{p+2}/N_p$ for $p = 1, \ldots, n_{\max}$, where

$$N_p = \sqrt{\int_0^{r_{\text{cut}}} (r_{\text{cut}} - r)^{2(p+2)} dr} = \sqrt{\frac{{r_{\text{cut}}}^{2p+5}}{2p+5}}\,. \tag{4.14}$$

Defining the overlap of the functions ϕ_p and ϕ_q as the L^2 inner product,

$$S_{pq} = \int_0^{r_{\text{cut}}} \phi_p(r)\phi_q(r)dr = \frac{\sqrt{(5+2p)(5+2q)}}{5+p+q}, \tag{4.15}$$

then admits the formation of an orthonormal radial basis g_n by defining

$$g_n(r) = \sum_{p=1}^{n_{\max}} W_{np}\phi_p(r), \tag{4.16}$$

where $\mathbf{W} = \mathbf{S}^{-1/2}$. The final AFS descriptor of atom α is then given by

$$\mathrm{AFS}_\alpha^{n,l} \triangleq \sum_{\beta=1}^{N_{\mathrm{neigh}}} \sum_{\gamma>\beta}^{N_{\mathrm{neigh}}} g_n(r_\beta) g_n(r_\gamma) \cos\left(l\theta_{\beta\gamma}\right), \tag{4.17}$$

where $n = 1, \ldots, n_{\max}$ and $l = 0, 1, \ldots, \ell_{\max}$ for some finite integer $\ell_{\max}$.

4.2 Spectral representations

We began this chapter with the observation that by writing the total internal potential energy $\mathcal{V}$ in terms of the bond lengths, it would automatically satisfy the principle of material frame-indifference since these quantities are already invariant to rotations and translations. Similarly, if the total potential energy $\mathcal{V}$ is expressed as a sum of atomic energies ε_α, each of which is written in terms of the bond lengths and angles of the neighboring atoms in the associated atomic environment, it will again trivially satisfy frame-indifference. Accordingly, we proceeded to introduce two descriptors which are defined by the summation of functions of the individual bond lengths and bond angles within a neighborhood: the Behler–Parrinello symmetry functions and the AFS. In the remaining sections of this chapter, we consider additive descriptors which, like those aforementioned, innately account for invariance to Euclidean transformations. However, while they are related to the bond lengths/angles, they are not explicitly written in terms of them, and must instead appeal to auxiliary means to establish the requisite invariances.

4.2.1 Geometric moments

The most historically celebrated method of characterizing the shape of a distribution is based on the use of *moment expansions*. In the treatment we consider for the present application, the spatial distribution of the neighbors in an atomic environment is described by a three-dimensional *atomic neighborhood density function*, $\rho(x, y, z)$. Each atom in the neighborhood, potentially including the target atom itself, contributes its mass to the neighborhood density in a localized fashion, while atoms outside of the cutoff radius make no contribution. To begin analyzing this density, we can start by taking its inverse Fourier

transform, which we denote by φ_ρ:

$$\varphi_\rho(\boldsymbol{u}) \triangleq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \exp(i\boldsymbol{u}\cdot\boldsymbol{x})\rho(x,y,z)dxdydz, \tag{4.18}$$

where $\boldsymbol{u} = [u_x, u_y, u_z]^T$ is a spatial frequency vector, $\boldsymbol{x} = [x, y, z]^T$, and $i = \sqrt{-1}$. In the standard terminology used in the probability literature, this is referred to as the *characteristic function* of ρ, and is guaranteed to exist. If we rewrite the exponential in (4.18) by making use of its power series,[4] we can write

$$\varphi_\rho(\boldsymbol{u}) = \sum_{p=0}^{\infty} \frac{i^p}{p!} H_p(\boldsymbol{u}), \tag{4.19}$$

where, as in the work of Lo and Don [64], we have leveraged the multinomial theorem[5] and defined

$$\begin{aligned} H_p(u_x,u_y,u_z) &\triangleq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (\boldsymbol{u}\cdot\boldsymbol{x})^p \rho(x,y,z)dxdydz \\ &= \sum_{\substack{l,m,n\geq 0 \\ l+m+n=p}} \frac{p!}{l!m!n!} u_x^l u_y^m u_z^n \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^l y^m z^n \rho(x,y,z)dxdydz \\ &= \sum_{\substack{l,m,n\geq 0 \\ l+m+n=p}} \frac{p!}{l!m!n!} \breve{M}_{lmn} u_x^l u_y^m u_z^n \end{aligned} \tag{4.20}$$

to be a homogeneous polynomial of degree p, and we consequently restrict $l+m+n=p$. In the expression for H_p, we have further defined the *geometric moments* as

$$\breve{M}_{lmn} \triangleq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x^l y^m z^n \rho(x,y,z)dxdydz, \tag{4.21}$$

where $l, m, n = 0, 1, 2, \ldots$, and we say that the geometric moment $\breve{M}_{lmn}$ has order $(l + m + n)$. A convenient way to write (4.20) followed in [64] is to collect all monomials of u_x, u_y, u_z of order p into a vector $\mathcal{U}_p$:

$$\mathcal{U}_p \triangleq [u_x^p, u_y^p, u_z^p, u_x^{p-1}u_y, u_x^{p-1}u_z, u_x u_y^{p-1}, u_y^{p-1}u_z, \ldots]^T. \tag{4.22}$$

[4] $\exp(x) = \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots$

[5] $(\boldsymbol{u}\cdot\boldsymbol{x})^p = \sum_{\substack{l,m,n\geq 0 \\ l+m+n=p}} \frac{p!}{l!m!n!}(u_x x)^l (u_y y)^m (u_z z)^n = \sum_{\substack{l,m,n\geq 0 \\ l+m+n=p}} \frac{p!}{l!m!n!}(u_x^l u_y^m u_z^n)(x^l y^m z^n)$

Each $\mathcal{U}_p$ has length $N_p \triangleq (p+1)(p+2)/2$; for example, $\mathcal{U}_2 = [u_x^2, u_y^2, u_z^2, u_x u_y, u_x u_z, u_y u_z]^T$ has 6 components. If we complementarily define the vector $\mathcal{M}_p$ of length N_p to contain the coefficients corresponding to the monomials of $\mathcal{U}_p$ in (4.20),

$$\mathcal{M}_p \triangleq \left[\breve{M}_{p00}, \breve{M}_{0p0}, \breve{M}_{00p}, \ldots, \frac{p!}{l!m!n!}\breve{M}_{lmn}, \ldots \right]^T, \tag{4.23}$$

then H_p may be succinctly expressed as

$$H_p(u_x, u_y, u_z) = \mathcal{M}_p \cdot \mathcal{U}_p. \tag{4.24}$$

It should be noted that some or all of the geometric moments may not exist depending on the convergence of the integrals in (4.21), rendering the transformation from (4.18) to (4.19) impossible. However, in the present application, we consider classes of atomic density functions which are restricted to be piecewise continuous and compactly supported in $\mathbb{R}^3$ (further implying boundedness). Under these conditions, geometric moments of all orders are guaranteed to exist[6] and can be interpreted as quantities which determine the coefficients of a Maclaurin series expansion of the characteristic function in the three-dimensional space defined by the variables u_x, u_y, and u_z.

Familiar moments include the mean, variance, kurtosis, and skewedness, which correspond to moments of orders one through four, respectively. Whether used in practice or theory, these quantities are more conveniently defined using *central geometric moments* rather than the ordinary geometric moments. Similar to (4.21), we define

$$M_{lmn} \triangleq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\bar{x})^l(y-\bar{y})^m(z-\bar{z})^n\rho(x,y,z)dxdydz, \tag{4.25}$$

where

$$\bar{x} = \frac{\breve{M}_{100}}{\breve{M}_{000}} \qquad \bar{y} = \frac{\breve{M}_{010}}{\breve{M}_{000}} \qquad \bar{z} = \frac{\breve{M}_{001}}{\breve{M}_{000}}. \tag{4.26}$$

The primary reason for doing this is that the central geometric moments are translation-invariant. However, it also has the fortuitous consequence of simplifying many analytical expressions which make use of moments, since the first-order central geometric moments

[6]When geometric moments exist, they can also be obtained directly from the derivatives of φ_ρ according to

$$\breve{M}_{lmn} = i^{-(l+m+n)}\left[\frac{\partial\varphi_\rho(\boldsymbol{u})}{\partial u_x^l \partial u_y^m \partial u_z^n}\right]_{\boldsymbol{u}=\boldsymbol{0}}.$$

Thus, the characteristic function can be understood as a convenient method of encoding a set of statistical parameters which describe a distribution.

are always identically zero. As there is no compelling reason to use the ordinary geometric moments of (4.21), all content relevant to moments shall hereafter refer to only central geometric moments, and the term "geometric moment" will be used with this understanding in mind. Moreover, we may define Eqs. (4.18), (4.19), and (4.20) all in terms of central moments, and henceforth by referencing these equations, we refer to this case.

Although any descriptor of atomic environments based on moments will automatically be invariant to global translations of the neighborhood density, as well as permutationally invariant since each moment M_{lmn} is additive, we are evidently left with the task of ensuring rotational invariance. Thus, the remainder of this section will be motivated by the action of a proper rotation operation $\mathcal{R} \in \mathrm{SO}(3)$ on an arbitrary density $\rho(\boldsymbol{x})$, which produces a rotated density $\rho'(\boldsymbol{x})$ according to the following definition:

$$\rho'(\boldsymbol{x}) = \mathcal{R}\rho(\boldsymbol{x}) \triangleq \rho(\mathbf{R}^{-1}\boldsymbol{x}), \tag{4.27}$$

where $\mathbf{R}$ is the element of the matrix representation of SO(3) corresponding to $\mathcal{R}$ (written in basis of $\boldsymbol{x}$). Since we are assuming that all of the moments M_{lmn} of the original density $\rho(\boldsymbol{x})$ exist, the moments M'_{lmn} of the rotated density exist, as well:

$$M'_{lmn} \triangleq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x')^l(y')^m(z')^n\rho'(x',y',z')dx'dy'dz'. \tag{4.28}$$

Our objective in deriving a descriptor based upon geometric moments is then to identify functions $\mathcal{F}$ of the moments which are invariant under rotation of the density:

$$\mathcal{F}(\{M_{lmn}\}_{l,m,n\geq 0}) = \mathcal{F}(\{M'_{lmn}\}_{l,m,n\geq 0}). \tag{4.29}$$

Such functions, which thus satisfy all three desired forms of invariance aforementioned, are known as *moment invariants*. The study of moment invariants began with the landmark 1962 publication of Hu [65], who used algebraic methods to derive a set of seven invariants for two-dimensional distributions involving second- and third-order moments. The approach Hu used to generate his invariants stems from what he termed the *Fundamental Theorem of Moment Invariants* (hereafter, "the fundamental theorem"),[7] which was later generalized to arbitrary dimensions by Mamistvalov [69]. As a prerequisite to the statement of the theorem, we will need to establish several definitions used in [69], which we restrict here to apply to the problem of three-dimensional moments.

[7] It has been pointed out in [66–68] that the original statement of the theorem given by Hu in [65] is incorrect.

The general form of $H_p(\boldsymbol{u})$ in (4.20) is that of a homogeneous polynomial f_p in three variables u_1, u_2, u_3:

$$f_p(u_1, u_2, u_3) = \sum_{\substack{p_1,p_2,p_3 \geq 0 \\ p_1+p_2+p_3=p}} \frac{p!}{p_1!p_2!p_3!} a_{p_1p_2p_3} u_1^{p_1} u_2^{p_2} u_3^{p_3}. \tag{4.30}$$

A function of this form is generally referred to as a *ternary quantic* or *ternary algebraic form* of order p [70]. For example, a ternary quantic of order two takes the form

$$f_2(u_1, u_2, u_3) = a_{200}u_1^2 + a_{020}u_2^2 + a_{002}u_3^2 + 2a_{110}u_1u_2 + 2a_{101}u_1u_3 + 2a_{011}u_2u_3. \tag{4.31}$$

For notational convenience, we collect the coefficients $\{a_{p_1p_2p_3}\}_{p_1+p_2+p_3=p}$ of a ternary quantic of order p into the vector $\mathbf{a}_p$. Now, consider a new set of variables $\boldsymbol{u'} = (u'_1, u'_2, u'_3)^T$ which are related to the original variables $\boldsymbol{u} = (u_1, u_2, u_3)^T$ by an invertible linear transformation $\mathbf{B}$ in the following manner:

$$\boldsymbol{u} = \mathbf{B}^T \boldsymbol{u'}, \tag{4.32}$$

i.e. the transformed variables $\boldsymbol{u'}$ are defined from the original variables $\boldsymbol{u}$ by $\boldsymbol{u'} = \mathbf{B}^{-T}\boldsymbol{u}$. Using (4.32) to substitute for $\boldsymbol{u}$ in (4.30), we obtain another ternary quantic f'_p of the same order in the new set of variables u'_1, u'_2, u'_3 containing coefficients which we denote $\mathbf{a}'_{p,\mathbf{B}} \triangleq \{a'_{p_1p_2p_3,\mathbf{B}}\}_{p_1+p_2+p_3=p}$. With these relations considered, we define a homogeneous polynomial I of degree k of the coefficients $\mathbf{a}_p$ to be a *relative algebraic invariant of weight w* if, for every nondegenerate linear transformation $\mathbf{B}$, it is true that

$$I(\mathbf{a}'_{p,\mathbf{B}}) = J^w I(\mathbf{a}_p), \tag{4.33}$$

where $J \triangleq \det(\mathbf{B}^T) = \det(\mathbf{B}) \neq 0$. In the event that (4.33) holds with $w = 0$, we refer to I as an *absolute algebraic invariant*. It is important to note that an algebraic invariant disregards the particular values of the coefficients $\mathbf{a}_p$, as well as the choice of variables used; therefore, an algebraic invariant of one ternary quantic is an algebraic invariant of all ternary quantics. Furthermore, although we have presented I here as depending on the coefficients of a single algebraic form, it may generally depend on the coefficients of multiple forms which differ in order [65, 66, 71–73]. Having collected these definitions, the fundamental theorem then stipulates the following for the special case of three dimensions:

Theorem. **Fundamental Theorem of Moment Invariants (ternary case)**

Consider a density $\rho(\boldsymbol{x})$ which is mapped to a new density $\rho'(\boldsymbol{x})$ by an invertible linear transformation $\tilde{\mathbf{B}}$ according to

$$\rho'(\boldsymbol{x}) = \tilde{\mathbf{B}}\rho(\boldsymbol{x}) \triangleq \rho(\tilde{\mathbf{B}}^{-1}\boldsymbol{x}).$$

Furthermore, suppose there exists a ternary quantic f_p of order p which has a relative algebraic invariant I of weight w and degree k. Then the three-dimensional geometric moments of order p of the density $\rho(\boldsymbol{x})$ have the same invariant, but with the additional factor $|J|^k$, where $J \triangleq \det \tilde{\mathbf{B}}$ and $|\cdot|$ denotes absolute value:

$$I(M'_{p00}, M'_{p-1,10}, \ldots, M'_{00p}) = J^w |J|^k I(M_{p00}, M_{p-1,10}, \ldots, M_{00p}).$$

Proof. The density $\rho(\boldsymbol{x})$ has an associated characteristic function $\varphi_\rho(\boldsymbol{u})$ which can, in turn, be written in terms of a set of homogeneous polynomials H_q $(q = 0, 1, 2, \ldots)$ as defined in (4.20). In particular, for $q = p$, we have the following term of $\varphi_\rho(\boldsymbol{u})$:

$$H_p(\boldsymbol{u}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} (\boldsymbol{u}\cdot\boldsymbol{x})^p \rho(\boldsymbol{x}) dxdydz. \tag{4.34}$$

By speaking of an algebraic invariant I of f_p, we are implicitly referring to the set of all nondegenerate linear transformations $\mathbf{B}$ under which the coefficients $\mathbf{a}_p$ of f_p are transformed to the coefficients $\mathbf{a}'_{p,\mathbf{B}}$ by the change of variables $\boldsymbol{u} = \mathbf{B}^T\boldsymbol{u}'$. Taking $\mathbf{B} = \tilde{\mathbf{B}}$ and rewriting $H_p(\boldsymbol{u})$ in this new transformed basis in Fourier space gives

$$H_p(\boldsymbol{u}) = H_p(\tilde{\mathbf{B}}^T\boldsymbol{u'}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} ((\tilde{\mathbf{B}}^T\boldsymbol{u'})\cdot\boldsymbol{x})^p \rho(\boldsymbol{x}) dxdydz. \tag{4.35}$$

Although $H_p(\boldsymbol{u})$ can be written in terms of the moments M_{lmn} as it was in (4.24), the integral to the right in (4.35) has no simple relation to the moments M'_{lmn} of the transformed density. However, defining a new set of coordinates $\boldsymbol{x'}$ as the application of the transformation $\tilde{\mathbf{B}}$ to the original coordinates $\boldsymbol{x}$,

$$\boldsymbol{x'} = \tilde{\mathbf{B}}\boldsymbol{x}, \tag{4.36}$$

admits simplification. Applying this transformation to (4.35), i.e. making the change of

variables $\boldsymbol{x} = \tilde{\mathbf{B}}^{-1}\boldsymbol{x'}$, we have

$$H_p(\boldsymbol{u}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(\tilde{\mathbf{B}}^T\boldsymbol{u'}\cdot\tilde{\mathbf{B}}^{-1}\boldsymbol{x'})^p\rho(\tilde{\mathbf{B}}^{-1}\boldsymbol{x'})\frac{1}{|J|}dx'dy'dz', \tag{4.37}$$

where we have used the Jacobian determinant J of $\tilde{\mathbf{B}}$ to substitute $dxdydz = dx'dy'dz'/|J|$. The density in the integrand of (4.37) can be identified as the transformed density in the coordinates $\boldsymbol{x'}$ using the definition $\rho'(\boldsymbol{x}) \triangleq \rho(\tilde{\mathbf{B}}^{-1}\boldsymbol{x})$ given in the theorem statement:

$$\rho(\tilde{\mathbf{B}}^{-1}\boldsymbol{x'}) = \rho'(\boldsymbol{x'}). \tag{4.38}$$

Further, note that

$$\begin{aligned}\boldsymbol{u}\cdot\boldsymbol{x} &= \tilde{\mathbf{B}}^T\boldsymbol{u'}\cdot\tilde{\mathbf{B}}^{-1}\boldsymbol{x'}\\ &= (\tilde{\mathbf{B}}^T\boldsymbol{u'})^T\tilde{\mathbf{B}}^{-1}\boldsymbol{x'}\\ &= \boldsymbol{u'}^T\tilde{\mathbf{B}}\tilde{\mathbf{B}}^{-1}\boldsymbol{x'}\\ &= \boldsymbol{u'}^T\boldsymbol{x'} = \boldsymbol{u'}\cdot\boldsymbol{x'},\end{aligned} \tag{4.39}$$

so that (4.37) becomes

$$H_p(\boldsymbol{u}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(\boldsymbol{u'}\cdot\boldsymbol{x'})^p\rho'(\boldsymbol{x'})\frac{1}{|J|}dx'dy'dz'. \tag{4.40}$$

Equating the expressions for $H_p(\boldsymbol{u})$ in (4.40) and (4.34), we have

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(\boldsymbol{u'}\cdot\boldsymbol{x'})^p\rho'(\boldsymbol{x'})\frac{1}{|J|}dx'dy'dz' = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(\boldsymbol{u}\cdot\boldsymbol{x})^p\rho(\boldsymbol{x})dxdydz. \tag{4.41}$$

Multiplying both sides by $|J|$, employing the multinomial theorem, and making use of the definition of the moments in the primed and unprimed coordinate systems as in (4.20), we may rewrite this as

$$\sum_{\substack{l,m,n\geq 0\\ l+m+n=p}}\frac{p!}{l!m!n!}M'_{lmn}{u'_x}^l{u'_y}^m{u'_z}^n = \sum_{\substack{l,m,n\geq 0\\ l+m+n=p}}\frac{p!}{l!m!n!}|J|M_{lmn}u_x^l u_y^m u_z^n. \tag{4.42}$$

The left and right sides of this expression are ternary quantics of order p with coefficients M'_{lmn} and $|J|M_{lmn}$, respectively. Since I is an algebraic invariant of f_p, it is an algebraic

invariant of any ternary quantic of order p and, therefore, also an invariant of the quantics of (4.42). Hence,

$$I(M'_{p00}, M'_{p-1,10}, \ldots, M'_{00p}) = J^w I(|J|M_{p00}, |J|M_{p-1,10}, \ldots, |J|M_{00p}). \tag{4.43}$$

Finally, since I is a homogeneous polynomial of degree k, we can factor out $|J|$ as

$$I(|J|M_{p00}, |J|M_{p-1,10}, \ldots, |J|M_{00p}) = |J|^k I(M_{p00}, M_{p-1,10}, \ldots, M_{00p}) \tag{4.44}$$

to arrive at

$$I(M'_{p00}, M'_{p-1,10}, \ldots, M'_{00p}) = J^w |J|^k I(M_{p00}, M_{p-1,10}, \ldots, M_{00p}). \tag{4.45}$$

■

In summary, the fundamental theorem tells us that if we are able to find an algebraic invariant of a ternary quantic of order p under rotation, it will also be an invariant of the three-dimensional geometric moments of order p under rotation since $J = 1$ in this case. The subsequent difficulty we are faced with amounts to identifying the algebraic invariants themselves: a task for which the fundamental theorem provides no guidance. Hu and others have been able to derive different sets of moment invariants by leveraging work on the theory of algebraic invariants pioneered by Sylvester, Boole, Cayley, and Hilbert [70, 73–75]. For example, Sadjadi and Hall [71] derived three-dimensional moment invariants using the fact that the determinant of the Hessian is an algebraic invariant. However, while these efforts comprise a significant contribution to the study of shape invariants and their results have been utilized in numerous publications,[8] we will see next that the task of deriving moment invariants is greatly facilitated by a prudent choice of basis capable of encompassing the algebraic complexities we have encountered.

4.2.2 Complex moments

In the previous section, we analyzed an atomic neighborhood density $\rho(\boldsymbol{x})$ by computing its characteristic function $\varphi_\rho(\boldsymbol{u})$, i.e. its inverse Fourier transform. Examining (4.18), it is

[8] The original publication by Hu [65] has been cited over seven thousand times.

apparent that if we define the L^2 inner product of two complex functions f and g on $\mathbb{R}^3$ as

$$\langle f, g \rangle \triangleq \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \overline{f(x,y,z)} g(x,y,z) dx dy dz, \tag{4.46}$$

where $\overline{(\,\cdot\,)}$ denotes complex conjugation, then the characteristic function of $\rho(\boldsymbol{x})$ is obtained by projecting it onto the basis of functions $\{e^{-i\boldsymbol{u}\cdot\boldsymbol{x}} : \boldsymbol{u} \in \mathbb{R}^3\}$. The geometric moments M_{lmn} of $\rho(\boldsymbol{x})$, which are obtained by taking its projection onto the basis of monomials,

$$M_{lmn} = \left\langle x^l y^m z^n, \rho \right\rangle, \tag{4.47}$$

were shown to be intimately connected to its characteristic function, determining the coefficients of its Maclaurin series expansion. In pursuit of our end goal of deriving moment invariants, i.e. functions of the moments M_{lmn} which are invariant under rotations $\mathcal{R} \in \mathrm{SO}(3)$ of $\rho(\boldsymbol{x})$, we thereafter faced the unenviable task of formulating absolute algebraic invariants of homogeneous polynomials of u_x, u_y, u_z under a corresponding rotation defined by $\boldsymbol{u'} = \mathbf{R}^{-T}\boldsymbol{u} = \mathbf{R}\boldsymbol{u}$. The primary motivation for the current section is that the process of finding moment invariants can be expedited by eschewing the monomial basis which defines the moments M_{lmn} in favor of a basis of functions which is endowed with precisely the symmetry we seek to enforce, and accordingly gives rise to its own type of moments.

It was observed by Courant and Hilbert [76, p. 540–541] that there is a natural correspondence between the homogeneous polynomials and a set of harmonic polynomials known as the *regular solid harmonics* [77–82], which are defined as

$$\mathcal{Y}_\ell^m(r, \theta, \phi) \triangleq r^\ell Y_\ell^m(\theta, \phi). \tag{4.48}$$

The quantity r and the angles $\theta \in [0, \pi]$ and $\phi \in [0, 2\pi)$ in (4.48) are obtained from the vector $\boldsymbol{x} = [x, y, z]^T$ using the standard transformations between Cartesian and spherical coordinates:

$$\begin{aligned} x &= r\sin\theta\cos\phi, & r &= \sqrt{x^2 + y^2 + z^2}\,, \\ y &= r\sin\theta\sin\phi, & \phi &= \mathrm{atan}(y/x), \\ z &= r\cos\theta, & \theta &= \arccos(z/r). \end{aligned} \tag{4.49}$$

The indices ℓ and m of $\mathcal{Y}_\ell^m$ are restricted such that ℓ is a non-negative integer and m is an

integer confined to $-\ell \leq m \leq \ell$. For a given ℓ and m fulfilling these criteria, the *spherical harmonics* Y_ℓ^m appearing on the right side of (4.48) are defined as

$$Y_\ell^m(\theta,\phi) \triangleq \sqrt{\frac{2\ell+1}{4\pi}\frac{(\ell-m)!}{(\ell+m)!}}\, P_\ell^m(\cos\theta)e^{im\phi}, \tag{4.50}$$

where $i = \sqrt{-1}$ and P_ℓ^m are the associated Legendre polynomials. We will revisit the spherical harmonics shortly, but for now we note that they are particularly desirable in defining a basis for problems which feature angular dependence because they form a complete orthonormal basis for (square-integrable) functions defined on the unit sphere:

$$\int_0^{2\pi}\int_0^{\pi} \overline{Y_\ell^m(\theta,\phi)}Y_{\ell'}^{m'}(\theta,\phi)\sin\theta d\theta d\phi = \delta_{\ell\ell'}\delta_{mm'}. \tag{4.51}$$

Additional details of the definition given in (4.50) can be found in Appendix A.

The relation between the homogeneous polynomials and the solid harmonics can be most readily understood by writing the latter in Cartesian coordinates [83]. This can be accomplished by making the substitutions

$$e^{i\phi} = \frac{x+iy}{\sqrt{x^2+y^2}}, \tag{4.52}$$

$$\theta = \arccos\left(\frac{z}{r}\right) \tag{4.53}$$

in (4.50). For example, for $\ell = 1$,

$$\mathcal{Y}_1^1(\boldsymbol{x}) = rY_1^1(x,y,z) = r\left(-\frac{1}{2}\sqrt{\frac{3}{2\pi}}\,\frac{x+iy}{r}\right) = -\left(\frac{1}{2}\sqrt{\frac{3}{2\pi}}\right)x - \left(\frac{i}{2}\sqrt{\frac{3}{2\pi}}\right)y,$$

$$\mathcal{Y}_1^0(\boldsymbol{x}) = rY_1^0(x,y,z) = r\left(\frac{1}{2}\sqrt{\frac{3}{\pi}}\,\frac{z}{r}\right) = \left(\frac{1}{2}\sqrt{\frac{3}{\pi}}\right)z,$$

$$\mathcal{Y}_1^{-1}(\boldsymbol{x}) = rY_1^{-1}(x,y,z) = r\left(\frac{1}{2}\sqrt{\frac{3}{2\pi}}\,\frac{x-iy}{r}\right) = \left(\frac{1}{2}\sqrt{\frac{3}{2\pi}}\right)x - \left(\frac{i}{2}\sqrt{\frac{3}{2\pi}}\right)y.$$

For $\ell = 2$,

$$\begin{aligned}\mathcal{Y}_2^2(\boldsymbol{x}) = r^2Y_2^2(x,y,z) &= r^2\left(\frac{1}{4}\sqrt{\frac{15}{2\pi}}\,\frac{(x+iy)^2}{r^2}\right)\\ &= \left(\frac{1}{4}\sqrt{\frac{15}{2\pi}}\right)x^2 + \left(\frac{i}{2}\sqrt{\frac{15}{2\pi}}\right)xy - \left(\frac{1}{4}\sqrt{\frac{15}{2\pi}}\right)y^2,\end{aligned}$$

$$
\begin{aligned}
\mathcal{Y}_2^1(\boldsymbol{x}) = r^2 Y_2^1(x,y,z) &= r^2 \left(-\frac{1}{2}\sqrt{\frac{15}{2\pi}}\,\frac{(x+iy)z}{r^2} \right) \\
&= -\left(\frac{1}{2}\sqrt{\frac{15}{2\pi}} \right) xz - \left(\frac{i}{2}\sqrt{\frac{15}{2\pi}} \right) yz, \\
\mathcal{Y}_2^0(\boldsymbol{x}) = r^2 Y_2^0(x,y,z) &= r^2 \left(\frac{1}{4}\sqrt{\frac{5}{\pi}}\,\frac{(2z^2-x^2-y^2)}{r^2} \right) \\
&= \left(\frac{1}{2}\sqrt{\frac{5}{\pi}} \right) z^2 - \left(\frac{1}{4}\sqrt{\frac{5}{\pi}} \right) x^2 - \left(\frac{1}{4}\sqrt{\frac{5}{\pi}} \right) y^2, \\
\mathcal{Y}_2^{-1}(\boldsymbol{x}) = r^2 Y_2^{-1}(x,y,z) &= r^2 \left(\frac{1}{2}\sqrt{\frac{15}{2\pi}}\,\frac{(x-iy)z}{r^2} \right) \\
&= \left(\frac{1}{2}\sqrt{\frac{15}{2\pi}} \right) xz - \left(\frac{i}{2}\sqrt{\frac{15}{2\pi}} \right) yz, \\
\mathcal{Y}_2^{-2}(\boldsymbol{x}) = r^2 Y_2^{-2}(x,y,z) &= r^2 \left(\frac{1}{4}\sqrt{\frac{15}{2\pi}}\,\frac{(x-iy)^2}{r^2} \right) \\
&= \left(\frac{1}{4}\sqrt{\frac{15}{2\pi}} \right) x^2 - \left(\frac{i}{2}\sqrt{\frac{15}{2\pi}} \right) xy - \left(\frac{1}{4}\sqrt{\frac{15}{2\pi}} \right) y^2.
\end{aligned}
$$

From the above expressions, we see that each $\mathcal{Y}_1^m$ is a homogeneous polynomial of degree 1 in the variables x, y, z. Similarly, each $\mathcal{Y}_2^m$ is a homogeous polynomial of degree 2 in the variables x, y, z. The trend continues, with each $\ell \geq 0$ defining a set of solid harmonics $\mathcal{Y}_\ell^m$ $(m = -\ell, \ldots, \ell)$ which are homogeneous polynomials of degree ℓ in the variables x, y, z. Finally, we note that because x, y, z are nothing more than symbolic variables, we may choose instead to write any of the solid harmonics $\mathcal{Y}_\ell^m$ defined above as functions of the (Cartesian) variables $\boldsymbol{u} = [u_x, u_y, u_z]^T$.

All of the above statements can be conveniently summarized if we define for each integer $p \geq 0$ a vector $\mathcal{U}_p$ to contain the set of all $N_p = (p+1)(p+2)/2$ monomials of degree p in the variables u_x, u_y, u_z, just as was done in (4.22):

$$
\mathcal{U}_p \triangleq [u_x^p, u_y^p, u_z^p, u_x^{p-1}u_y, u_x^{p-1}u_z, u_x u_y^{p-1}, u_y^{p-1}u_z, \ldots]^T. \tag{4.54}
$$

Next, let the N_p-dimensional vector

$$
\Upsilon_p \triangleq [\mathcal{Y}_p^p(\boldsymbol{u}), \ldots, \mathcal{Y}_p^{-p}(\boldsymbol{u}), \mathcal{Y}_{p-2}^{p-2}(\boldsymbol{u}), \ldots, \mathcal{Y}_{p-2}^{-(p-2)}(\boldsymbol{u}), \ldots]^T \tag{4.55}
$$

contain the set of all $\mathcal{Y}_\ell^m(\boldsymbol{u})$ with $\ell = p, p-2, p-4, \ldots, 0$ and where $m = -\ell, \ldots, \ell$ for

each ℓ. Following Lo and Don [64], we assert with these definitions that there exists for each $p \geq 0$ a nonsingular, complex matrix $\mathcal{A}_p$ of dimensions $N_p \times N_p$ such that[9]

$$\Upsilon_p = \mathcal{A}_p \mathcal{U}_p. \tag{4.56}$$

Rewriting $H_p(\boldsymbol{u})$ in the basis of functions contained in Υ_p leads to a corresponding set of coefficients which we collect in the vector $\mathcal{C}_p$:

$$\mathcal{C}_p \triangleq \mathcal{A}_p^{-\dagger} \mathcal{M}_p, \tag{4.57}$$

so that $H_p(\boldsymbol{u}) = \mathcal{M}_p \cdot \mathcal{U}_p = \mathcal{C}_p \cdot \Upsilon_p$. Here, we have used $(\,\cdot\,)^\dagger$ to denote the conjugate transpose and, as in (4.23), we have assembled the length-N_p vector of geometric moments of order $l + m + n = p$ of the density $\rho(\boldsymbol{x})$ as

$$\mathcal{M}_p \triangleq \left[M_{p00}, M_{0p0}, M_{00p}, \ldots, \frac{p!}{l!m!n!} M_{lmn}, \ldots \right]^T. \tag{4.58}$$

The elements of $\mathcal{C}_p$, which we denote by c_ℓ^m and term *complex moments*, are arranged similarly to those of Υ_p in (4.55):

$$\mathcal{C}_p \triangleq [\underbrace{c_p^p, \ldots, c_p^{-p}}_{\mathbf{c}_p}, \underbrace{c_{p-2}^{p-2}, \ldots, c_{p-2}^{-(p-2)}}_{\mathbf{c}_{p-2}}, \underbrace{c_{p-4}^{p-4}, \ldots, c_{p-4}^{-(p-4)}}_{\mathbf{c}_{p-4}}, \ldots]^T, \tag{4.59}$$

and we have defined the $(2\ell + 1)$-dimensional vectors of complex moments

$$\mathbf{c}_\ell \triangleq [c_\ell^\ell, \ldots, c_\ell^{-\ell}]. \tag{4.60}$$

Just as the moments $M_{\ell mn}$ $(\ell + m + n = p)$ present in $\mathcal{M}_p$ are obtained by projecting the atomic neighborhood density $\rho(\boldsymbol{x})$ onto the monomial basis $x^l y^m z^n$ ($l, m, n \geq 0$), the complex moments are obtained by projecting the density onto the basis functions $\mathcal{Y}_\ell^m(\boldsymbol{u})$ present in Υ_p:

$$c_\ell^m = \langle \mathcal{Y}_l^m, \rho \rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} r^\ell \, \overline{Y_\ell^m(x, y, z)} \rho(x, y, z) dx dy dz. \tag{4.61}$$

The advantage gained by performing the above transformations to define the complex mo-

[9] The generating equation for the elements of $\mathcal{A}_p$ can be found in [84, Eq. 6.6], as well as [82]. The reader is also referred to the related work of Kakarala *et al.* [85, 86].

ments is that, rather than deriving the moment invariants $\mathcal{F}$ of (4.29) from the geometric moments $\mathcal{M}_p$ ($p = 0, 1, \dots$), they may equivalently be found by determining invariants of the complex moments $\mathcal{C}_p$. This alternative means of derivation is particularly expedient because of the behavior of the basis Υ_p under rotation. While in the case of geometric moments, a rotation $\boldsymbol{u'} = \mathbf{R}\boldsymbol{u}$ results in a complicated transformation of the monomial bases $\mathcal{U}_p$, the transformation of the bases Υ_p is decidedly more basic because the spherical harmonics transform under a rotation $\mathcal{R}$ according to a special set of matrices known as the *Wigner D-matrices* or *Wigner D-functions* [87, 88] (see Appendix A for an explicit definition of these quantities). Denoting the elements of the Wigner matrices $\mathbf{D}^\ell(\mathcal{R})$ by $D^\ell_{m'm}(\mathcal{R})$ ($m', m = -\ell, \dots, \ell$), we write the action of a rotation $\mathcal{R}$ on the spherical harmonic Y^m_ℓ as

$$\mathcal{R}Y^m_\ell(\boldsymbol{x}) = Y^m_\ell(\mathbf{R}^{-1}\boldsymbol{x}) = \sum_{m'=-\ell}^{\ell} D^\ell_{m'm}(\mathcal{R})Y^{m'}_\ell(\boldsymbol{x}). \tag{4.62}$$

Collecting the spherical harmonics $Y^m_\ell(\boldsymbol{x})$ ($m = -\ell, \dots, \ell$) into a $(2\ell+1)$-dimensional vector $\mathbf{Y}_\ell(\boldsymbol{x})$, this can be expressed more compactly as

$$\mathbf{Y}_\ell(\mathbf{R}^{-1}\boldsymbol{x}) = (\mathbf{D}^\ell(\mathcal{R}))^T\mathbf{Y}_\ell(\boldsymbol{x}). \tag{4.63}$$

Next, without loss of generality, we may replace the operator $\mathcal{R}$ in (4.63) by $\mathcal{R}^{-1}$ (and the corresponding matrix representation element $\mathbf{R}$ by $\mathbf{R}^{-1}$) to obtain

$$\mathbf{Y}_\ell(\mathbf{R}\boldsymbol{x}) = \mathbf{Y}_\ell(\boldsymbol{x'}) = (\mathbf{D}^\ell(\mathcal{R}^{-1}))^T\mathbf{Y}_\ell(\boldsymbol{x}), \tag{4.64}$$

where we have defined the rotated basis $\boldsymbol{x'} \triangleq \mathbf{R}\boldsymbol{x}$. The defining feature of the Wigner D-matrices is that they are unitary:

$$\mathbf{D}^\ell(\mathcal{R})^\dagger\mathbf{D}^\ell(\mathcal{R}) = \mathbf{D}^\ell(\mathcal{R})\mathbf{D}^\ell(\mathcal{R})^\dagger = \mathbf{I}. \tag{4.65}$$

It can further be verified from their definition that $\mathbf{D}^\ell(\mathcal{R})\mathbf{D}^\ell(\mathcal{R}^{-1}) = \mathbf{I}$, so that the unitarity above implies that

$$\mathbf{D}^\ell(\mathcal{R}^{-1}) = \mathbf{D}^\ell(\mathcal{R})^\dagger. \tag{4.66}$$

Using (4.66), we may rewrite (4.64) as $\mathbf{Y}_\ell(\boldsymbol{x'}) = \overline{\mathbf{D}^\ell(\mathcal{R})}\mathbf{Y}_\ell(\boldsymbol{x})$, and introduce an additional simplification to our notation[10] by expressing the action of a rotation $\mathcal{R}$ on a spherical

[10]We transition to this notation in order to be consistent with the work of Bartók *et al.* [62], which will become relevant as we proceed further.

harmonic vector $\mathbf{Y}_\ell(\boldsymbol{x})$ by writing

$$\mathbf{Y}_\ell(\boldsymbol{x}) \xrightarrow{R} \overline{\mathbf{D}^\ell(\mathcal{R})}\mathbf{Y}_\ell(\boldsymbol{x}). \tag{4.67}$$

Finally, using the definition of each individual complex moment c_ℓ^m in (4.61), it can be shown that each complex moment vector $\mathbf{c}_\ell$ accordingly transforms under rotation as

$$\mathbf{c}_\ell \xrightarrow{R} \mathbf{D}^\ell(\mathcal{R})\mathbf{c}_\ell. \tag{4.68}$$

Because the Wigner matrices possess SO(3) symmetry, the significance of (4.68) is that a rotation $\mathcal{R}$ of the atomic neighborhood density $\rho(\boldsymbol{x})$ gives rise to rotations of each individual vector $\mathbf{c}_\ell$ of complex moments which make up the vector $\mathcal{C}_p$ in (4.59). Like the gears of a clock movement, a rotation of the density begets a concerted set of rotations in complex spaces of dimensions $2p+1$, $2p-3$, $2p-7$, $\ldots$. More explicitly, the vector $\mathcal{C}_p$ transforms under rotation as $\mathcal{C}_p \xrightarrow{R} \mathfrak{D}\,\mathcal{C}_p$, where the block diagonal matrix $\mathfrak{D}$ is composed of Wigner D-matrices $\mathbf{D}^\ell(\mathcal{R})$ with $\ell = p, p-2, p-4, \ldots$:[11]

$$\mathfrak{D} = \begin{bmatrix} \mathbf{D}^p(\mathcal{R}) & & & \\ & \mathbf{D}^{p-2}(\mathcal{R}) & & \\ & & \mathbf{D}^{p-4}(\mathcal{R}) & \\ & & & \ddots \end{bmatrix}.$$

The particularly simple form of (4.68) allows rotation invariants to be derived with relative ease thanks to the unitarity of the Wigner matrices. Collecting subsets of these rotation invariants then allows the definition of a descriptor of atomic environments comparable to those considered in the previous sections of this chapter. The first set of rotation invariants which we consider, known as the *power spectrum* of the density $\rho(\boldsymbol{x})$, is defined by

$$p_\ell \triangleq ||\mathbf{c}_\ell||^2 = \mathbf{c}_\ell^\dagger \mathbf{c}_\ell, \tag{4.69}$$

where $\ell = 0, \ldots, \infty$. The rotational invariance of the power spectrum is made evident by

[11] Because the Wigner matrices $\mathbf{D}^\ell(\mathcal{R})$ of a given ℓ form an irreducible representation of SO(3), our derivation could be summarized from a group-theoretic perspective by saying that we have transitioned from the geometric moments, which transform under rotation according to a reducible representation, to the complex moments, which transform irreducibly. The interested reader may consult [82] for more on this topic.

substitution of (4.68), where we now omit the dependence of $\mathbf{D}^\ell$ on $\mathcal{R}$ for brevity:

$$p_\ell = \mathbf{c}_\ell^\dagger \mathbf{c}_\ell \xrightarrow{R} (\mathbf{D}^\ell \mathbf{c}_\ell)^\dagger (\mathbf{D}^\ell \mathbf{c}_\ell) = \mathbf{c}_\ell^\dagger (\mathbf{D}^\ell)^\dagger \mathbf{D}^\ell \mathbf{c}_\ell = \mathbf{c}_\ell^\dagger \mathbf{c}_\ell. \tag{4.70}$$

Higher-order polyspectra[12] can also be formed by coupling complex moment vectors which have different values of ℓ. For instance, consider the tensor product of $\mathbf{c}_{\ell_1}$ and $\mathbf{c}_{\ell_2}$, which is transformed under rotation by the tensor product of the corresponding Wigner matrices:

$$\mathbf{c}_{\ell_1} \otimes \mathbf{c}_{\ell_2} \xrightarrow{R} (\mathbf{D}^{\ell_1} \otimes \mathbf{D}^{\ell_2}) (\mathbf{c}_{\ell_1} \otimes \mathbf{c}_{\ell_2}). \tag{4.71}$$

This tensor product is given by

$$\mathbf{D}^{\ell_1} \otimes \mathbf{D}^{\ell_2} = (\mathbf{C}^{\ell_1,\ell_2})^\dagger \left[\bigoplus_{\ell=|\ell_1-\ell_2|}^{\ell_1+\ell_2} \mathbf{D}^\ell \right] \mathbf{C}^{\ell_1,\ell_2}, \tag{4.72}$$

where $\mathbf{C}^{\ell_1,\ell_2}$ are unitary matrices whose elements are the *Clebsch–Gordan coefficients* [88, Ch. 8]. Combining (4.71) and (4.72) allows the unitarity of the matrices $\mathbf{C}^{\ell_1,\ell_2}$ to be exploited through the observation that the vector $\mathbf{C}^{\ell_1,\ell_2}(\mathbf{c}_{\ell_1} \otimes \mathbf{c}_{\ell_2})$ transforms under rotation as

$$\begin{aligned} \mathbf{C}^{\ell_1,\ell_2}(\mathbf{c}_{\ell_1} \otimes \mathbf{c}_{\ell_2}) &\xrightarrow{R} \mathbf{C}^{\ell_1,\ell_2}(\mathbf{C}^{\ell_1,\ell_2})^\dagger \left[\bigoplus_{\ell=|\ell_1-\ell_2|}^{\ell_1+\ell_2} \mathbf{D}^\ell \right] \mathbf{C}^{\ell_1,\ell_2}(\mathbf{c}_{\ell_1} \otimes \mathbf{c}_{\ell_2}) \\ &= \left[\bigoplus_{\ell=|\ell_1-\ell_2|}^{\ell_1+\ell_2} \mathbf{D}^\ell \right] \mathbf{C}^{\ell_1,\ell_2}(\mathbf{c}_{\ell_1} \otimes \mathbf{c}_{\ell_2}). \end{aligned} \tag{4.73}$$

The form of (4.73) implies that the vector $\mathbf{C}^{\ell_1,\ell_2}(\mathbf{c}_{\ell_1} \otimes \mathbf{c}_{\ell_2})$ itself possesses a block diagonal structure, and we may thus write

$$\mathbf{C}^{\ell_1,\ell_2}(\mathbf{c}_{\ell_1} \otimes \mathbf{c}_{\ell_2}) = \bigoplus_{\ell=|\ell_1-\ell_2|}^{\ell_1+\ell_2} \mathbf{g}_{\ell\,\ell_1\ell_2} \tag{4.74}$$

for some vectors $\mathbf{g}_{\ell\,\ell_1\ell_2}$ accordingly defined, which thereby transform under rotation as

$$\mathbf{g}_{\ell\,\ell_1\ell_2} \xrightarrow{R} \mathbf{D}^\ell \mathbf{g}_{\ell\,\ell_1\ell_2}. \tag{4.75}$$

Finally, we once again exploit the unitarity of the Wigner matrices in order to define the

[12] A compilation of literary citations on the subject of polyspectra can be found in [89].

bispectrum [90] of $\rho(\boldsymbol{x})$ as

$$b_{\ell\ell_1\ell_2} = \mathbf{c}_\ell^\dagger \mathbf{g}_{\ell\,\ell_1\ell_2}, \tag{4.76}$$

(where $\ell = |\ell_1 - \ell_2|, \ldots, \ell_1 + \ell_2$) which is also invariant to rotation, since

$$b_{\ell\ell_1\ell_2} = \mathbf{c}_\ell^\dagger \mathbf{g}_{\ell\,\ell_1\ell_2} \xrightarrow{R} (\mathbf{D}^\ell \mathbf{c}_\ell)^\dagger \mathbf{D}^\ell \mathbf{g}_{\ell\,\ell_1\ell_2} = \mathbf{c}_\ell^\dagger (\mathbf{D}^\ell)^\dagger \mathbf{D}^\ell \mathbf{g}_{\ell\,\ell_1\ell_2} = \mathbf{c}_\ell^\dagger \mathbf{g}_{\ell\,\ell_1\ell_2}. \tag{4.77}$$

In indicial notation, the bispectrum may be written as

$$b_{\ell\ell_1\ell_2} = \sum_{m=-\ell}^{\ell} \sum_{m_1=-\ell_1}^{\ell_1} \sum_{m_2=-\ell_2}^{\ell_2} \overline{c_\ell^m}\, C_{\ell_1 m_1 \ell_2 m_2}^{\ell m} c_{\ell_1}^{m_1} c_{\ell_2}^{m_2}, \tag{4.78}$$

where, by definition, the Clebsch–Gordan coefficients vanish for any combination of their indices for which $m_1 + m_2 \neq m$.

4.2.3 Alternative radial bases

In the previous section, we found that invariants of the geometric moments could equivalently be found by defining complex moments which transformed under rotation according to the unitary Wigner D-matrices. However, it should be recognized that the specific radial basis which complements the spherical harmonics Y_ℓ^m in the definition of the solid harmonics of (4.48) may be interchanged with alternative radial bases in order to define quantities which transform under rotation in the same way as complex moments. Although invariants derived using these latter quantities will not generally be equivalent to the moment invariants, they are nonetheless valid rotation invariants and can equally well be used to form descriptors of atomic environments. For example, Bartók *et al.* [62] demonstrated that the well-known Steinhardt bond order parameters [91] can be obtained by expanding an atomic neighborhood density which consists of Dirac δ distributions placed on each neighboring atom in a basis of spherical harmonics (without the use of any radial basis whatsoever) and taking the power spectrum and bispectrum of the resulting expansion coefficients.

In [62], the authors also review the construction of the power spectrum and bispectrum of an atomic neighborhood density using general radial bases.[13] Let us therefore define an orthonormal radial basis $g_n(r)|_{n=1}^{\infty}$ and write

$$\rho(\boldsymbol{x}) = \sum_{n=1}^{\infty} \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} c_{n\ell m} g_n(r) Y_\ell^m(\hat{\boldsymbol{x}}), \tag{4.79}$$

[13] See also [92].

where we have introduced the notation $r = ||\boldsymbol{x}||$ and $\hat{\boldsymbol{x}} \triangleq \boldsymbol{x}/r$. An important point made in [62] is that, although one could form the power spectrum and bispectrum of the coefficients $c_{n\ell m}$ by

$$p_{n\ell} = \sum_{m=-\ell}^{\ell} \overline{c_{n\ell m}} c_{n\ell m},$$
$$b_{n\ell\ell_1\ell_2} = \sum_{m=-\ell}^{\ell} \sum_{m_1=-\ell_1}^{\ell_1} \sum_{m_2=-\ell_2}^{\ell_2} \overline{c_{n\ell m}}\, C^{\ell m}_{\ell_1 m_1 \ell_2 m_2} c_{n\ell_1 m_1} c_{n\ell_2 m_2},$$

this is problematic because each radial basis channel n is independent. Consequently, the power spectrum or bispectrum of an arbitrary atomic environment as defined above would be invariant to rotations of different radial shells of neighbors relative to one another—transformations which generally produce significant qualitative changes to an atomic environment. Rather, the radial basis functions $g_n(r)$ should be coupled for different values of n, so that the power spectrum and bispectrum are defined as

$$p_{n_1 n_2 \ell} \triangleq \sum_{m=-\ell}^{\ell} \overline{c_{n_1 \ell m}} c_{n_2 \ell m},$$
$$b_{n_1 n_2 \ell \ell_1 \ell_2} \triangleq \sum_{m=-\ell}^{\ell} \sum_{m_1=-\ell_1}^{\ell_1} \sum_{m_2=-\ell_2}^{\ell_2} \overline{c_{n_1 \ell m}}\, C^{\ell m}_{\ell_1 m_1 \ell_2 m_2} c_{n_2 \ell_1 m_1} c_{n_2 \ell_2 m_2}.$$

Finally, we note in passing that yet another possibility of capturing radial dependence presented in [62] is the use of four-dimensional hyperspherical harmonics as a basis for $\rho(\boldsymbol{x})$, giving rise to a four-dimensional power spectrum and bispectrum.

4.3 Defining measures of environment similarity

The way in which most descriptors of atomic environments are used is to assign a similarity between any given pair of them. This is typically taken to be a real number $\mathcal{S}$ which falls in the range $(\,0, 1\,]$, where environments which are identical possess similarity $\mathcal{S}$=1 and environments which are highly dissimilar will have values of $\mathcal{S}$ near 0. The ways in which $\mathcal{S}$ is defined depend on the particular descriptor being considered. The first type of descriptors presented in this work were based on bond lengths and bond angles and, as a result, were guaranteed to be translationally and rotationally invariant. However, because they are not additive, they are not invariant to permutations of the neighbor indices. Thus, in order to define a similarity measure between two environments characterized by these descriptors, it is necessary to consider all possible such permutations. In the case of the

Weyl matrix, for example, a natural choice for defining a measure of similarity between environments which possess corresponding descriptors $\mathbf{\Sigma}$ and $\mathbf{\Sigma}'$ is $\mathcal{S}_{\mathbf{\Sigma}} \triangleq 1/(1+d(\mathbf{\Sigma}, \mathbf{\Sigma'}))$, where

$$d(\mathbf{\Sigma}, \mathbf{\Sigma'}) \triangleq \min_{\mathbf{P}} ||\mathbf{\Sigma} - \mathbf{P}\mathbf{\Sigma}'\mathbf{P}^T||, \tag{4.80}$$

and the minimization is carried out over all permutation matrices $\mathbf{P}$.[14] The next category of descriptors we encountered were once again based on bond lengths and bond angles, but attained permutational invariance by means of additivity: the Behler–Parrinello symmetry functions and the AFS. As these descriptors satisfy all three desired types of symmetry (translational, rotational, permutational), a similarity measure $\mathcal{S}$ can thus be defined in terms of the normed difference of the vectors containing the descriptor components for each of the two environments, e.g. vectors containing the values[15] of the Behler–Parrinello symmetry functions $[G^1_\alpha, G^2_\alpha, G^4_\alpha, G^5_\alpha]$ for each environment. Proceeding further, it was suggested that an atomic environment be considered as a distribution in terms of an atomic neighborhood density function $\rho(\boldsymbol{x})$—an alternative method of enforcing permutational invariance. Although finding invariants of the geometric moments of $\rho(\boldsymbol{x})$ directly was found to be insurmountable, leveraging specific bases for ρ which consisted of explicit representations of SO(3) admitted the identification of descriptors which were analytically designed to be rotationally invariant. Once more, a similarity measure $\mathcal{S}$ between two environments can be defined using the normed difference of the corresponding descriptor vectors in a manner similar to the case of the Weyl matrix.

4.3.1 Smooth Overlap of Atomic Positions (SOAP)

We now review one final possibility of defining atomic environment similarity due to Bartók *et al.* [62] which is based on the use of an atomic neighborhood density function. In this approach, an *overlap* function S is defined between the densities $\rho(\boldsymbol{x})$ and $\rho(\boldsymbol{x}')$ by

$$S(\rho, \rho') \triangleq \langle \rho, \rho' \rangle = \int_{\boldsymbol{r} \in \mathbb{R}^3} \rho(\boldsymbol{r})\rho'(\boldsymbol{r})d\boldsymbol{r} = \int_{\boldsymbol{r} \in \mathbb{R}^3} \overline{\rho(\boldsymbol{r})}\rho'(\boldsymbol{r})d\boldsymbol{r}. \tag{4.81}$$

[14] Another common way of defining similarity here makes use of a squared exponential form, so that $\mathcal{S}_{\mathbf{\Sigma}} \triangleq \exp(-d(\mathbf{\Sigma}, \mathbf{\Sigma'})^2)$.

[15] As stated in Section 4.1.2, the symmetry function G^3_α is not used in practice.

The rightmost equality, which is convenient for simplifying analytical evaluation, follows trivially because $\rho(\boldsymbol{r})$ is real-valued. Further, let us define the function $k(\rho, \rho')$ by

$$\begin{aligned} k(\rho, \rho') &\triangleq \int\limits_{\mathcal{R}\in \mathrm{SO}(3)} |S(\rho, \mathcal{R}\rho')|^n d\mathcal{R} \\ &= \int\limits_{\mathcal{R}\in \mathrm{SO}(3)} \left| \int\limits_{\boldsymbol{r}\in\mathbb{R}^3} \rho(\boldsymbol{r})\rho'(\mathbf{R}^{-1}\boldsymbol{r}) d\boldsymbol{r} \right|^n d\mathcal{R}, \end{aligned} \tag{4.82}$$

where the integration over rotation operations $\mathcal{R}$ is accompanied by an integration over corresponding rotation matrices $\mathbf{R}$.

We now choose to expand the density in a basis of unnormalized Gaussians, each of width $a \triangleq 1/(2\sigma^2_{\mathrm{atom}})$, centered on each of the atoms (including the target atom itself). Modulating each Gaussian by a cutoff function $f_{\mathrm{cut}}(r)$, we write

$$\rho(\boldsymbol{r}) = e^{-ar^2} + \sum_{\beta=1}^{N_{\mathrm{neigh}}} e^{-a|\boldsymbol{r}-\boldsymbol{r}_\beta|^2} f_{\mathrm{cut}}(r_\beta) = \sum_{\beta=0}^{N_{\mathrm{neigh}}} e^{-a|\boldsymbol{r}-\boldsymbol{r}_\beta|^2} f_{\mathrm{cut}}(r_\beta). \tag{4.83}$$

Here, we follow the convention that the central atom is designated as a neighbor of itself with index $\beta = 0$ and $\boldsymbol{r}_0 = \mathbf{0}$, and define the cutoff function

$$f_{\mathrm{cut}}(r) \triangleq \begin{cases} 1 & 0 < r \leq (r_{\mathrm{cut}} - r_{\mathrm{trans}}) \\ \frac{1}{2}\left[\cos\left(\pi \frac{r - (r_{\mathrm{cut}} - r_{\mathrm{trans}})}{r_{\mathrm{trans}}}\right) + 1\right] & (r_{\mathrm{cut}} - r_{\mathrm{trans}}) < r \leq r_{\mathrm{cut}} \\ 0 & r > r_{\mathrm{cut}}, \end{cases} \tag{4.84}$$

where r_{cut} and r_{trans} are fixed parameters. As pointed out by Kaufmann and Baumeister [93], each individual Gaussian in (4.83) can be approximately expanded in a basis of spherical harmonics centered at the target atom up to some finite bandwidth ℓ_{max}:

$$e^{-a|\boldsymbol{r}-\boldsymbol{r}_\beta|^2} = \sum_{\ell=0}^{\ell_{\mathrm{max}}} \sum_{m=-\ell}^{\ell} c^\beta_{\ell m}(r) Y^m_\ell(\hat{\boldsymbol{r}}), \tag{4.85}$$

where

$$c^\beta_{\ell m}(r) = 4\pi e^{-a(r^2+r_\beta^2)} \iota_\ell(2arr_\beta) \overline{Y^m_\ell(\hat{\boldsymbol{r}}_\beta)}, \tag{4.86}$$

ι_ℓ is a regular modified spherical Bessel function [94], and we have introduced the notation $\hat{\boldsymbol{r}} = \boldsymbol{r}/||\boldsymbol{r}|| = (\theta, \phi)$ to indicate the point on the unit sphere obtained by projecting the

vector $\boldsymbol{r}$ onto it.[16] Applying the expansion of (4.85) to each Gaussian in (4.83), we finally write the density as

$$\rho(\boldsymbol{r}) = \sum_{\beta=0}^{N_{\text{neigh}}} \sum_{\ell=0}^{\ell_{\max}} \sum_{m=-\ell}^{\ell} c_{\ell m}^{\beta}(r) Y_{\ell}^{m}(\hat{\boldsymbol{r}}) f_{\text{cut}}(r_{\beta}). \tag{4.87}$$

Next, in order to calculate $k(\rho, \rho')$, we must compute $S(\rho, \mathcal{R}\rho')$, which we define as the function $\breve{S}(\mathcal{R})$:

$$\begin{aligned}
\breve{S}(\mathcal{R}) \triangleq S(\rho, \mathcal{R}\rho') &= \int \rho(\boldsymbol{r}) \rho'(\mathbf{R}^{-1}\boldsymbol{r}) d\boldsymbol{r} = \int \overline{\rho(\boldsymbol{r})} \rho'(\mathbf{R}^{-1}\boldsymbol{r}) d\boldsymbol{r} \\
&= \sum_{\beta,\beta'} \sum_{\substack{\ell,\ell' \\ m,m'}} \int \overline{c_{\ell m}^{\beta}(r) Y_{\ell}^{m}(\hat{\boldsymbol{r}})} c'^{\beta'}_{\ell' m'}(r) Y_{\ell'}^{m'}(\hat{\boldsymbol{r}}) d\boldsymbol{r} \\
&= \sum_{\beta,\beta'} \sum_{\substack{\ell,\ell' \\ m,m'}} \int \overline{c_{\ell m}^{\beta}(r)} c'^{\beta'}_{\ell' m'}(r) r^2 dr \int \overline{Y_{\ell}^{m}(\hat{\boldsymbol{r}})} Y_{\ell'}^{m'}(\hat{\boldsymbol{r}}) d\hat{\boldsymbol{r}}.
\end{aligned} \tag{4.88}$$

Here, the indices β, β' run from 0 to N_{neigh} and N'_{neigh}, respectively, while the indices ℓ, ℓ' run from 0 to $\ell_{\max}$; m (m') runs from $-\ell$ to ℓ ($-\ell'$ to ℓ'). In the last equality, we have switched to integrating in spherical coordinates, i.e. $d\boldsymbol{r} = r^2 \sin\theta dr d\theta d\phi$ and we have defined $d\hat{\boldsymbol{r}} = \sin\theta d\theta d\phi$. From the orthonormality of the spherical harmonics in (4.51), the rightmost integral is equal to $\delta_{\ell\ell'}\delta_{mm'}$ (where δ is the Kronecker delta), and substitution gives

$$\breve{S}(\mathcal{R}) = \sum_{\beta,\beta'} \sum_{\ell,m} \int \overline{c_{\ell m}^{\beta}(r)} c'^{\beta'}_{\ell m}(r) r^2 dr. \tag{4.89}$$

The consequence of the rotation $\mathcal{R}$ acting on $\rho'(\boldsymbol{r})$ is that it alters the coefficients $c'^{\beta'}_{\ell m}(r)$ so that

$$\begin{aligned}
c'^{\beta'}_{\ell m}(r) &= 4\pi e^{-a(r^2 + r_{\beta'}^2)} \iota_{\ell}(2arr_{\beta'}) \overline{\mathcal{R} Y_{\ell}^{m}(\hat{\boldsymbol{r}}_{\beta'})} \\
&= 4\pi e^{-a(r^2 + r_{\beta'}^2)} \iota_{\ell}(2arr_{\beta'}) \overline{Y_{\ell}^{m}(\mathbf{R}^{-1}\hat{\boldsymbol{r}}_{\beta'})} \\
&= 4\pi e^{-a(r^2 + r_{\beta'}^2)} \iota_{\ell}(2arr_{\beta'}) \sum_{m'=-\ell}^{\ell} \overline{D_{m'm}^{\ell}(\mathcal{R}) Y_{\ell}^{m'}(\hat{\boldsymbol{r}}_{\beta'})},
\end{aligned} \tag{4.90}$$

where we have exploited the behavior of spherical harmonics under rotation given by (4.62).

[16] Note from (4.86) that $c_{\ell m}^{0}(r)$ is equal to 0 unless $\ell = m = 0$, in which case $c_{00}^{0}(r) = 4\pi e^{-ar^2} Y_0^0 = 2\sqrt{\pi}\, e^{-ar^2}$.

Substituting this expression along with the expression for $c_{\ell m}^{\beta}(r)$ from (4.86) gives

$$\breve{S}(\mathcal{R}) = (4\pi)^2 \sum_{\beta,\beta'} e^{-a(r_\beta^2 + r_{\beta'}^2)} \sum_{\ell,m,m'} \overline{D_{m'm}^{\ell}(\mathcal{R})} \left(\int e^{-2ar^2} \iota_\ell(2arr_\beta) \iota_\ell(2arr_{\beta'}) r^2 dr \right) \times Y_\ell^m(\hat{\boldsymbol{r}}_\beta) \overline{Y_\ell^{m'}(\hat{\boldsymbol{r}}_{\beta'})}. \tag{4.91}$$

The value of the integral in parentheses is also given in [93],

$$\int e^{-2ar^2} \iota_\ell(2arr_\beta) \iota_\ell(2arr_{\beta'}) r^2 dr = \frac{1}{4} \sqrt{\frac{\pi}{8a^3}} \, \iota_\ell(2ar_\beta r_{\beta'}) \exp\left(\frac{a(r_\beta^2 + r_{\beta'}^2)}{2} \right), \tag{4.92}$$

so that finally we obtain

$$\breve{S}(\mathcal{R}) = \sum_{\beta,\beta'} \sum_{\ell,m,m'} \tilde{I}_{mm'}^{\ell}(a, r_\beta, r_{\beta'}) \overline{D_{m'm}^{\ell}(\mathcal{R})} = \sum_{\ell,m,m'} I_{mm'}^{\ell} \overline{D_{m'm}^{\ell}(\mathcal{R})}, \tag{4.93}$$

where we have defined

$$\tilde{I}_{mm'}^{\ell}(a, r_\beta, r_{\beta'}) \triangleq \sqrt{\frac{2\pi^5}{a^3}} \, \exp\left(\frac{-a(r_\beta^2 + r_{\beta'}^2)}{2} \right) Y_\ell^m(\hat{\boldsymbol{r}}_\beta) \overline{Y_\ell^{m'}(\hat{\boldsymbol{r}}_{\beta'})}, \tag{4.94}$$

$$I_{mm'}^{\ell} \triangleq \sum_{\beta,\beta'} \tilde{I}_{mm'}^{\ell}(a, r_\beta, r_{\beta'}). \tag{4.95}$$

Substituting the expression for $\breve{S}(\mathcal{R})$ in (4.93) into $k(\rho, \rho')$, where we take $n = 2$,

$$\begin{aligned} k(\rho, \rho') &= \int_{\mathcal{R} \in \mathrm{SO}(3)} \overline{\breve{S}(\mathcal{R})} \breve{S}(\mathcal{R}) dR \\ &= \sum_{\substack{\ell,m,m' \\ \lambda,\mu,\mu'}} \overline{I_{mm'}^{\ell}} I_{\mu\mu'}^{\lambda} \int_{\mathcal{R}} D_{m'm}^{\ell}(\mathcal{R}) \overline{D_{\mu'\mu}^{\lambda}(\mathcal{R})} dR \\ &= \sum_{\substack{\ell,m,m' \\ \lambda,\mu,\mu'}} \overline{I_{mm'}^{\ell}} I_{\mu\mu'}^{\lambda} \left(\frac{8\pi^2}{2\ell + 1} \delta_{\ell\lambda} \delta_{m\mu} \delta_{m'\mu'} \right) \\ &= \sum_{\ell,m,m'} \left(\frac{8\pi^2}{2\ell + 1} \right) \overline{I_{mm'}^{\ell}} I_{mm'}^{\ell}. \end{aligned} \tag{4.96}$$

Finally, a similarity is defined from $k(\rho, \rho')$ by normalizing it (cf. [95]) and taking it to a power ζ in order to arrive at what Bartók et al. [62] refer to as the *Smooth Overlap of*

Atomic Positions (SOAP) similarity kernel:[17]

$$k_{\mathrm{SOAP}}(\rho,\rho') \triangleq \left(\frac{k(\rho,\rho')}{\sqrt{k(\rho,\rho)}\sqrt{k(\rho',\rho')}}\right)^{\zeta}. \tag{4.97}$$

A practical issue with this expression pointed out in [62] is that the calculation of the quantity $I^{\ell}_{mm'}$ of (4.95) which appears in (4.96) requires a nested summation over the neighbors of each environment ($\beta = 0,\ldots,N_{\mathrm{neigh}}$ and $\beta' = 0,\ldots,N'_{\mathrm{neigh}}$), which scales poorly for environments containing a large number of neighbors. However, the authors address this problem by returning to the expansion of the density in spherical harmonics in (4.87) and making use of an orthonormal radial basis $g_n(r)|_{n=1}^{n_{\max}}$:

$$\rho(\boldsymbol{r}) = \sum_{n=1}^{n_{\max}}\sum_{\ell=0}^{\ell_{\max}}\sum_{m=-\ell}^{\ell} c_{n\ell m} g_n(r) Y_{\ell}^{m}(\hat{\boldsymbol{r}}). \tag{4.98}$$

Moreover, expand the density ρ' as

$$\rho'(\boldsymbol{r}) = \sum_{n'=1}^{n_{\max}}\sum_{\ell'=0}^{\ell_{\max}}\sum_{m=-\ell}^{\ell} c'_{n'\ell'm'} g_{n'}(r) Y_{\ell'}^{m'}(\hat{\boldsymbol{r}}), \tag{4.99}$$

and denote the coefficients corresponding to the rotated density $\mathcal{R}\rho'(\boldsymbol{r})$ as $(c'_{n'\ell'm'})^{\mathcal{R}}$. In this case, the overlap of $\rho(\boldsymbol{r})$ and $\mathcal{R}\rho'(\boldsymbol{r})$ becomes

$$\begin{aligned}
\breve{S}(\mathcal{R}) \triangleq S(\rho,\mathcal{R}\rho') &= \int \rho(\boldsymbol{r})\rho'(\mathbf{R}^{-1}\boldsymbol{r})d\boldsymbol{r} \\
&= \sum_{n,n'}\sum_{\substack{\ell,\ell'\\ m,m'}} \int \overline{c_{n\ell m} g_n(r) Y_{\ell}^{m}(\hat{\boldsymbol{r}})}\,(c'_{n'\ell'm'})^{\mathcal{R}} g_{n'}(r) Y_{\ell'}^{m'}(\hat{\boldsymbol{r}}) d\boldsymbol{r} \\
&= \sum_{n,n'}\sum_{\substack{\ell,\ell'\\ m,m'}} \overline{c_{n\ell m}}\,(c'_{n'\ell'm'})^{\mathcal{R}} \int g_n(r) g_{n'}(r) r^2 dr \int \overline{Y_{\ell}^{m}(\hat{\boldsymbol{r}})} Y_{\ell'}^{m'}(\hat{\boldsymbol{r}}) d\hat{\boldsymbol{r}} \\
&= \sum_{n,n'}\sum_{\ell,m} \overline{c_{n\ell m}}\,(c'_{n'\ell m})^{\mathcal{R}} \int g_n(r) g_{n'}(r) r^2 dr.
\end{aligned} \tag{4.100}$$

The coefficients $(c'_{n'\ell'm'})^{\mathcal{R}}$ of the rotated density $\mathcal{R}\rho'(\boldsymbol{r})$ are related to the coefficients $c'_{n'\ell'm'}$ of $\rho'(\boldsymbol{r})$ by

$$(c'_{n'\ell m})^{\mathcal{R}} = \langle g_{n'}(r)\mathcal{R}Y_{\ell}^{m}(\hat{\boldsymbol{r}}), \rho'(\boldsymbol{r})\rangle \tag{4.101}$$

[17] The term *kernel* will be clarified in Section 5.3.1. For now, it may be taken to be informally equivalent to the similarity measure $\mathcal{S}$.

$$= \sum_{m'=\ell}^{\ell} \overline{D^{\ell}_{m'm}(\mathcal{R})} \left\langle g_{n'}(r) Y_{\ell}^{m'}(\hat{\boldsymbol{r}}), \rho'(\boldsymbol{r}) \right\rangle$$
$$= \sum_{m'=\ell}^{\ell} \overline{D^{\ell}_{m'm}(\mathcal{R})} c'_{n'\ell m'}, \tag{4.102}$$

so that

$$\breve{S}(\mathcal{R}) = \sum_{n,n'} \sum_{\ell,m,m'} \overline{D^{\ell}_{m'm}(\mathcal{R})} \, (\overline{c_{n\ell m}} c'_{n'\ell m'}) \int g_n(r) g_{n'}(r) r^2 dr. \tag{4.103}$$

Defining

$$\tilde{I}^{\ell}_{m,m'}(n,n') \triangleq \overline{c_{n\ell m}} c'_{n'\ell m'} \int g_n(r) g_{n'}(r) r^2 dr, \tag{4.104}$$
$$I^{\ell}_{m,m'} \triangleq \sum_{n,n'} \tilde{I}^{\ell}_{m,m'}(n,n'), \tag{4.105}$$

we have, similar to before,

$$\breve{S}(\mathcal{R}) = \sum_{n,n'} \sum_{\ell,m,m'} \tilde{I}^{\ell}_{mm'}(a,n,n') \overline{D^{\ell}_{m'm}(\mathcal{R})} = \sum_{\ell,m,m'} I^{\ell}_{mm'} \overline{D^{\ell}_{m'm}(\mathcal{R})}, \tag{4.106}$$

so that, as before, the similarity between ρ and ρ' is given by (4.96). Now, notice that if the radial basis functions $g_n(r)$ are orthonormal, i.e.

$$\int g_n(r) g_{n'}(r) r^2 dr = \delta_{nn'}, \tag{4.107}$$

then

$$I^{\ell}_{m,m'} = \sum_{n} \overline{c_{n\ell m}} c'_{n\ell m'}. \tag{4.108}$$

and that

$$k(\rho,\rho') = \sum_{n,n'} \sum_{\ell,m,m'} \left(\frac{8\pi^2}{2\ell+1} \right) (c_{n\ell m} \overline{c'_{n\ell m'}}) (\overline{c_{n'\ell m}} c'_{n'\ell m'})$$
$$= \sum_{n,n'} \sum_{\ell} \left(\frac{8\pi^2}{2\ell+1} \right) \sum_{m,m'} (\overline{c_{n'\ell m}} c_{n\ell m}) (\overline{c'_{n\ell m'}} c'_{n'\ell m'})$$
$$= \sum_{n,n'} \sum_{\ell} \left(\frac{8\pi^2}{2\ell+1} \right) p_{nn'\ell} p'_{nn'\ell}, \tag{4.109}$$

whence we see that $k(\rho,\rho')$ for $n = 2$ is given by the dot product of the power spectra of ρ and ρ',[18] weighted by the factor $(8\pi^2/(2\ell+1))$. In practice, it is the expression

[18] In the way of terminology, the quantity $k(\rho,\rho')$ for $n = 2$ could be said to be the *cross-power spectrum*

in (4.109) which is used in (4.97), where the power spectra $p_{nn'\ell}$ are determined according to the prescription in Appendix B. Aside from [62], applications of the SOAP kernel can be found in [96, 97].

of the densities ρ and ρ'. Similarly, the argument in parentheses shown in (4.97) is sometimes referred to as the *coherence* of the distributions ρ and ρ'.

Chapter 5

Machine Learning Perspectives of Empirical Potentials

In Chapter 2, a taxonomy of traditional empirical potentials (EPs) was presented which divided them into four groups: pair potentials, pair functionals, cluster potentials, and cluster functionals. However, it is possible to characterize EPs in a manner which subsumes this categorization by placing them in the broader context of the theory of *machine learning* [98].[1] The field of machine learning consists of the development of adaptive, heuristic algorithms motivated by applied statistics to solve modern data science problems, and consists of two primary topics: *unsupervised learning* and *supervised learning*. In each topic, a *training set* of data is considered within which each instance has an associated *training input* (oftentimes a vector of real numbers), but the two are differentiated by the presence of output values known as *training objectives* which accompany each instance in the latter case. Whereas in unsupervised learning the objective is to identify patterns among the inputs of the wider population represented by the training set, e.g. clustering into distinct groups, the goal of supervised learning is to infer a mapping which transforms the inputs of the population to their corresponding outputs.

Supervised learning is further separated into two subcategories. In *classification*, the function which maps the inputs of the population to their associated outputs is assumed to be discrete. In terms of the training set, this means that each instance consists of a training input along with an objective which takes the form of a label designating it as belonging to

[1] There are several other fields which share a great deal of overlap with machine learning, including *information theory* [99], *statistical learning theory* [100], *pattern recognition* [101], and *artificial intelligence* [102].

one of a number of discrete *classes*. To the contrary, in *regression* the outputs corresponding to the members of the population are allowed to assume a continuous range of values. Because EPs can be defined abstractly as apparati which take as input the Cartesian positions of an atomic configuration and provide as output the total internal potential energy of the system, they may thus be regarded as algorithms which attempt to perform regression on the Born–Oppenheimer PES (BOPES). Accordingly, we describe in this chapter the three general types of regression methods found in machine learning, discussing the unique qualities inherited by EPs which are constructed using each technique.

5.1 Parametric methods

5.1.1 Functional forms

The type of regression most familiar to the materials modeling community involves the use of explicit, analytical functional forms. Starting from the Cartesian positions of an atomic configuration, conventional *parametric empirical potentials* begin by calculating a corresponding set of bond lengths and bond angles between atoms which fall within a prescribed cutoff distance of one another (see Section 4.1.1). Supplementing this geometric description with the atomic species and charge states, closed functional forms containing a fixed number of parameters are evaluated in order to render a prediction of the total internal potential energy and atomic forces. As there are no restrictions on the positions of the atoms, this procedure can in principle be used to construct an approximation of any desired portion of the BOPES.

Using the bond lengths and bond angles as the arguments to a parametric EP holds appealing simplicity compared to the alternative representations surveyed in the previous chapter; because they are a *complete* descriptor, i.e. a perfectly faithful representation, there is no possibility of losing or corrupting the information of each atomic environment. However, the significant task of defining the functional form of a parametric potential which makes use of this descriptor remains. Insofar as condensed matter physics is concerned, perhaps the most notable period of innovation in parametric EP development occurred in the 1980s. It began with the introduction of the quasiatom method of Stott and Zaremba [103] and the Effective Medium Theory (EMT) by Nørskov and Lang [104–107], which subsequently gave rise to the Embedded-Atom Method (EAM) of Daw and Baskes [46, 47, 108]. Following soon thereafter was the work of Finnis and Sinclair [109], the empirical bond order potentials of Abell [110], and the Glue Potentials of Ercolessi *et al.* [111, 112]. Thanks in part to the taxonomy of Carlsson [41], the forms of each of these models have come to

be understood as closely resemblant of one another, with all of them corresponding to pair functionals. However, despite the fact that all of these frameworks were developed more or less concurrently and share similar forms, it is in retrospect remarkable that they did not all share a common physical origin. The premise of EMT, which serves as the foundation of the EAM and glue potentials, is to ascribe to each atom α an energy which is defined as the change in energy which it induces when placed in a simplified ("effective") host medium, which is usually taken to be a homogeneous electron gas whose density ϱ_α is determined by the neighborhood of atom α. As shown by Daw [48], this ansatz can be justified theoretically from the local density approximation (LDA) within DFT, and has stood the test of time as its validity continues to fall under scrutiny to the present day [113]. On the other hand, potentials of the Finnis–Sinclair or bond order varieties are based on the tight-binding approximation[2] to quantum mechanics, specifically the second-order truncation of the moment expansion of the electronic density of states. Additional literature on these classes of potentials can be found in various review articles and references therein [115–119].

While the above pair functional frameworks each have a sound physical basis, they are insufficient in many materials of interest where angular effects are important, e.g. where covalent bonding is present. Rather than adhering to rigorous derivation from quantum mechanics, these methods are most often extended to include angular dependence by making modifications to them which are decidedly heuristic in nature. The most widely proliferated such extension is the Modified Embedded-Atom Method (MEAM) of Baskes [120–122], which is formulated by superposing additional "partial" background electronic densities onto the original EAM density ϱ_α; these partial densities have no obvious physical motivation, effectively corresponding instead to a moment expansion of the atomic neighborhood density function $\rho(\boldsymbol{r})$ [92]. Extensions of the bond order potentials are handled with similar empiricism, beginning with the pair functional derivation of Abell and modifying the bond order term with ad hoc functional forms which reproduce experimental or first-principles data which capture the orientation dependence of bond energy.

As an illustration of the sort of reasoning used to construct an empirical potential which includes angular dependence, consider the three-body cluster functional of Tersoff (T2)[3] for silicon given in Appendix C, whose form is defined as

$$\mathcal{V} = \frac{1}{2} \sum_{\alpha} \sum_{\beta \neq \alpha} f_C(r_{\alpha\beta}) [f_R(r_{\alpha\beta}) - b_{\alpha\beta} f_A(r_{\alpha\beta})], \tag{5.1}$$

[2]Introductory material for the tight-binding approximation may be found in [5], and a more detailed discussion is given in [114].

[3]The relationship between the Tersoff potential and the EAM form was presented by Brenner in [123].

where

$$f_R(r) = A\exp(-\lambda_1 r), \tag{5.2a}$$

$$f_A(r) = B\exp(-\lambda_2 r), \tag{5.2b}$$

$$b_{\alpha\beta} = (1 + \delta^n \zeta_{\alpha\beta}^n)^{-1/2n}, \tag{5.2c}$$

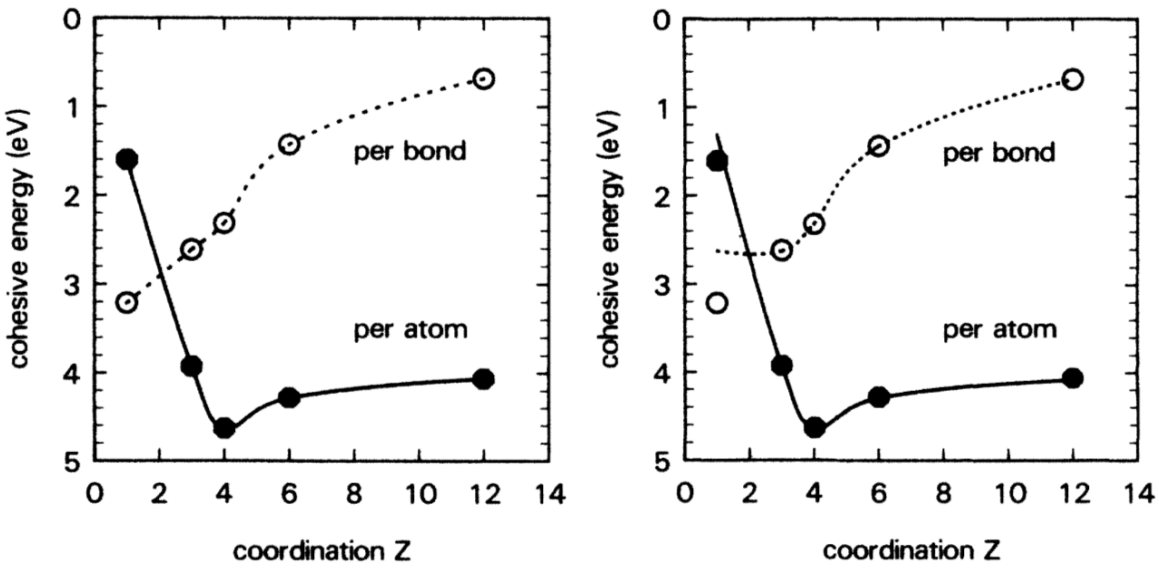
$$\zeta_{\alpha\beta} = \sum_{\gamma\neq\alpha,\beta} f_C(r_{\alpha\gamma}) g(\theta_{\beta\gamma}) \exp(\lambda_3^3 (r_{\alpha\beta} - r_{\alpha\gamma})^3), \tag{5.2d}$$

$$g(\theta) = 1 + \frac{c^2}{d^2} - \frac{c^2}{d^2 + (h - \cos\theta)^2}, \tag{5.2e}$$

and f_C is a smooth cutoff function. As Tersoff explains in the original publication of this potential [124], its derivation begins by starting from a generalized Morse pair potential [125], as can be seen from the exponential forms of f_A and f_R. The reasoning for this choice is rooted in the fact that, as shown by Abell [110] in his original publication on bond order, the Morse form closely reproduces the Universal Binding Energy Relation (UBER) of Rose *et al.* [126–128]. The UBER form is based on the exponential decay of the electron density experienced as one proceeds away from any of the nuclei of an atomistic system into vacant space [129]. Regardless of the specific chemical elements present, this fundamental tendency appears to be common to nearly all systems bound by metallic or covalent bonding for a wide range of geometric configurations they assume. Thus, the ability to approximate the UBER form to reasonable accuracy is considered by many to be a first-order requirement for any EP intended to model such systems.[4]

In order to introduce coordination dependence, the bond order term $b_{\alpha\beta}$ is added as a modulating factor to the attractive portion of the pair potential, f_A. In determining a suitable form for $b_{\alpha\beta}$, Tersoff extracted the cohesive energy per atom and per bond of various high-symmetry structures reported within the LDA-DFT data of Yin, Cohen, and Chang [131–134] and computed the average values across all of the structures as a function of coordination. The results of Tersoff's compilation, reproduced in Figure 5.1a, reveal a surprising degree of simplicity and guide the design of $b_{\alpha\beta}$ as follows: If the cutoff function f_C is chosen to include only first-nearest neighbors and $\lambda_1 \approx 2\lambda_2$ in (5.2a) and (5.2b), then taking $b_{\alpha\beta} \propto Z^{-1/2}$ (where Z denotes coordination) results in a cohesive energy per atom which is approximately independent of coordination, in agreement with the weak dependence of the corresponding curve in Figure 5.1a for $Z \geq 3$. Ergo, the form of $b_{\alpha\beta}$ should become proportional to $Z^{-1/2}$ as Z becomes large. However, albeit the coordination dependence of the

[4] Still, others argue that reproducing the behavior of UBER should not constitute a strict requirement on a potential, as it becomes an inaccurate description for highly compressed structures [130].



(a) (b)

Figure 5.1: *Reprinted from* [124] *with permission. Copyright 1988 by the American Physical Society.* (a) Cohesive energy vs. coordination compiled by Tersoff from the LDA-DFT calculations published by Yin, Cohen, and Chang [131–134]. The energies were averaged over data for the following high-symmetry structures: dimer, diamond, graphite, face-centered cubic, and simple cubic. The solid and dashed lines were created using spline fits solely as a guide for the eye. (b) The cohesive energy per atom and per bond as a function of coordination for the final parametrization used by Tersoff in [124] (whose parameters are given in Appendix C).

per-atom energy is weak for $Z \geq 3$, one must still account for the local minimum observed in the LDA-DFT data in this regime. Accordingly, Tersoff chose the form given in (5.2c), which allows for a bond order that grows more quickly than $Z^{-1/2}$ with decreasing Z in the range $Z \geq 3$ (as governed by the exponential parameter n) and converges smoothly but abruptly to unity for $Z < 3$. The saturation of the bond order at low coordination indicates that the energy per bond remains constant, as the bonds are not considerably affected by one another. But because the number of bonds increases more rapidly than the number of atoms as coordination increases, the result is a local minimum which occurs in the energy per atom, consistent with the DFT results. Altogether, the cohesive energy per atom and per bond calculated using the final form and parameters of the T2 potential are shown in Figure 5.1b, which deviate only from the first-principles data in the energy per bond at very low coordinations which, in the case of silicon, would presumably only occur for atoms in small clusters or atop surface terminations. Finally, although we defer the matter of fitting EP parameters momentarily, we further note that those of the T2 potential are chosen such that tetravalent coordination is most energetically favorable (as is observed in the DFT

data in Figure 5.1a), a quality necessary (but not sufficient) if the diamond ground state of silicon under ambient conditions is to be correctly predicted.

Finally, an aspect of the T2 potential not conveyed in Figure 5.1 is that the bond order is not written as a function of the conventional coordination Z, depending instead on an effective coordination ζ which incorporates angular dependence into the potential. Using the effective coordination serves two purposes. First, the function $g(\theta)$ present in the definition of ζ in (5.2d) allows the potential to explicitly specify an energetically preferable bond angle through the parameter h, which then allows particular geometries to be favored and thus allows the diamond lattice to be stabilized against shear.[5] Second, it enables the bond order to be weighted for each bond angle according to the lengths of the two bonds involved by means of the exponential factor in (5.2d). Roughly speaking, this means that the bond order can be tuned so that shorter bonds will not be affected as greatly by longer bonds which are formed from a common atom as vice versa. However, while the purpose of the effective coordination is clear, the precise form given in (5.2d) and (5.2e) was chosen by Tersoff on empirical grounds rather than chemical principle.

Shortcomings of the Tersoff form have been communicated by Pettifor *et al.* [135–139], who pointed out that it is incapable of correctly discerning the energy differences between some structures, and includes no contribution which accounts for π bonds (which are important for surface reconstructions in silicon, as well as for the hydrocarbon systems to which the Tersoff potential was extended by Brenner [140]). However, it was also uncovered in [135] that despite its informal derivation, the Tersoff formalism nonetheless corresponds to a low-order approximation to tight-binding (see also [115, 141]). In the same publications, the authors put forth an analytical bond order potential which, unlike those derived from Abell's work, includes higher-order moments of the electronic density of states [142]. Unfortunately, as pointed out by Kress and Voter [143], this expansion is slow to converge as more terms are successively included, marginalizing its benefit over direct application of more rigorous ab initio methods such as DFT. Therefore, empirical potentials such as Tersoff's continue to be used in the majority of large-scale applications,[6] and their numerous successes signify the efficacy of physical intuition in designing potentials.

[5] Although the favored bond angle is 90° in the T2 parametrization, most subsequent EPs for silicon inspired by the Tersoff formalism maximize the effective coordination for the tetrahedral bonding angle $\cos^{-1}(-1/3) \approx 109.5°$ found in the diamond structure. However, some have argued that this may act to reduce the transferability of the resulting potentials by emphasizing the diamond structure too stringently [115].

[6] Tersoff's potentials (T2 [124] and T3 [144]) have been cited in over 3,500 publications.

5.1.2 Parameter selection

Specifying the functional form of a parametric EP defines its physical foundation and is the most dominant factor of its qualitative predictions. However, fitting its parameters is also critical in determining its ability to yield a sensible approximation of the BOPES. Consistent with the philosophy of their functional forms, EPs such as the Tersoff potential described above are traditionally fitted to reproduce material properties which have a definite physical interpretation. These properties may be static in nature, e.g. zero-temperature lattice constant and bulk modulus, or might instead relate to dynamical quantities such as thermal expansion coefficients or melting temperature. At first glance, these two types of properties may seem disparate. Dynamical material properties appear far more abstruse than static properties because the trajectory of an atomistic system moves erratically across its configuration space under the influence of temperature. Static properties, meanwhile, are thought of intuitively in terms of how they are typically calculated, i.e. as regular paths on the BOPES which are explored by gradient-based optimization methods. However, common to both static and dynamic properties is the fact that they are related to *connected* domains of the energy landscape; the practical distinction between the two is that the former can be associated with a single local basin of attraction, whereas the latter may generally correspond to multiple basins which are traversed by the system. The continuous nature of these properties, which we henceforth term *canonical material properties*, is precisely what lends them their physical interpretability. The connectedess of the paths along the BOPES related to canonical properties (and, consequently, the connectedness of the relevant basin(s)) accords with the physical notion that atomic forces are the fundamental quantities which act to equilibrate any atomistic system. If the energy landscape is assumed to be smooth, its gradients will also be smooth, and the atomic forces will always act to evolve the system along continuous trajectories in configuration space.

In Figure 5.2, we show the overall procedure used in training and applying a parametric EP. First, a training set containing N_t instances is constructed and initial values for the $N_{\mathcal{P}}$ parameters of the potential are chosen. After using the EP defined with the initial parameter set to compute predictions corresponding to the training inputs, they are compared to the known objective values in the training set. If the accuracy of the potential meets some well-defined performance criteria, the training set is discarded and the parameter set (labeled $\mathcal{P}^*$ in Figure 5.2) is stored. Conversely, if the predictions made by the potential are not sufficiently accurate, the parameters $\mathcal{P}$ are adjusted and the above procedure is repeated until parameters are found for which the performance of the EP is deemed satisfactory, at which point the training set is discarded and the final parameters $\mathcal{P}^*$ stored as before. In

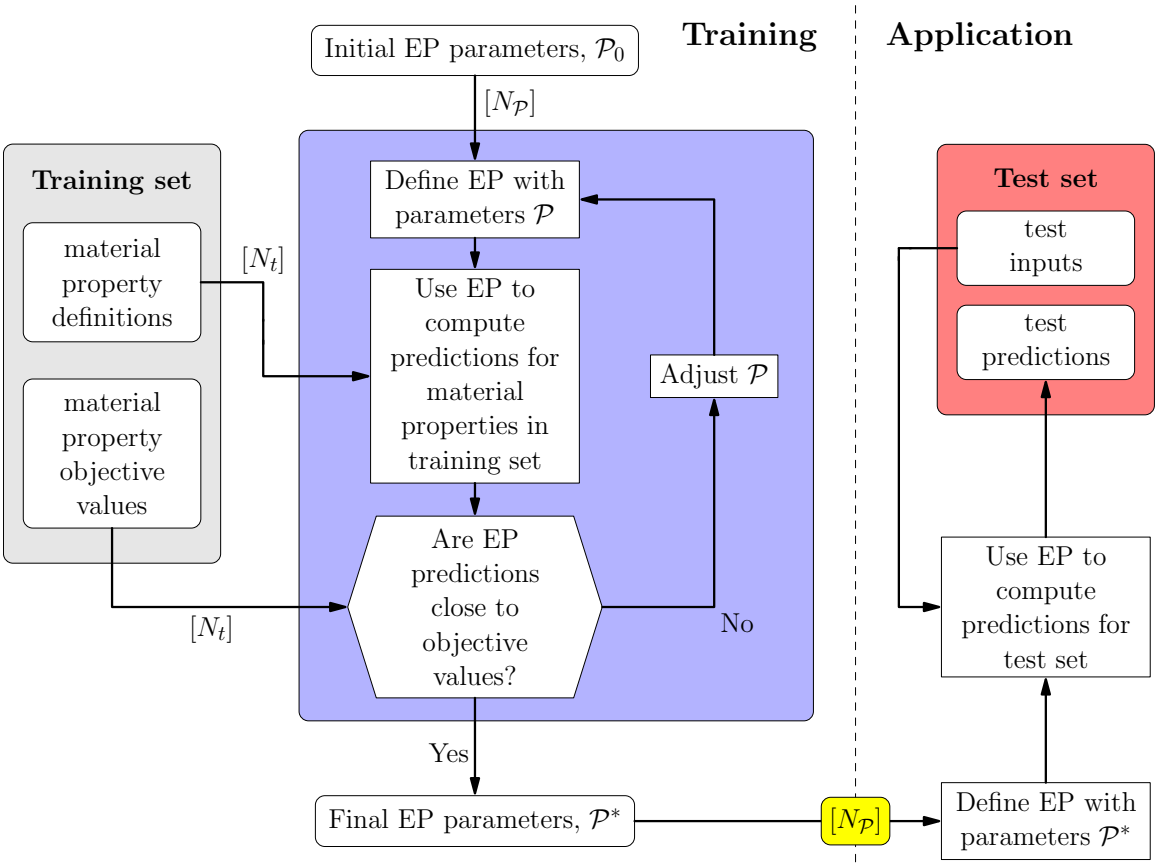


Figure 5.2: Training process of traditional parametric EPs and how they are used make predictions. Only the final parameter set $\mathcal{P}^*$ of size $N_\mathcal{P}$ determined from the training process is retained for making predictions in application.

the parlance of machine learning, the EP with its final parameters can then be applied to a *test set* consisting of inputs which were not included in the training set. We point our discussion next to the design choices which must be made in order to implement the fitting procedure outlined above: (1) the contents of the training set must be selected, including both the number and variety of training instances, (2) the performance assessment criteria must be identified, and (3) the optimization procedure used to adjust the parameters $\mathcal{P}$ must be specified.

Choosing a training set to fit the parameters of a parametric EP is contingent on its intended usage scenarios. Potentials designed for use in specific problems are typically fitted primarily, if not exclusively, to the properties considered most relevant to the intended application. The Stillinger–Weber potential for silicene [44, 145] listed in Appendix C is one such example, having been fit only to its (buckled) planar geometry and phonon dispersion curves rather than any properties of the bulk phases of silicon or their defects, etc. Likewise, an EP

intended to study epitaxial processes would be fit to properties such as surface formation energies and adsorption energies. Still, there are many potentials which are designed for general application and are accordingly fitted to properties which are conjectured to be central in some way in dictating a particular material's behavior. Such properties most often pertain to those structures which possess the lowest energies, and are thus most frequently observed in practice. For example, correctly predicting the geometry, energy, and elastic properties of the ground state of a material are usually considered fundamental requirements when determining the parameters of an EP intended for general use. A more nonlocal description of the energy landscape can be obtained by including properties related to each of the experimentally observed bulk structures, such as how the cohesive potential energy changes as a function of intrinsic volume. These properties can carry a surprising wealth of information related to the bonding of the material in question; in fact, it was shown by Bazant and Kaxiras [146] that a suitable inversion procedure can be used to derive a parametric EP for silicon solely from this type of training data which is comparable to the Tersoff potential aforementioned. However, it is not uncommon for the training set of general-use parametric EPs to additionally include dynamical properties, which might range in complexity from simple quantities such as dimer oscillation frequency [147] to the considerably more complicated examples of phonon dispersion curves [148], melting temperature [149], and even entire boundaries on the pressure–temperature phase diagram [150, 151]. Broadly speaking, canonical material properties are used in fitting parametric EPs because they provide a compact description of the overall shape of the most physically important portions of the energy landscape (and are hence available from experiment in many cases), with dynamical properties effectively describing a larger diversity of atomic configurations than static properties. For application-specific potentials, this helps to ensure transferability within the regime of configuration space pertinent to the application at hand. In the case of general parametric EPs, incorporating a host of canonical properties into the training set allows them to capture the prevailing topographic features of the energy landscape across a large portion of the region of configuration space which is likely to be occupied in a majority of applications, typically at the cost of diminishing their accuracy in any particular one of them.

During the fitting process, assessment of the predictions of a parametric EP with a set of candidate parameters $\mathcal{P}$ for a training set is usually carried out by defining a single *fitness function* $f(\mathcal{P})$ similar to

$$f(\mathcal{P}) \triangleq \sum_{i=1}^{N_t} w_i \left[A_i^{\mathrm{calc}}(\mathcal{P}) - A_i^{\mathrm{ref}}\right]^2, \tag{5.3}$$

where $A_i^{\mathrm{calc}}(\mathcal{P})$ are the objective values computed using the parametric EP with parameters $\mathcal{P}$, A_i^{ref} are the corresponding training objectives, and w_i are a set of weights which alter the relative influence of each training set instance in the fitting process. Thus, assessing parametric EP performance reduces in most cases to a single criterion: if $f(\mathcal{P})$ does not exceed some chosen threshold f^*, then the parameters $\mathcal{P}$ are considered satisfactory and the fitting process is terminated. However, despite the simplicity of the assessment criterion, adjusting the parameters of a parametric EP during fitting in the typical case where $f(\mathcal{P}) > f^*$ is far from trivial. While there have been efforts to help automate this process using genetic algorithms [152, 153], parameter optimization is difficult because it is unclear in many cases what precise effect each parameter will have in determining the various canonical material properties in the training set, particularly if they are dynamical properties related to multiple regions of the BOPES. Further compounding the difficulty of the problem is the fact that the parametric EPs discussed thus far typically only have around a dozen or so parameters at most, restricting their flexibility and therefore their ability to simultaneously fit each of the training set instances to arbitrary precision.[7] A common approach to the fitting process is to first fit the simplest parts of a potential and incrementally add the functions and parameters of its more elaborate constituents [154]. For example, one might begin by fitting the two-body terms of a many-body parametric EP to the properties of dimers and/or other small clusters, proceeding thereafter to fit the parameters of its remaining terms and/or embedding functional to more complex properties related to bulk structures or defects. However, while this allows some level of systematization, the difficulties mentioned above more often than not necessitate a manual course of trial-and-error optimization in which the developer must ascertain the physical significance of the parameters of their potential and use chemical intuition to locate a set of parameters $\mathcal{P}^*$ which result in a reasonable compromise of accuracy between the properties present in the training set in order to satisfy $f(\mathcal{P}^*) < f^*$.

In 1994, Ercolessi and Adams [155] proposed a novel method of fitting potentials which helps to mitigate the difficulty imparted to the fitting process by the complicated nature of canonical material properties. Rather than dealing exclusively with canonical properties, which are related to entire segments of the BOPES, the core proposition behind their solution is to use *point observations* on the energy landscape as training instances. That is, the training set is chosen to include quantities which describe the BOPES at individual points in configuration space. Naturally, there are only a few types of information which exist at

[7] The restricted flexibility of the functional forms of parametric EPs is a direct consequence of their physical motivation. While it makes fitting them arduous, it is also beneficial in the sense that it prevents overfitting. We will return to this point later in the chapter when we discuss the *bias–variance dilemma*.

a single point on the energy landscape, namely the value of the energy itself and its derivatives with respect to the atomic coordinates.[8] Moreover, because second-order derivatives are already prone to significant noise in first-principles calculations, usually only the energy and its first gradients (which are the negative of the atomic forces) are retained. This leads to a general fitness function which divides the N_t training instances into two separate sums, one of which takes place over the N_{CMP} canonical material properties in the training set and the other of which takes place over the $N_{\mathcal{C}}$ point observations in the training set:[9]

$$\breve{f}(\mathcal{P}) \triangleq \sum_{i=1}^{N_{\mathrm{CMP}}} w_i \left[A_i^{\mathrm{calc}}(\mathcal{P}) - A_i^{\mathrm{ref}}\right]^2 + \sum_{\mathcal{C}=1}^{N_{\mathcal{C}}} \left(\breve{w}_{\mathcal{C}} \left[E_{\mathcal{C}}^{\mathrm{calc}}(\mathcal{P}) - E_{\mathcal{C}}^{\mathrm{ref}}\right]^2 + \breve{\breve{w}}_{\mathcal{C}} \sum_{\alpha=1}^{\mathcal{A}_{\mathcal{C}}} \left\| \mathbf{F}_{\mathcal{C}\alpha}^{\mathrm{calc}}(\mathcal{P}) - \mathbf{F}_{\mathcal{C}\alpha}^{\mathrm{ref}} \right\|^2 \right). \tag{5.4}$$

Here, $E_{\mathcal{C}}$ is the total internal potential energy of configuration $\mathcal{C}$, $\mathcal{A}_{\mathcal{C}}$ is the number of atoms in configuration $\mathcal{C}$, $\mathbf{F}_{\mathcal{C}\alpha}$ is the net force vector acting on atom α of configuration $\mathcal{C}$, $\hat{w}_{\mathcal{C}}$ and $\breve{\breve{w}}_{\mathcal{C}}$ are weights which may be assigned to the energy and force terms, and $N_{\mathrm{CMP}} + N_{\mathcal{C}} = N_t$. Because the seminal publication of Ercolessi and Adams focused primarily on introducing the force term in (5.4) rather than the energy term, the general strategy of using a fitness function of this sort was first known as the *force-matching method* [156].[10] Examples of common parametric EPs which have been trained using point observations include specific parametrizations of the Tersoff potential [159, 160], the Stillinger–Weber potential [161], and the Sutton–Chen potential [162].

Apart from the inclusion of the point observations in the training set and the augmentation of the parameter adjustment step to include contributions from energy and force errors, the fitting process remains essentially unchanged from Figure 5.2 when using the fitness function of (5.4), i.e. there is no abstract difference in the fitting procedure. However, an important practical distinction is the dramatic increase in the volume of the training set which arises from the fact that the cost of computing the total energy and atomic forces of an individual configuration is insignificant, allowing a more or less arbitrary number of

[8] Assuming, as we have thus far, that the BOPES is analytic over its entire domain and, consequently, that all derivatives exist.

[9] In some cases, the terms in the sum over configurations in the fitness function are normalized to prevent configurations which have many atoms from contributing more than those which have fewer atoms. This amounts to dividing the energy term by $\mathcal{A}_{\mathcal{C}}$ and dividing the force term by $\sum_{\mathcal{C}=1}^{N_{\mathcal{C}}} \mathcal{A}_{\mathcal{C}}$. Sometimes, as in the original force-matching publication, the force term is divided by $3\sum_{\mathcal{C}=1}^{N_{\mathcal{C}}} \mathcal{A}_{\mathcal{C}}$ as a pseudo-average over the force components. In any case, these normalizations can be absorbed into the weights $\breve{w}_{\mathcal{C}}$ and $\breve{\breve{w}}_{\mathcal{C}}$.

[10] They did, however, also include canonical material properties in the training set they used to construct the prototype "force-matching" potential for aluminum presented in [155]. The reader is also referred to the two follow-up publications to the original [157, 158].

training instances to be included. Supplementing the training set with this abundance of information accordingly enables the use of an expanded variety of optimization techniques which, when provided with canonical properties alone, lack enough constraints to function effectively. By utilizing these methods, automation of the fitting process is made a closer reality. For example, in [163], the authors present a systematic method of fitting MEAM potentials. Furthermore, parametric EP fitting algorithms which make use of canonical properties and point observations have been introduced in the POTFIT package[11,12] [164–166] and demonstrated in the literature [167].

5.2 Semiparametric methods

5.2.1 Tabulated potentials

In addition to easing the automation of the fitting process, the access to a vast amount of training data in the form of point observations also enables alternative paradigms to handle the process of developing functional forms for EPs. For example, it has been proposed that point observations can be used with genetic algorithms to breed an analytical functional form for a parametric EP [168–170]. However, a more common use of the combined mass of point observations and canonical properties which was demonstrated in the original force-matching publication of Ercolessi and Adams [155] is the creation of what are known as *tabulated empirical potentials* (TEPs). Like parametric potentials, tabulated EPs adhere to an overarching functional form which ultimately uses the bond lengths and angles as a descriptor. However, they are distinguished by the inclusion of one or more functions which underlie the overall form, but are not expressed analytically. Instead, these functions consist of a table of their values at specific sampling points of their respective domains, along with an interpolation method which "fills in the gaps" between these values. Here, we shall refer to such sampling points as *interpolative inputs* and the corresponding function values at these points as *interpolative objectives*. Furthermore, because the act of interpolation per se generally requires its own training procedure, we term the union of all interpolative inputs and objectives the *interpolative training set*, while we refer to the collection of point observations and/or canonical material properties used for the overall fitting of an EP as the *primary training set*.

The aforementioned EAM pair functional is one overarching form which is frequently used with tabulated subordinate functions. Recalling the definition of a pair functional, where

[11] http://www.potfit.net/wiki/doku.php
[12] https://github.com/potfit/

the energy is given by

$$\mathcal{V} = \frac{1}{2}\sum_{\substack{\alpha,\beta \\ \alpha\neq\beta}} \mathcal{V}_2(r_{\alpha\beta}) + \sum_{\alpha} U_\alpha[\varrho_\alpha^{\mathrm{PF}}], \qquad \varrho_\alpha^{\mathrm{PF}} = \sum_{\substack{\beta \\ \beta\neq\alpha}} g(r_{\alpha\beta}), \tag{5.5}$$

these potentials thus consist of tabulated forms for the density function $g(r)$, the embedding functional $U[\varrho^{\mathrm{PF}}]$, and in many cases the two-body interaction $\mathcal{V}_2(r)$ (although it is sometimes defined analytically). The interpolation itself is usually carried out with the aid of *splines* [171]: functions which span an interval $x \in [a, b]$ using n piecewise polynomials defined on subintervals $a = x_0 < x_1 < x_2 < \cdots < x_n = b$. In this case, the interpolative inputs correspond to the "knots" of the spline, at which its values are required to match the values of the interpolative objectives.[13] Requirements imposed on the continuity of the derivatives of the piecewise polynomials at the subinterval boundaries define the degree of the spline, and lead to a simple linear system of equations which can be solved to acquire the coefficients of each of the polynomials, fully specifying the interpolative fit to the data. The interpolative training process to perform spline fits is thus usually trivial in terms of both time and complexity. However, the interpolative inputs/objectives are a direct component of the splines thus defined, and must be retained in order to evaluate them at arbitrary points of their domains as is necessary in application.[14]

Figure 5.3 shows the general fitting process used with a spline-based TEP. In addition to canonical material properties, the primary training set now includes point observations, i.e. atomic configurations and corresponding objectives describing their total energy and/or forces. The set $\mathcal{P}$ now contains the parameters of any closed functional subforms which exist inside of the EP, as well as the extent associated with every function of the potential. For any functions which depend on a spatial distance r, these parameters correspond to their cutoff distances; for all other functions, they specify the compact support over which they are tabulated (extension beyond the range of tabulation is usually conducted with linear extrapolation). Further contained in $\mathcal{P}$ are the number of spline knots in each tabulated function, as well as the degree of the splines employed. The training process begins by first using the cutoff-type parameters and spline knot counts for each tabulated function found in $\mathcal{P}_0$ to create an initial interpolative training set for each tabulated function of the potential, collectively designated $\mathcal{I}_0$. Next, the (independent) training subprocesses which produce an interpolative fit of each tabulated function based on $\mathcal{I}_0$ are executed; this con-

[13] While the subinterval boundaries $x_0, \ldots, x_n$ are often taken to coincide with the knots for splines composed of odd-degree polynomials, they are usually chosen as the midpoints of the knots for those consisting of even-degree polynomials.

[14] The evaluation of splines is typically done according to the `splint` function given in [172, Section 3.3].

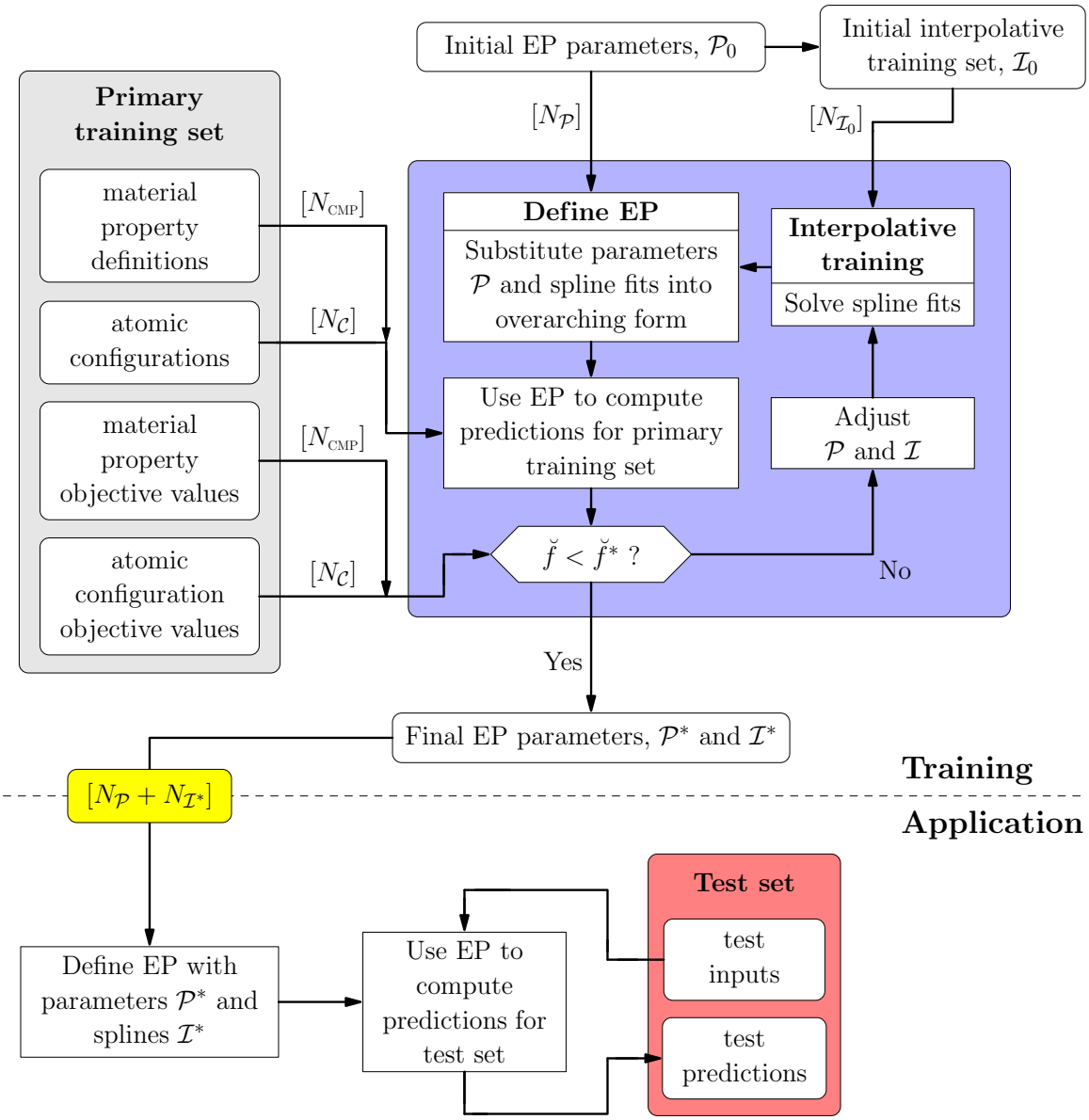


Figure 5.3: Training process of spline-based TEPs and how they are used to make predictions. Both the final parameter set $\mathcal{P}^*$ of size $N_{\mathcal{P}_0}$ and the final interpolative training set (contained in the definition of the splines of the tabulated subforms) of size $N_{\mathcal{I}^*}$ are retained for making predictions in application.

sists of solving the linear systems associated with the splines of each tabulated function. At this point, substituting the remaining parameters of $\mathcal{P}$ into any analytical functional forms of the potential which might exist and combining them with the spline fits of the previous step fully defines the overarching form of the TEP with all of its initial parameters and analytical/tabulated functional subforms. The potential is then used to compute predictions for each of the canonical properties and atomic configurations in the primary training set, and

the accuracy of the predictions is evaluated using the fitness function $\breve{f}$ of (5.4), which now formally depends on $\mathcal{I}$ as well as $\mathcal{P}$. For simplicity, we assume that the fitness threshold $\breve{f}^*$ and the weights w, $\breve{w}$, $\breve{\breve{w}}$ have been fixed before the training procedure was initiated and remain unchanged throughout.[15] Unless the EP's fitness value falls under the required threshold, adjustments are made to $\mathcal{P}$ and $\mathcal{I}$. Generally speaking, the cutoff-type parameters found in $\mathcal{P}$ and the number/location of spline knots in each tabulated function are only changed manually by the developer during training, while the interpolative objectives and any parameters of analytical forms in $\mathcal{P}$ are usually adjusted according to an automated optimization scheme. Finally, the revised interpolative training set is used to regenerate the tabulated functions of the EP and the parameters $\mathcal{P}$ are substituted into any analytical forms. The procedure repeats until a set of parameters $\mathcal{P}^*$ and an interpolative training set $\mathcal{I}^*$ are found for which the EP satisfies the fitness criterion.

The largest discrepancy between the training process of a parametric EP shown in Figure 5.2 and that of tabulated EPs illustrated in Figure 5.3 is the presence of the final interpolative training set in the latter case, which is retained after the training process has completed. Beginning with a primary training set of size $N_t = N_{\mathrm{CMP}} + N_{\mathcal{C}}$ and an initial interpolative training set of size $N_{\mathcal{I}_0}$, the effective output of the fitting process is a set of $N_{\mathcal{P}} + N_{\mathcal{I}^*}$ parameters which define the final tabulated EP which is to be used in application. While the size $N_{\mathcal{P}}$ of the parameter set $\mathcal{P}$ remains unchanged, the size $N_{\mathcal{I}^*}$ of the final interpolative training set $\mathcal{I}^*$ will generally differ from the size $N_{\mathcal{I}_0}$ of the initial interpolative training set $\mathcal{I}$ because the interpolative cardinalities contained in $\mathcal{P}$ can be updated during training. Considering the union $\mathscr{P}_0 \triangleq \{\mathcal{P}_0, \mathcal{I}_0\}$ as a single set of parameters which are input to the training process and $\mathscr{P}^* \triangleq \{\mathcal{P}^*, \mathcal{I}^*\}$ as a single set of parameters which define the final tabulated EP which results from it allows for a general comparison to be made between the training processes of parametric and tabulated EPs. While the training process of parametric EPs only ever contains a static number of parameters which are subject to fitting, the number of parameters present in a tabulated EP will generally change in the course of training. This observation serves to define the latter as one instantiation of the more abstract concept of a *semiparametric empirical potential*: an empirical potential for which the cardinality $N_{\mathscr{P}}$ of its parameter set scales with the cardinality N_t of its primary training set. One might reasonably assume that if additional instances were adding to the primary training set (increasing N_t), there might be a corresponding increase in the number

[15] Although it adds to the already high-dimensional parameter space associated with fitting an EP, one can also allow the weights of the fitness function to be adjusted during fitting. This technique allows one to effectively remove a canonical material property or atomic configuration from the training set by decaying its weight to zero, or add one to the training set by increasing its weight from zero to a finite real number. The reader may consult the recent work of Zhang and Trinkle [57] for more on this strategy.

of parameters $N_{\mathscr{P}}$ of the potential. However, we impose no specific requirement in our definition as to how $N_{\mathscr{P}}$ changes with N_t, i.e. whether it increases or decreases, the rate at which it changes relative to the change in N_t, or any regularity of its change.

Tabulated EPs have become ubiquitous in molecular dynamics calculations, and fitting them has been greatly expedited by packages such as POTFIT which allow the user to combine multiple optimization techniques such as Powell's method [173, 174], simulated annealing [175, 176], and differential evolution [177, 178]. However, they must still be designed with caution because of the flexibility permitted in the interpolative processes used to generate their tabulated subforms. An example of how this can lead to a sacrifice of physical interpretation is provided by the spline EP of Lenosky *et al.* [179], which will be used in the case study of Chapter 6 where we refer to it as the LSA potential. Based on the MEAM formalism, the overarching form of the LSA potential is given by[16]

$$\mathcal{V} = \sum_{\substack{\alpha,\beta \\ \beta>\alpha}} \mathcal{V}_2(r_{\alpha\beta}) + \sum_{\alpha} U\left[\sum_{\beta\neq\alpha} \varrho(r_{\alpha\beta}) + \sum_{\substack{\beta\neq\alpha \\ \beta<\gamma}} f_c(r_{\alpha\beta}) f_c(r_{\alpha\gamma}) g(\cos(\theta_{\beta\gamma}))\right] \tag{5.6}$$

where all five functions $\mathcal{V}_2$, U, ϱ, f_c, and g are determined by cubic interpolations (see Section C.3 for plots of these functions). Despite adhering to the MEAM parent form and providing accurate predictions of many material properties, not all of the tabulated subforms of the LSA potential have physical meaning. In particular, the authors of the potential note that the effective density function given by the argument of U, $\varrho^{\mathrm{CF}} = \sum_{\beta} \varrho(r_{\alpha\beta}) + \sum_{\beta,\gamma} f(r_{\alpha\beta}) f(r_{\alpha\gamma}) g(\cos(\theta_{\beta\gamma}))$, cannot be interpreted as a physical density for two reasons. First, the value of the effective density is negative for an atom in the equilibrium geometry of the diamond ground state predicted by the potential. Second, the two-body contribution ϱ to the effective density is not monotonic, but instead possesses a local minimum at the nearest-neighbor distance in diamond.

Aside from the inference of functional forms which are physically suspect, the interpolation procedure can also give rise to numerical issues. In [180], the authors show that the choice of spline interpolants used in a tabulated potential, i.e. the order of the polynomials used and the requirements on their continuity at the subinterval boundaries, have a pronounced effect on its predictions for applications such as lattice dynamics calculations which make use of its higher-order derivatives, regardless of the density of spline knots used. Moreover, the same effect occurs even if a fully analytical EP is tabulated, as is oftentimes done for

[16] In practice, one also subtracts $\sum_{\alpha} U[\,0\,]$ from (5.6) so that the energy of an isolated atom is taken to be zero.

computational expediency [181]. This result reinforces the importance of the philosophy of tabulated potentials adopted in KIM which was touched upon in Section 3.1.2, wherein the interpolation method itself is considered to be an essential part of the potential.

5.2.2 Neural network potentials

In the last section, we saw that semiparametric EPs differed from parametric EPs in that the cardinality of their parameter set was capable of changing (in some general way left unspecified) as additional instances are added to their primary training set. Tabulated EPs satisfy the definition of a semiparametric EP because of the extensible parameter sets present in the tabulated subforms which underlie their overarching functional form. In this section, we present another possibility of defining semiparametric EPs in the form of *artificial neural networks* [99, 182, 183]. Originating in the 1950s from the work of Rosenblatt [184], neural networks were the first substantial attempt at direct empirical modeling of low-level processes in the brain using an adaptive mechanism. Since then, neural networks have been used extensively in a wide range of scientific fields, and entire journals have even been dedicated to their study [185, 186]. While they went more or less unutilized in the analysis and generation of chemical potential energy surfaces until the early 1990s, there has since been a swift acceleration of interest in this particular domain.[17]

We confine our attention here to the simplest type of neural networks, known as *feedforward neural networks*. A feedforward neural network is defined as a collection of nodes known as *neurons* or *units* which are assembled into M layers and connected to one another in such a way that information flows unidirectionally. The first layer of the network is referred to as the *input layer*, followed in sequence by the *hidden layers* which finally lead to the *output layer*. In keeping with biological terminology, the neurons are said to be connected by *synapses*,[18] each of which is associated with a *synaptic weight*. Indexing each layer by j, where $j = 0$ corresponds to the input layer and $j = (M - 1)$ corresponds to the output layer, we denote the weight associated with the synapse connecting neuron k of layer $(j - 1)$ to neuron i of layer j by $\omega_{k,i}^{j-1,j}$, and the collection of all of the synaptic weights of a neural network by Ω. Figure 5.4 depicts a feedforward neural network which possesses two neurons in its input layer, two hidden layers containing three neurons and two neurons, respectively, and an output layer containing a single neuron. Each neuron is

[17] An appreciation of the increasing popularity of neural networks in chemistry can be gained from Figure 1 of [187], where the author has compiled an estimate of the annual number of publications in chemistry involving neural networks over the past three decades.

[18] The human brain is believed to contain on the order of 10^{11} neurons, each of which is connected via synapses to approximately 10^4 other neurons [98].

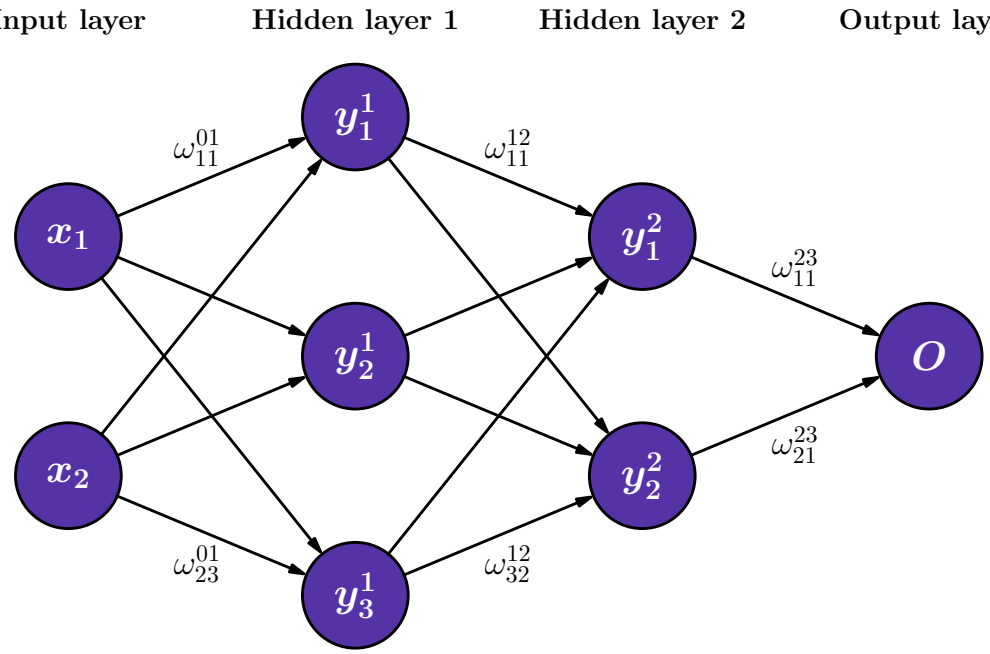


Figure 5.4: An example feedforward neural network with two hidden layers (for a total of $M = 4$ layers). For visual clarity, not all of the weights are labeled, and the bias neurons are omitted.

labeled by its output value, denoted y_i^j for neuron i in layer j, which is determined by two factors. First, the synaptic weights play the role of determining the input to an individual neuron in the form of a weighted sum of the outputs of the previous layer, with the addition of a bias constant b_i^j which allows the resultant sum to be shifted arbitrarily. For example, the input to the neuron labeled y_1^2 in Figure 5.4 is $b_1^2 + \omega_{11}^{12} y_1^1 + \omega_{21}^{12} y_2^1 + \omega_{31}^{12} y_3^1$. Second, an *activation function* ψ is applied within each neuron (excluding those in the input layer), so that the output of neuron i in the first hidden layer is given by

$$y_i^1 = \psi\left(b_i^1 + \sum_{k=1}^{N_0} \omega_{k,i}^{0,1} x_k\right), \tag{5.7}$$

while in general for a node in layer $j > 1$,

$$y_i^j = \psi\left(b_i^j + \sum_{k=1}^{N_{j-1}} \omega_{k,i}^{j-1,j} y_k^{j-1}\right), \tag{5.8}$$

where we have denoted the number of neurons in layer j by N_j. Thus, the output O of the

neural network in Figure 5.4 is

$$O = \psi\left(b_1^3 + \sum_{k=1}^{2} \omega_{k1}^{23} \psi\left(b_k^2 + \sum_{j=1}^{3} \omega_{jk}^{12} \psi\left(b_j^1 + \sum_{i=1}^{2} \omega_{ij}^{01}\right)\right)\right), \tag{5.9}$$

where we have tacitly assumed that the same activation function ψ is used in all neurons. Common choices for ψ include linear, sigmoid, hyperbolic tangent, and Gaussian functions.

As general learning devices, neural networks have been exploited as a means to optimize the parameters of parametric EPs such as the Tersoff potential [188, 189]. However, they have also been employed to construct potential energy surfaces directly, which we term *neural network potentials* (NNPs). In the field of chemistry, many practitioners have used neural networks to study the properties of individual molecules or reactions between a small collection of molecules [190]. In these cases, the systems under investigation possess a well-defined number of degrees of freedom, e.g. the $(3N - 6)$ variables describing the positions of the constituent atoms up to rotations and translations. Thus, the input layer to these NNPs is taken to contain all of the global molecular degrees of freedom, and the output of the network corresponds to the total energy of the system. The Coulomb matrix mentioned in footnote 2 of Chapter 4 is one descriptor which is well-suited to these scenarios, having been used in many of the NNP publications of von Lilienfeld and coworkers (see [191–194] and references within). However, when studying problems in condensed matter physics, where the total number of atoms present in the system differs between applications and individual atomic environments contain various numbers of neighboring atoms due to the imposition of a spatial cutoff, the question arises as to how to proceed when defining the inputs to a NNP, i.e. the descriptor of atomic environments which is fed to the input layer of the neural network. The property of dimensional invariance of a descriptor mentioned in Chapter 4 now comes into play. As we will see shortly, the architecture of a NNP is usually adjusted as part of its overall fitting process. However, the synaptic weights of the network are always optimized for a fixed network architecture, and attempting to alter the architecture of a NNP during use, either in terms of the input layer or the hidden layers, necessitates retraining of the weights. Thus, for the NNPs relevant to the present work, the neurons of the input layer correspond to the different components of a dimensionally invariant descriptor of atomic environments, while the output of the single neuron in the final layer corresponds to the potential energy ε_α which is assigned to atom α, and the total energy is calculated according to $\mathcal{V} = \sum_\alpha \varepsilon_\alpha$. Because for a fixed architecture, NNPs have a well-defined functional form similar to (5.9), the total force on each

atom in a configuration can similarly be calculated by taking analytical derivatives using the chain rule and summing over all of the remaining atoms of the configuration.[19] Behler *et al.* [196] have proposed a number of NNPs which might be applied to condensed matter problems by making use of the Behler–Parrinello descriptor outlined in Chapter 4. The Spectral Neighbor Analysis Potential (SNAP) of Thompson *et al.* [197, 198] constitutes another condensed matter NNP, albeit in a somewhat trivial sense because it corresponds to a neural network containing no hidden layers,[20] producing atomic energies which are simply weighted linear combinations of the components of the four-dimensional bispectrum. For additional introduction to NNPs, we refer the reader to a general background of the NNP concept given by Cartwright [199], the book of Raff *et al.* [190], and a collection of articles which review the various NNPs which may be found in literature [187, 195, 200–202].

In Figure 5.5, we see the general training process of a condensed matter NNP. Rather than including canonical material properties in the primary training set, only point observations are used ($N_t = N_{\mathcal{C}}$). The primary training set shown in block A is broken up into two complementary segments: the *secondary training set* (block B) and the *validation set* (block C), whose significance will be clarified in the explanation of the training process given below. In this case, the set $\mathcal{P}$ contains the parameters of the descriptor used as input to the NNP, a specification of the network architecture including the number of layers and the number of neurons in each layer, and the form and parameters of the activation function of the neurons. Training begins by using the initial parameters $\mathcal{P}_0$ in block D to construct a neural network and fit its weights to the secondary training set. First, the descriptors corresponding to each atomic configuration present in the secondary training set are computed. Next, the initial network architecture and activation function ψ specified in $\mathcal{P}_0$ are used to build a neural network with an initial set of synaptic weights Ω_0. An interpolative training set $\mathcal{I}$ is formed using the descriptors from the secondary training set and the corresponding point objectives, and the network training process is executed (block E). The resulting weights are passed to block F, where the NNP is formally specified with all of its descriptor parameters and its initial neural network. Next, the descriptors corresponding to the configurations in the validation set are computed and fed to the NNP to render predictions (block G) which are subsequently compared to the point objectives of the validation set in block H. Presuming the fitness criterion $\breve{f} < \breve{f}^*$ is not satisfied, adjustments are made to the descriptor parameters and/or the network architecture and activation function

[19] The force on a given atom in a configuration will generally be influenced by all of the other atoms of the configuration which fall within twice the descriptor cutoff distance of it. See [195] for details.

[20] A neural network which consists of only an input layer and a single-neuron output layer is commonly known as a *perceptron*, whereas those with hidden layers are often referred to as *multilayer perceptrons*.

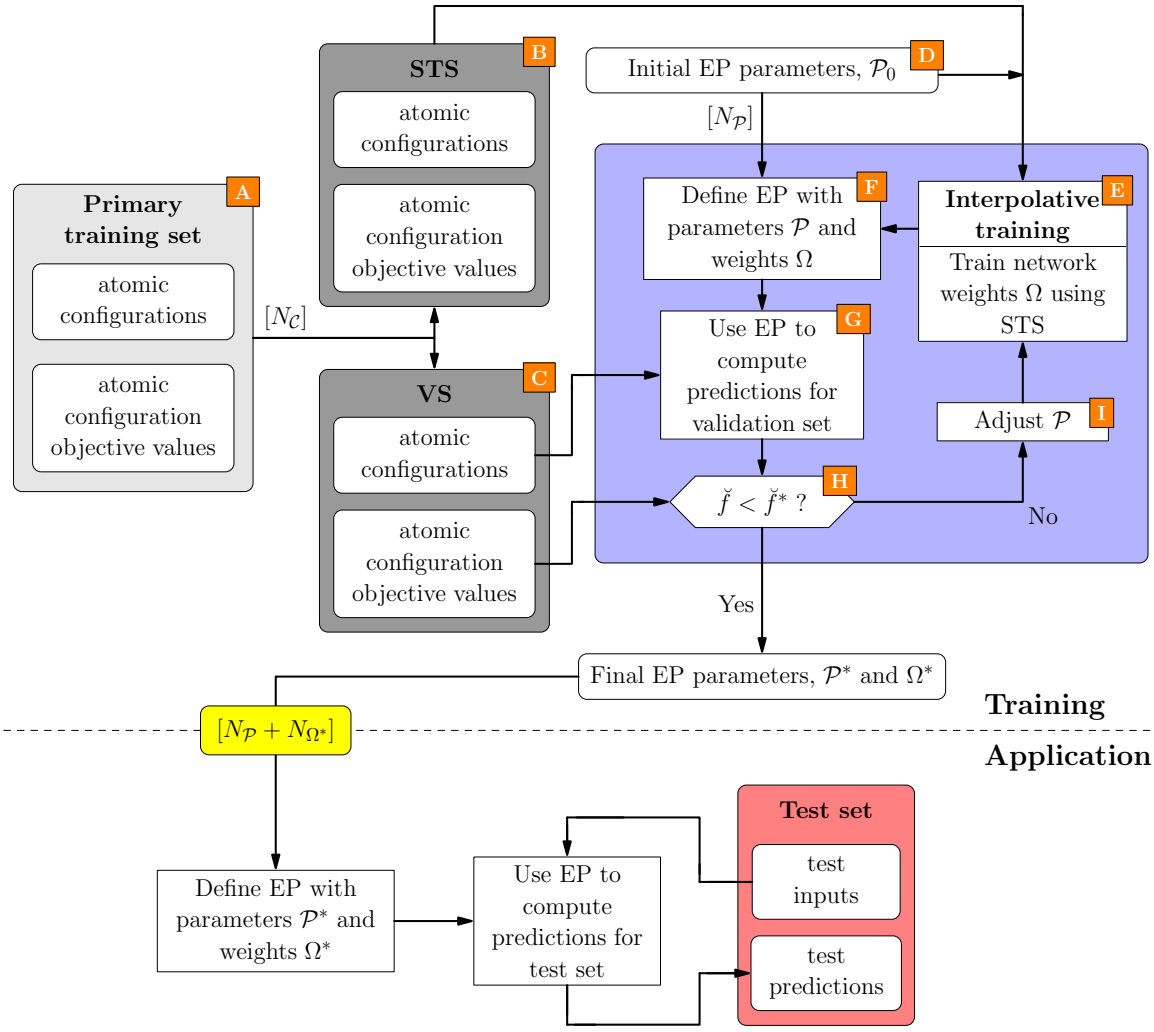


Figure 5.5: Training process of NNPs and how they are used to make predictions. The blocks labeled STS and VS correspond to the secondary training set and validation set, respectively. Both the final parameter set $\mathcal{P}^*$ of size $N_{\mathcal{P}_0}$ and the final set of network weights of size N_{Ω^*} are retained for making predictions in application.

in block I. Finally, the descriptors are recomputed for each configuration in the secondary training set if necessary, and the new network is retrained to yield weights Ω. The process repeats until a collection of parameters $\mathcal{P}^*$ and synaptic weights Ω^* is found for which the performance criterion is met.

From Figure 5.5, we can see that the cumulative parameter set $\mathscr{P}$ of a NNP consists of the fixed-size parameter set $\mathcal{P}$, as well as the synaptic weights Ω. By adding or removing neurons/layers, the cardinality of the weight set can be adjusted freely during training relative to the number of instances N_t in the primary training set, i.e. $N_{\Omega^*} \neq N_{\Omega_0}$ in general, and thus NNPs satisfy the definition of a semiparametric potential. However, there are several

prominent differences between the methods by which TEPs and NNPs are trained. First, in contrast to TEPs, where the interpolative objectives are essentially model parameters which are adjusted during training, the interpolative objectives used in NNP training come directly from the secondary training set and remain fixed both in number and value. Additionally, whereas in typical TEPs the spline coefficients defining the interpolated functions are obtained by solving a basic linear system of equations, the interpolative training process of NNPs shown in block E of Figure 5.5 in which the network weights are fitted is non-trivial, requiring an iterative solution procedure which attempts to reproduce the total energies and atomic forces which constitute the interpolative objectives. The optimization methods employed for this task can be performed in two different ways: *batch learning* or *online learning* [183, p. 161]. In the former, the entire interpolative training set is used simultaneously to adjust the weights, while in online learning the interpolative training instances are fed to the network in serial, with weight adjustments being made as each instance is encountered. While online learning methods of training the network weights are commonly used, the precise order in which the training instances are encountered by the network can be significant, and presents another complication to the overall fitting process of a NNP.[21] It may further be noted that while the interpolative training set is necessarily retained post-training in TEPs in order to fully define them, the interpolative training set of a NNP is discarded once training is complete. A final distinction from TEP training is the presence of a descriptor which has its own parameters in addition to a spatial cutoff, and is applied to the atomic configurations of the secondary training set in order to define the interpolative inputs used to train the network weights, as well as applied to the validation set to assess performance. As evidenced by Chapter 4, the selection of a descriptor is far from simple, and even identifying a viable set of descriptor parameters can be daunting. Accordingly, the descriptor used in a NNP should be considered as important a part of the potential as the neural network used to assign energies to atomic environments.

Aside from the differences which separate their training processes, a further point of variation between TEPs and NNPs is the hierarchical structure of the interpolation method used in the latter, which presents a unique and interesting possibility for creating and studying energy landscapes. As can be seen from (5.9), each hidden layer amounts to another stage of convolution of the descriptor fed to the network before the output layer is eventually reached. Hence, it is conceivable that with many hidden layers, using simple activation functions (as is often done in NNPs) could nonetheless expose highly convoluted aspects of the BOPES. Furthermore, because the outputs of each layer can be individually exam-

[21] For more information on training NNPs, the reader may consult [203–206], as well as the aforementioned review articles.

ined, it is possible to explicitly follow the incremental processing of a descriptor vector which is input to the network as it propagates toward the output layer. This particular type of analysis, demonstrated in [192], provides a potential bridge to discovering the individual features of atomic environments which are most influential in determining the total energy of an atomic configuration.

The above discussion notwithstanding, if the hierarchical structure of a neural network is foregone and only a single hidden layer is used, there are several informal similarities which may be drawn between NNPs and TEPs. For instance, the specific choice of splines (the degree of the piecewise polynomials and the continuity requirements at subinterval boundaries) used to perform interpolation in a TEP plays a role not unlike that of the activation function in NNPs which possess only a single hidden layer. Moreover, choosing the number of interpolative inputs (knots) used in the spline fits is intuitively no different than adjusting the number of neurons in such a NNP. This equivalence derives from the fact that each of these cases notionally amount to fitting a function to a set of interpolative objectives which is expressed as a linear combination of general basis functions (polynomials in the case of spline-based TEPs and activation functions in the case of NNPs). This gives rise to what we refer to as the *capacity* of a regression model, which describes the space of functions which it can approximate to arbitrary precision. As we have just alluded to, there are two independent aspects which dictate the capacity of a model: (1) the local variation of the basis functions used to express the model,[22] and (2) the number of basis functions which are employed. The capacity of an empirical potential is thus gauged by the size of its effective parameter set $\mathscr{P}$, as well as the nature of its constituent functions which make use of these parameters. Consequently, a semiparametric EP can alternatively be defined as a potential whose capacity can be adjusted arbitrarily to the whim of the developer regardless of the size of the primary training set. By including a sufficient number of spline knots or neurons, the capacity of the regression models underlying TEPs and NNPs can be extended to capture a vast array of functions, and it is because of this characteristic adaptability that they can be applied to a wide diversity of materials.[23]

[22]Here, the term *variation* is used in the mathematical sense. For a real-valued continuous univariate function $f(x)$ defined on an interval $x \in [a, b]$, the variation is defined as

$$V = \sup_{\Gamma} \sum_{i=1}^{m} |f(x_i) - f(x_{i-1})|,$$

where the summation runs over the m elements of a partition $\Gamma = \{x_0, \ldots, x_m\}$ of the interval $[a, b]$ with $x_0 = a$ and $x_m = b$, and the supremum is taken over all such partitions. The variation can hence be seen as a measure of how rapidly a function changes value on an interval. For functions which are not monotonic, it is indicative of its corrugation.

[23]The general approach of extending the capacity of a model by including an increasing number of com-

However, the versatility of semiparametric potentials comes at a price, as elucidated by the celebrated *bias–variance dilemma* of machine learning [211]. In supervised learning, the *bias* of a model reflects the deviation of its predictions from the (unknown) latent function which underlies its training set. Complementing the bias of a model is its *variance*, which is comparable to its capacity and indicates how greatly its predictions fluctuate in response to changes in its training set. The bias–variance dilemma states that, in all models, there is a trade-off which occurs between bias and variance. Models which possess excessive bias restrict their flexibility by confining themselves to specific solution spaces, giving them insufficient variance to accurately model complicated data observations; this corresponds to what is known as *underfitting* of the data. A hypothetical example of underfitting in the context of EPs would be attempting to use a pair potential to model a material with considerable environment-dependent bonding such as carbon or silicon, where the energies of individual bonds are related to the local density of bonds, as well as their relative orientation. On the other hand, EPs which have excessive variance and insufficient bias are likely to *overfit* a set of data observations, disregarding the physics of the latent function which underlies a set of chemical observations in favor of reproducing its training set to high accuracy, including any noise present. An example of overfitting was included at the end of Section 5.2.1, where it was shown that the tabulated subforms of the LSA spline potential were unphysical, assuming heuristic forms which were overtly predisposed to reproducing the properties of the ground state diamond structure.

In application, neural networks with a large number of neurons are likely to overfit their training set just as are tabulated subforms of TEPs which have an overabundance of interpolative inputs. In fact, it has been shown that under mild assumptions on the activation function, even a neural network with only a single hidden layer is capable of approximating any continuous mapping over a finite domain, provided it is allowed a sufficient number of neurons [212]. This leads us, however, to the final distinction to be made between NNPs and TEPs. Whereas TEPs possess an overarching form which acts as an eventual source of physical bias and constrains their complexity, the capacity of NNPs is limited only by the discretion of the developer. Thus, there is a significantly higher risk of overfitting NNPs than there is TEPs because of the complete lack of physical guidance present in the former. Following Behler [195], we refer to such EPs as *mathematical potentials*. Without the influence of physical bias, mathematical potentials are thus left to infer their own rules (in terms of their descriptor space) which explain the chemical data supplied in their training set, rendering them pliant to virtually any data with which they are presented,

ponents is also known in some contexts as the *method of sieves* [207–210].

regardless of its origin. The practical implication of this autonomy is that NNPs typically possess high generalization error, i.e. their ability to perform accurate extrapolation outside the proximity of their training set is severely hindered, and it is for this reason that we maintained the terminology of the "interpolative training set" when discussing NNP training. However, despite numerous specific comments acknowledging this fact in the literature of NNPs (see, for example, [63, 187, 190, 192, 195, 202, 206, 213–220]),[24] there is still, at least in principle, recourse. Including a wealth of configurations in the training set of a NNP which cover a considerable portion of descriptor space which is anticipated to be visited in application offers a way of avoiding the extrapolative regime in favor of the interpolative regime. Configuration sampling of this sort has traditionally been done by extracting random configurations from molecular dynamics simulation using a common EP [228], although methods which aim to uniformly probe configuration space have also been proposed [229, 230].

5.3 Nonparametric methods

The first two sections of this chapter were dedicated to the delineation of parametric and semiparametric potentials. The defining characteristic of parametric potentials is that their underlying regression model uses a well-defined set of parameters which remains fixed in size irrespective of the training set. Semiparametric potentials were shown to diverge from the framework of parametric EPs by having a parameter set which is allowed to change in a general fashion as instances are added to or removed from the training set. While many researchers in the materials community are conversant with these types of potentials and, consequently, the vast majority of EPs found in literature can be associated with one of these two paradigms, it is the intent of the current section to reveal the final category of potentials which might exist: *nonparametric empirical potentials*.

Nonparametric regression methods are defined by the quality that their effective parameter set grows in size proportionately to their training set. As such, the instances of the training set used for nonparametric regression can themselves be construed as parameters which play a crucial part in defining the final fit to the data, and must accordingly be retained in order to make predictions for inputs not included in the training set. The method of nonparametric regression was, in fact, tacitly encountered earlier in this chapter in the form of spline regression. Because the piecewise polynomials which define a spline are writ-

[24] Remarks on the lack of transferability of neural networks as general regression mechanisms can be found in [183, 221]. For a broad background of the principle that machine learning models which incorporate only modest bias are incapable of extrapolating with consistent accuracy, we refer the reader to the work of Wolpert [222–227].

ten directly in terms of the values of the interpolative inputs and objectives used to train them, spline interpolation satisfies the definition of a nonparametric regression technique. It is well-established in the machine learning literature that, much like neural networks, nonparametric methods of regression such as splines impose only a minor amount of bias when determining their optimal fit to a training set, and therefore have an extremely limited capability for extrapolation. Note, however, that this does not imply that TEPs are mathematical potentials by simple virtue of the fact that they make use of spline interpolation—as previously mentioned, the physically motivated overarching functional form of a TEP subsumes its potentially unphysical subforms, inevitably reducing its overall variance. Conversely, neural networks of fixed architecture are parametric regression methods since the number of parameters specifying their activation function and synaptic weights is independent of the size of their interpolative training set, yet NNPs are mathematical potentials nonetheless. A third combination of these characteristics is found in nonparametric EPs, which are mathematical potentials which make evident use of nonparametric regression algorithms. Like NNPs, nonparametric potentials are mathematical because they lack an enveloping physical form, and similarly make use of dimensionally invariant additive descriptors which possess their own unique parameter set in order to produce permutationally invariant atomic energies ε_α. However, they are set apart from NNPs by the fact that they must, by definition, grow in capacity as instances are added to their training set.

In light of the statements above, the abstract relation between parametric, semiparametric, and nonparametric EPs can be summarized as follows: parametric EPs compress their training set to a finite parameter vector regardless of its size, the parameter vector associated with a nonparametric EP grows asymptotically in size as its training set grows, and semiparametric EPs are an intermediate of the two which may coincide with either limit depending on the desires of the developer. We reemphasize that the notion of a mathematical potential is formally independent of this categorization, as evidenced by the fact that NNPs and TEPs are both semiparametric potentials, yet one is mathematical and the other is not. However, despite the varying nature of semiparametric EPs, it may safely be assumed in most cases that nonparametric EPs are mathematical potentials, while parametric EPs are based on physical insight. Finally, one should be aware that, in application, where the training process has ceased and the parameter set of an EP is fixed, all potentials are effectively parametric no matter their origin. In the remaining section of this chapter, we consider a specific class of nonparametric techniques and, for concreteness, we defer our explanation of the training of nonparametric potentials to this specific case.

5.3.1 Reproducing kernel Hilbert space potentials

By far the most popular nonparametric regression methods used to construct nonparametric EPs are derived from the theory of *reproducing kernel Hilbert spaces* (RKHS) [231–233]. The mathematical foundation of RKHS regression methods begins in ordinary least-squares (linear) regression. Suppose we are given a training set consisting of sample inputs $\{\mathbf{x}^t\}_{t=1}^{N_t}$ and corresponding sample outputs $\{y^t\}_{t=1}^{N_t}$, where $\mathbf{x}^t \in \mathbb{R}^{D+1}$, $y^t \in \mathbb{R}$ for $t = 1, \ldots, N_t$. We further assume that each observation y^t is subject to noise ν which follows a Gaussian distribution with a mean of zero and variance σ_n^2, and we restrict the first component of each vector $\mathbf{x}^t$ to be equal to 1. For notational convenience, we may arrange the N_t column vectors $\mathbf{x}^t$ into a $(D+1) \times N_t$ matrix $\mathbf{X}$ such that[25]

$$\mathbf{X} \triangleq \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1^1 & x_1^2 & \cdots & x_1^{N_t} \\ x_2^1 & x_2^2 & \cdots & x_2^{N_t} \\ \vdots & \vdots & \ddots & \vdots \\ x_D^1 & x_D^2 & \cdots & x_D^{N_t} \end{bmatrix}, \tag{5.10}$$

and we collect the output values into a column vector $\mathbf{y} \in \mathbb{R}^{N_t}$. As a supervised learning problem, our goal is to find a function $f(\mathbf{x})$ which accurately captures the correlation between the observed sample input/output combinations, and can subsequently be used to generalize to inputs not included in the training set:

$$y = f(\mathbf{x}) + \nu. \tag{5.11}$$

The simplest ansatz that could be made is that the latent function f which underlies the data set is linear,

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{w}, \tag{5.12}$$

where $\mathbf{w} \in \mathbb{R}^{D+1}$ is a set of *weights* which define the precise slope and intercept of f. The premise of least-squares regression is to select, among all functions f which might be used to approximate the training set, that which possesses the lowest sum of squared errors,

$$E_{\text{LSQ}}[f(x)] \triangleq \sum_{t=1}^{N_t} \left(y^t - f(\mathbf{x}^t) - \nu\right)^2. \tag{5.13}$$

[25] Usually the convention assumed for the "design matrix" $\mathbf{X}$ is as the transpose of how we have defined it here. However, since we will follow the notation of Rasmussen and Williams [234] in the rest of this chapter, we adopt the convention found there.

For the present case where $f(x)$ is assumed to be linear, E_{LSQ} evidently takes the form

$$E_{\mathrm{LSQ}}[f(x)] = \sum_{t=1}^{N_t} \left(y^t - (\mathbf{x}^t)^T \mathbf{w} - \nu \right)^2 . \tag{5.14}$$

Taking the derivatives of this equation with respect to the weights $\mathbf{w}$ and equating the results to zero gives a linear system of equations,

$$\mathbf{A}\mathbf{w} = \hat{\mathbf{y}}, \tag{5.15}$$

where $\mathbf{A}$ is a symmetric matrix which takes the form

$$\mathbf{A} \triangleq \begin{bmatrix} N_t & \sum_t x_1^t & \sum_t x_2^t & \cdots & \sum_t x_D^t \\ \sum_t x_1^t & \sum_t (x_1^t)^2 & \sum_t x_1^t x_2^t & \cdots & \sum_t x_1^t x_D^t \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \sum_t x_D^t & \sum_t x_D^t x_1^t & \sum_t x_D^t x_2^t & \cdots & \sum_t (x_D^t)^2 \end{bmatrix}, \tag{5.16}$$

and $\hat{\mathbf{y}}$ is a column vector such that

$$\hat{\mathbf{y}} \triangleq \begin{bmatrix} \sum_t y^t & \sum_t y^t x_1^t & \sum_t y^t x_2^t & \ldots & \sum_t y^t x_D^t \end{bmatrix}^T . \tag{5.17}$$

The set of weights $\mathbf{w}_{\mathrm{LSQ}}$ which is optimal in the least-squares sense is then obtained by solving (5.15) as $\mathbf{w}_{\mathrm{LSQ}} = \mathbf{A}^{-1}\hat{\mathbf{y}}$. Note that $\mathbf{A}$ can also be factored using the input matrix $\mathbf{X}$ as $\mathbf{A} = \mathbf{X}\mathbf{X}^T$, and that it is further true that $\hat{\mathbf{y}} = \mathbf{X}\mathbf{y}$, so that we may finally write

$$\mathbf{w}_{\mathrm{LSQ}} = (\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{y}. \tag{5.18}$$

In order to avoid pathological behavior of the least-squares solution in the case that $\mathbf{X}\mathbf{X}^T$ is ill-conditioned, it is common to regularize the problem by constraining the squared magnitude of the weights, effectively reducing the model variance.[26] The resulting technique, known as *ridge regression* (RR), is implemented by augmenting the least-squares error function using a Lagrange multiplier λ:

$$E_{\mathrm{RR}}[f(x)] \triangleq \sum_{t=1}^{N_t} \left(y^t - f(\mathbf{x}^t) - \nu \right)^2 + \lambda \sum_{i=1}^{D} w_i^2. \tag{5.19}$$

[26]The regularization of the least-squares optimization by penalizing the squared magnitude of the weight vector is alternatively known as *Tikhonov regularization* in statistics literature. This strategy has been used in a vast array of optimization problems, including the training of neural networks, where it is usually called *weight decay*.

It can be shown that taking derivatives with respect to the weights and equating to zero as before yields an optimal solution of the form

$$\mathbf{w}_{\mathrm{RR}} = (\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I}_{D+1})^{-1}\mathbf{X}\mathbf{y}, \tag{5.20}$$

where $\mathbf{I}_{D+1}$ is the $(D+1) \times (D+1)$ identity matrix, and it is from the "ridge" term $\lambda \mathbf{I}_{D+1}$ that RR derives its name. At this point, we may perform some minor algebraic manipulation. Notice that if we examine a somewhat altered version of the prefactor of $\mathbf{y}$ on the right side of (5.20) which has no inverse present, distributivity gives the equality

$$(\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I}_{D+1})\mathbf{X} = \mathbf{X}\mathbf{X}^T\mathbf{X} + \lambda \mathbf{X} = \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I}_{N_t}), \tag{5.21}$$

where $\mathbf{I}_{N_t}$ is the $N_t \times N_t$ identity matrix. Now, multiplying both sides on the left by $(\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I}_{D+1})^{-1}$, and on the right by $(\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I}_{N_t})^{-1}$, we arrive at the expression

$$\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I}_{N_t})^{-1} = (\mathbf{X}\mathbf{X}^T + \lambda \mathbf{I}_{D+1})^{-1}\mathbf{X}, \tag{5.22}$$

which may be substituted on the right-hand side of (5.20) to yield

$$\mathbf{w}_{\mathrm{RR}} = \mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I}_{N_t})^{-1}\mathbf{y}. \tag{5.23}$$

Substituting into (5.12), we see that the final linear function fit to the data from ridge regression yields a prediction for an arbitrary test input $\mathbf{x}_*$ to be

$$f_{\mathrm{RR}}(\mathbf{x}_*) = (\mathbf{x}_*)^T\mathbf{w}_{\mathrm{RR}} = (\mathbf{x}_*)^T\mathbf{X}(\mathbf{X}^T\mathbf{X} + \lambda \mathbf{I})^{-1}\mathbf{y}, \tag{5.24}$$

where we have dropped the subscript on $\mathbf{I}_{N_t}$ for brevity.

We next explore the possibility of extending the linear model considered above to perform nonlinear regression. Notice from (5.24) that the input vectors ($\{\mathbf{x}^t\}_{t=1}^{N_t}$ and $\mathbf{x}_*$) only enter the equation in the matrix $\mathbf{X}^T\mathbf{X}$ and the vector $(\mathbf{x}_*)^T\mathbf{X}$. Furthermore, closer inspection reveals that within these quantities, the input vectors are only present in the form of inner products:

$$\mathbf{X}^T\mathbf{X} = \begin{bmatrix} \mathbf{x}^1 \cdot \mathbf{x}^1 & \mathbf{x}^1 \cdot \mathbf{x}^2 & \cdots & \mathbf{x}^1 \cdot \mathbf{x}^{N_t} \\ \mathbf{x}^2 \cdot \mathbf{x}^1 & \mathbf{x}^2 \cdot \mathbf{x}^2 & \cdots & \mathbf{x}^2 \cdot \mathbf{x}^{N_t} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}^{N_t} \cdot \mathbf{x}^1 & \mathbf{x}^{N_t} \cdot \mathbf{x}^2 & \cdots & \mathbf{x}^{N_t} \cdot \mathbf{x}^{N_t} \end{bmatrix}, \tag{5.25}$$

$$(\mathbf{x}_*)^T\mathbf{X} = \begin{bmatrix} \mathbf{x}_* \cdot \mathbf{x}^1 & \mathbf{x}_* \cdot \mathbf{x}^2 & \cdots & \mathbf{x}_* \cdot \mathbf{x}^{N_t} \end{bmatrix}. \tag{5.26}$$

Now, suppose that instead of constraining the function f which is fit to the data to be linear as in (5.12), we allow some nonlinear function $\boldsymbol{\phi}(\mathbf{x}) : \mathbb{R}^D \to \mathbb{R}^h$ to transform the original inputs so that

$$f(\mathbf{x}_*) = \boldsymbol{\phi}(\mathbf{x}_*)^T \boldsymbol{w}, \tag{5.27}$$

where $\boldsymbol{w} \in \mathbb{R}^h$ and we have dropped the restriction that the first element of each vector $\mathbf{x}^t$ be equal to one, so that $\mathbf{x}^t \in \mathbb{R}^D$ for all $t = 1, \ldots, N_t$. Our regression model is thus still linear, but now in a lifted space of dimension h, which is often referred to as a *feature space* in machine learning. It can readily be verified that the only consequence of this is to replace the inner products of (5.25) and (5.26) with inner products of the transformed inputs, so that the ridge regression procedure renders a predictive fit which takes the form

$$f(\mathbf{x}_*) = \boldsymbol{\phi}(\mathbf{x}_*)^T \Phi(\mathbf{X})(\Phi(\mathbf{X})^T\Phi(\mathbf{X}) + \lambda\mathbf{I})^{-1}\mathbf{y}, \tag{5.28}$$

where we have defined $\Phi(\mathbf{X})$ to consist of the element-wise application of the function $\boldsymbol{\phi}$ to the elements of $\mathbf{X}$. Let us define a real-valued *kernel* function $k : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$ to correspond to the inner products of the transformed inputs:

$$k(\mathbf{x}, \mathbf{x}') \triangleq \boldsymbol{\phi}(\mathbf{x}) \cdot \boldsymbol{\phi}(\mathbf{x}'), \tag{5.29}$$

whence we write

$$f(\mathbf{x}_*) = K(\mathbf{x}_*, \mathbf{X})(K(\mathbf{X}, \mathbf{X}) + \lambda\mathbf{I})^{-1}\mathbf{y}. \tag{5.30}$$

Here, we have further defined

$$K(\mathbf{X}, \mathbf{X}) \triangleq \begin{bmatrix} k(\mathbf{x}^1, \mathbf{x}^1) & k(\mathbf{x}^1, \mathbf{x}^2) & \cdots & k(\mathbf{x}^1, \mathbf{x}^{N_t}) \\ k(\mathbf{x}^2, \mathbf{x}^1) & k(\mathbf{x}^2, \mathbf{x}^2) & \cdots & k(\mathbf{x}^2, \mathbf{x}^{N_t}) \\ \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{x}^{N_t}, \mathbf{x}^1) & k(\mathbf{x}^{N_t}, \mathbf{x}^2) & \cdots & k(\mathbf{x}^{N_t}, \mathbf{x}^{N_t}) \end{bmatrix} \tag{5.31}$$

and

$$K(\mathbf{x}_*, \mathbf{X}) \triangleq \begin{bmatrix} k(\mathbf{x}_*, \mathbf{x}^1) & k(\mathbf{x}_*, \mathbf{x}^2) & \cdots & k(\mathbf{x}_*, \mathbf{x}^{N_t}) \end{bmatrix}. \tag{5.32}$$

In accordance with the introduction of the kernel function, the predictive equation (5.30) is

known as *kernel ridge regression* (KRR), and can also be written in the form

$$f_{\mathrm{KRR}}(\mathbf{x}_*) = \sum_{t=1}^{N_t} a_t k(\mathbf{x}_*, \mathbf{x}^t), \tag{5.33}$$

where the vector $\mathbf{a} = (K(\mathbf{X}, \mathbf{X}) + \lambda \mathbf{I})^{-1}\mathbf{y}$.

As we have imposed no restrictions on $\boldsymbol{\phi}$, the dimension h of its target space could be arbitrarily large, leaving the inner products in (5.29) prohibitively expensive to compute. However, it is here that we can gain an advantage by exploiting the so-called "kernel trick" found in machine learning. It turns out that, so long as the kernel is chosen to be continuous and satisfy the properties of an inner product:

1. Symmetry: $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x}', \mathbf{x}) \quad \forall\, \mathbf{x}, \mathbf{x}' \in \mathbb{R}^D$,
2. Nonnegativity: $k(\mathbf{x}, \mathbf{x}) \geq 0 \quad \forall\, \mathbf{x} \in \mathbb{R}^D \quad$ and $\quad k(\mathbf{x}, \mathbf{x}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{0}$,
3. Linearity: $k(a\mathbf{x} + b\mathbf{x}', \mathbf{x}'') = ak(\mathbf{x}, \mathbf{x}'') + bk(\mathbf{x}', \mathbf{x}'') \quad \forall\, a, b \in \mathbb{R}$,

it is possible to utilize basis functions $\boldsymbol{\phi}$ in our nonlinear regression which define extremely high-dimensional (or even infinite-dimensional) feature spaces without ever explicitly computing the feature space images $\boldsymbol{\phi}(\mathbf{x}_*)$ or $\boldsymbol{\phi}(\mathbf{x}^t)$ $(t = 1, \ldots, N_t)$. Briefly, kernels which meet the above criteria, called *Mercer* kernels, have been shown to fall in one-to-one correspondence with a special set of Hilbert spaces. More specifically, any given Mercer kernel corresponds to a Hilbert space of functions for which it is the unique *reproducing kernel*, and such Hilbert spaces are referred to as RKHSs. A kernel $k : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$ is said to be the reproducing kernel of a Hilbert space $\mathcal{H}$ of functions defined on $\mathbb{R}^D$ if it is true that

$$f(\mathbf{x}) = \langle f, k(\mathbf{x}, \cdot\,) \rangle_{\mathcal{H}} \qquad \forall f \in \mathcal{H},\ \forall\, \mathbf{x} \in \mathbb{R}^D, \tag{5.34}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ is the inner product in $\mathcal{H}$. Because it is also true that $k(\mathbf{x}, \cdot\,) \in \mathcal{H}$ for any $\mathbf{x} \in \mathbb{R}^D$, a vital implication of (5.34) is that

$$k(\mathbf{x}, \mathbf{x}') = \langle k(\mathbf{x}, \cdot\,), k(\mathbf{x}', \cdot\,) \rangle_{\mathcal{H}}\,. \tag{5.35}$$

The significance of this result is that evaluating the kernel $k(\mathbf{x}, \mathbf{x}')$ using the original input space, with no reference to the basis functions $\boldsymbol{\phi}$, is exactly equivalent to taking an inner product between the potentially infinite-dimensional functions found in $\mathcal{H}$. Despite the fact that feature spaces associated with many kernels can be explicitly derived, they are

Kernel	Expression
Linear	$\sum_{d=1}^{D} \sigma_d^2 x_d x'_d$
Polynomial	$(\mathbf{x} \cdot \mathbf{x}' + \sigma_0^2)^p$
Laplacian	$\sigma_f \exp\left(-\frac{\|\|\mathbf{x} - \mathbf{x}'\|\|_1}{\ell}\right)$
Squared exponential	$\sigma_f^2 \exp\left(-\frac{\|\|\mathbf{x} - \mathbf{x}'\|\|_2^2}{2\ell^2}\right)$

Table 5.1: Common kernel functions used in RKHS regression. The notation $||\cdot||_1$ is used to represent the Manhattan norm, while $||\cdot||_2$ represents the Euclidean norm.

not usually constructed based on any specific desired feature mappings ϕ, but are rather formulated directly to satisfy criteria 1–3 above and examined retrospectively. Moreover, in applying RKHS regression methods in practice, kernel functions are usually selected by a developer from a relatively small precompiled list known to the machine learning community to be valid and which have proven useful in application. Several of the most common kernels used to perform regression in RKHS methods are shown in Table 5.1, and we note that recent examples of nonparametric potentials for individual molecules [60, 235] and condensed matter systems [236–238] based on KRR favor the Laplacian and squared exponential kernels.[27] Further explanation of the application of KRR in the creation of empirical potentials may be found in these references, as well as the review given in [240].

Finally, we conclude this chapter with an informal introduction to the regression method which will be used in our study of transferability: *Gaussian process regression* (GPR) [99, 234]. As we will show below, the framework of GPR provides an alternative derivation which yields the predictive equation (5.30) of KRR. However, the fundamental difference between the two is that GPR is a Bayesian method, which has two principal consequences. First, we must specify prior beliefs about the parameters of our regression model. In order to do this, we must first extend the definition of a distribution; in order to characterize GPR, we are specifically interested in a generalization of the Gaussian distribution known as a *Gaussian process* ($\mathcal{GP}$). A $\mathcal{GP}$ can be thought of as the infinite-dimensional analog of a multivariate Gaussian distribution. Whereas a Gaussian distribution is associated with a finite-dimensional space, e.g. $\mathbb{R}^s$, a $\mathcal{GP}$ can be thought of as a distribution within a function

[27] The recent article by Vu *et al.* [239] explores the effects of parameter selection in the case of the squared exponential kernel.

space. Accordingly, whereas a Gaussian distribution is specified by a mean vector $\mu \in \mathbb{R}^s$ and a covariance matrix $\Sigma \in \mathbb{R}^{s \times s}$, a $\mathcal{GP}$ over the space of functions which map $\mathbb{R}^D \to \mathbb{R}$ is defined by a mean function $m(\mathbf{x}) : \mathbb{R}^D \to \mathbb{R}$ and a covariance function $k(\mathbf{x}, \mathbf{x}') : \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$. Similar to the notation of Gaussian distributions, we specify a Gaussian process by writing $\mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}'))$. In most cases, the mean function of a $\mathcal{GP}$ prior is simply taken as the constant function $\mathbf{0}$, as no prior knowledge of the value of the latent function underlying the data is available. However, the choice of covariance function is essential, as it serves to define the functions which fall under the support of the $\mathcal{GP}$. In practice, the covariance function is taken to be a kernel function exactly like those discussed in the context of KRR which, by virtue of their definition, will always yield valid (i.e. positive semidefinite) covariance matrices $K(\mathbf{X}, \mathbf{X})$.

The second consequence of the Bayesian nature of GPR is that its prediction for a given test input $\mathbf{x}_* \in \mathbb{R}^D$ is an entire "posterior" distribution rather than a single value. Substituting the prior $\mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}'))$ into the standard form of Bayes' rule (only treating functions as the parameters over which inference is performed) yields the following posterior distribution for a specific test input:

$$\bar{f}(\mathbf{x}_*) = K(\mathbf{x}_*, \mathbf{X}) \left[K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathbf{y}, \tag{5.36}$$

$$\text{var}(f(\mathbf{x}_*)) = k(\mathbf{x}_*, \mathbf{x}_*) - K(\mathbf{x}_*, \mathbf{X}) \left[K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} K(\mathbf{X}, \mathbf{x}_*), \tag{5.37}$$

where we see that the first equation is identical to the predictive equation (5.33) of KRR, only with the variance σ_n^2 of the noise assumed to be present in the sample outputs playing the role of the Lagrange multiplier λ in regularizing the solution. While (5.36) and (5.37) only serve to define a univariate Gaussian distribution, the characterization of a $\mathcal{GP}$ as a distribution over functions can be grasped by considering such a distribution placed at each and every input $\mathbf{x}_* \in \mathbb{R}^D$, where the mean and variance of the distribution at $\mathbf{x}_*$ is determined by its location with respect to the training set observations according to the right-hand sides of these equations. Finally, note that we may also write equations (5.36) and (5.37) in vector form for an arbitrary set of N_{t_*} test inputs arranged into a matrix $\mathbf{X}_*$ as

$$\bar{\mathbf{f}}(\mathbf{X}_*) = K(\mathbf{X}_*, \mathbf{X}) \left[K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathbf{y}, \tag{5.38}$$

$$\text{cov}(\mathbf{f}(\mathbf{X}_*)) = K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}_*, \mathbf{X}) \left[K(\mathbf{X}, \mathbf{X}) + \sigma_n^2 \mathbf{I}\right]^{-1} K(\mathbf{X}, \mathbf{X}_*), \tag{5.39}$$

and $K(\mathbf{X}_*, \mathbf{X}) = (K(\mathbf{X}, \mathbf{X}_*))^T$.

An example of GPR is shown in Figure 5.6, where the input space dimensionality $D = 1$. A

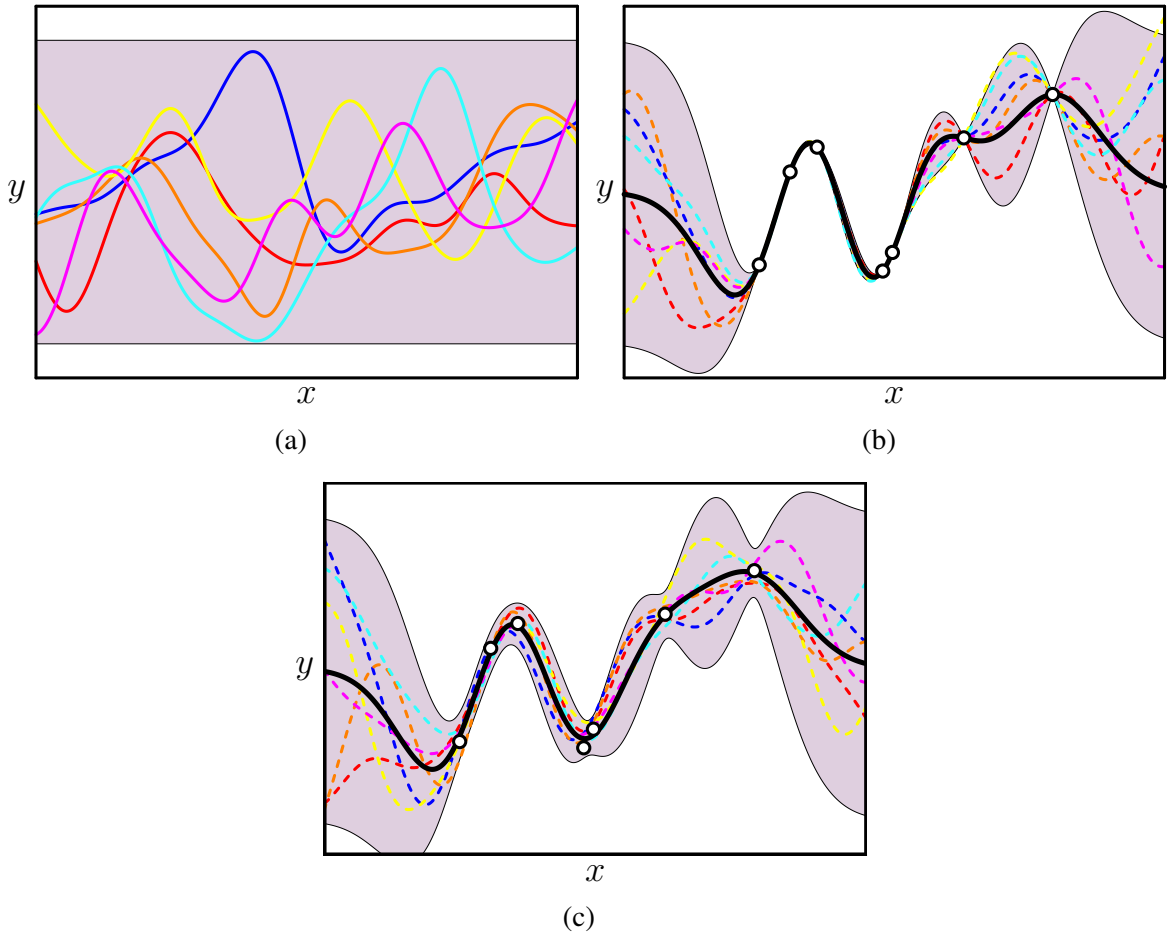


Figure 5.6: Example of a one-dimensional Gaussian process using a zero mean function and a covariance function given by a squared exponential kernel with $\sigma_f = 2.45$, $\ell = 1.2$. (a) The prior variance along with six arbitrary samples from the prior function distribution. (b) The posterior distribution is given for a set of seven training data points when a miniscule noise ($\sigma_n = 0.01$) is used. The mean function of the posterior (heavy black) is shown, along with six functions sampled from the posterior. (c) Posterior distribution along with six sampled functions for a larger noise value of $\sigma_n = 0.4$.

mean function of 0 and a squared exponential covariance function with $\sigma_f = 2.45$, $\ell = 1.2$ define the $\mathcal{GP}$ prior, which is shown in Figure 5.6a along with six functions sampled therefrom. The shaded region indicates the 95% confidence interval at each input x, which is centered about a value of 0 as specified by the prior mean function; the prior variance itself can be obtained by evaluating $k(\mathbf{x}, \mathbf{x}) = \sigma_f^2$. In Figure 5.6b, the prior has been combined with a training set of $N_t = 7$ observations using Bayes' rule with a small noise value of $\sigma_n = 0.01$ to give a posterior distribution over functions. Note that the variance and, consequently, the width of the 95% confidence intervals at each input $\mathbf{x}$ indicated by the shaded region, is reduced near the training set. This behavior is due to the form of (5.37), which can be understood as two terms, the first of which represents the prior variance and the second of which accounts for the supposition that the latent function underlying the data is continuous,[28] and thus that it should share roughly the same value as the training observations in their immediate vicinity. Because a minute noise value of $\sigma_n = 0.01$ has been used in the posterior distribution shown in Figure 5.6b, all functions from the posterior $\mathcal{GP}$ are required to pass almost exactly through the training observations. The precise rate at which the variance grows as one moves away from the training observations is dictated by the form of the covariance function and its parameters; in the present case, where a squared exponential covariance function has been used, the variance grows exponentially quickly away from the training set, with the parameter ℓ acting as a modifier which sets the length scale associated with the functions present in the prior and posterior $\mathcal{GP}$s. The prediction (5.36) made by GPR for an arbitrary test input $\mathbf{x}_*$ corresponds to the mean of the posterior distribution (shown in heavy black), and is thus also determined by the covariance function and its parameters, in addition to the training data. Finally, Figure 5.6c shows a posterior distribution obtained in the same manner, but using a larger noise parameter $\sigma_n = 0.4$. In this case, the functions of the posterior are allowed greater deviation from the training set, which acts to regularize the solution and render a smoother predictive mean.

In applying the GPR technique to create a nonparametric EP, the input vector $\mathbf{x} \in \mathbb{R}^D$ corresponds to a descriptor of atomic environments which, as in NNPs, is assumed to be dimensionally invariant. Also similar to NNPs, the primary training set of nonparametric EPs typically includes the total energies and atomic forces of a collection of atomic configurations $\mathcal{C}$, without the addition of any canonical material properties; for simplicity, we consider in this manuscript training sets containing only total energies $\mathcal{V}_\mathcal{C}$ which play the role of the training observations y^t.[29] Before continuing further, we must pause to acknowl-

[28] Recall that we consider only kernels (and thus, covariance functions) which are continuous.

[29] Additional information on incorporating derivative observations into GPR can be found in [234].

edge a complication which nonparametric EPs share with NNPs based on the statements above which we previously overlooked. Earlier, we formulated an NNP as a neural network which takes as input an environment descriptor $\mathbf{x}$ and outputs an atomic energy ε_α. While the process of training such a neural network to reproduce the net force acting on atom α is relatively straightforward, a detail that went unmentioned in our discussion was how the network is trained so as to reproduce the total energies of the configurations in the primary training set. The reason this is non-trivial is that the total energy $\mathcal{V}_\mathcal{C}$ of configuration $\mathcal{C}$ is not a function of only a single descriptor vector, but rather the collection of the descriptor vectors corresponding to each atom it contains, as dictated by the foundational assumption that the total energy is written as a superposition of atomic energies:

$$\mathcal{V}_\mathcal{C} = \sum_{\alpha=1}^{\mathcal{A}_\mathcal{C}} \varepsilon_\alpha, \tag{5.40}$$

where $\mathcal{A}_\mathcal{C}$ is the total number of atoms in configuration $\mathcal{C}$. While a further explanation of how NNP training accounts for this discrepancy was omitted for brevity, we will address it here in the context of GPR since the latter will be used in the remainder of this work.

In the predictive equations (5.38) and (5.39) of GPR, each training observation y^t is associated with a single input vector $\mathbf{x}^t$. To the contrary, in using GPR as a regression method for a nonparametric potential, each total energy observation $\mathcal{V}_\mathcal{C}$ in the training set is related to multiple descriptor vectors $\mathbf{x}$ which describe the environments of each atom belonging to configuration $\mathcal{C}$. Accordingly, we must augment equations (5.38) and (5.39) by replacing the matrices containing the covariance between individual atomic environments with matrices containing the covariance between entire atomic configurations. Denoting the set of all $N_\mathcal{C}$ atomic configurations in the training set as $\boldsymbol{\mathcal{C}}$ and the set of all $N_{\mathcal{C}_*}$ atomic configurations in the test set as $\boldsymbol{\mathcal{C}}_*$, we write

$$\bar{\mathbf{f}}(\boldsymbol{\mathcal{C}}_*) = \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}) \left[\mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathbf{y}, \tag{5.41}$$

$$\operatorname{cov}(\mathbf{f}(\boldsymbol{\mathcal{C}}_*)) = \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}_*) - \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}) \left[\mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}_*), \tag{5.42}$$

where the noise parameter σ_n now corresponds to the uncertainty in the total energy observations of the primary training set, $\mathbf{I}$ is the $N_\mathcal{C} \times N_\mathcal{C}$ identity matrix, and the configuration covariance matrices have dimensions

$$\mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}) : [N_\mathcal{C} \times N_\mathcal{C}],$$
$$\mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}) : [N_{\mathcal{C}_*} \times N_\mathcal{C}],$$

$$\mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}_*) : [\, N_{\mathcal{C}} \times N_{\mathcal{C}_*} \,],$$
$$\mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}_*) : [\, N_{\mathcal{C}_*} \times N_{\mathcal{C}_*} \,].$$

Using the linearity of the covariance function $k(\mathbf{x}, \mathbf{x}')$ between atomic environments and the superposition of the energy in (5.40), it can be shown that the covariance matrices $\mathcal{K}$ between atomic configurations are related to the covariance matrices K between atomic environments by

$$\begin{aligned} \mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}) &= LK(\mathbf{X}, \mathbf{X})L^T, \\ \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}) &= L_*K(\mathbf{X}_*, \mathbf{X})L^T, \\ \mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}_*) &= LK(\mathbf{X}, \mathbf{X}_*)L_*^T = \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}})^T, \\ \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}_*) &= L_*K(\mathbf{X}_*, \mathbf{X}_*)L_*^T. \end{aligned} \tag{5.43}$$

Denoting the total number of atoms present across all configurations in the training set by $N_{\mathcal{A}}$, the $N_{\mathcal{C}} \times N_{\mathcal{A}}$ matrix L above consists entirely of the values 0 and 1. The ij-th element of L is equal to 1 if the atom corresponding to column j belongs to training configuration i, and equal to zero otherwise.[30] Similarly, if the total number of atoms present across all test configurations is $N_{\mathcal{A}_*}$, the $N_{\mathcal{C}_*} \times N_{\mathcal{A}_*}$ matrix L_* also consists of the values 0 and 1, where the ij-th element is equal to 1 if the atom in the test set corresponding to column j belongs to test configuration i, and zero otherwise.

Bearing the above in mind, Figure 5.7 shows the training procedure of a GPR-based non-parametric EP. As with NNP training, the process begins by dividing the primary training set (block A) into a secondary training set and a complementary validation set (blocks B and C). The parameter set $\mathcal{P}$ now contains the parameters of the atomic descriptor, the form and parameters of the covariance function $k(\mathbf{x}, \mathbf{x}')$, and the noise parameter σ_n. The initial set of these parameters $\mathcal{P}_0$ (block D) is first used to conduct the interpolative training procedure (block E), which amounts to computing the vector $\mathbf{a}_{\mathrm{STS}} = [\mathcal{K}(\boldsymbol{\mathcal{C}}_{\mathrm{STS}}, \boldsymbol{\mathcal{C}}_{\mathrm{STS}}) + \sigma_n^2\mathbf{I}]^{-1}\mathbf{y}_{\mathrm{STS}}$, where $\boldsymbol{\mathcal{C}}_{\mathrm{STS}}$ refers to the set of all configurations in the secondary training set. After formally defining the EP with parameters $\mathcal{P}_0$ and the vector $\mathbf{a}_{\mathrm{STS}}$ in block F, it is used to compute predictions for the configurations present in the validation set (block G). This process simply consists of computing the matrix $\mathcal{K}(\boldsymbol{\mathcal{C}}_{\mathrm{VS}}, \boldsymbol{\mathcal{C}}_{\mathrm{STS}})$ (where $\boldsymbol{\mathcal{C}}_{\mathrm{VS}}$ corresponds to the configurations in the validation set) and multiplying it by the vector $\mathbf{a}_{\mathrm{STS}}$ from the previous step to arrive at a set of total energy predictions. Block H compares these results to the objectives in the validation set. Assuming the fitness crite-

[30] Note that this implies that out of all of the $N_{\mathcal{C}}$ values within each column of L, only a single entry will be equal to 1, while the rest are equal to 0.

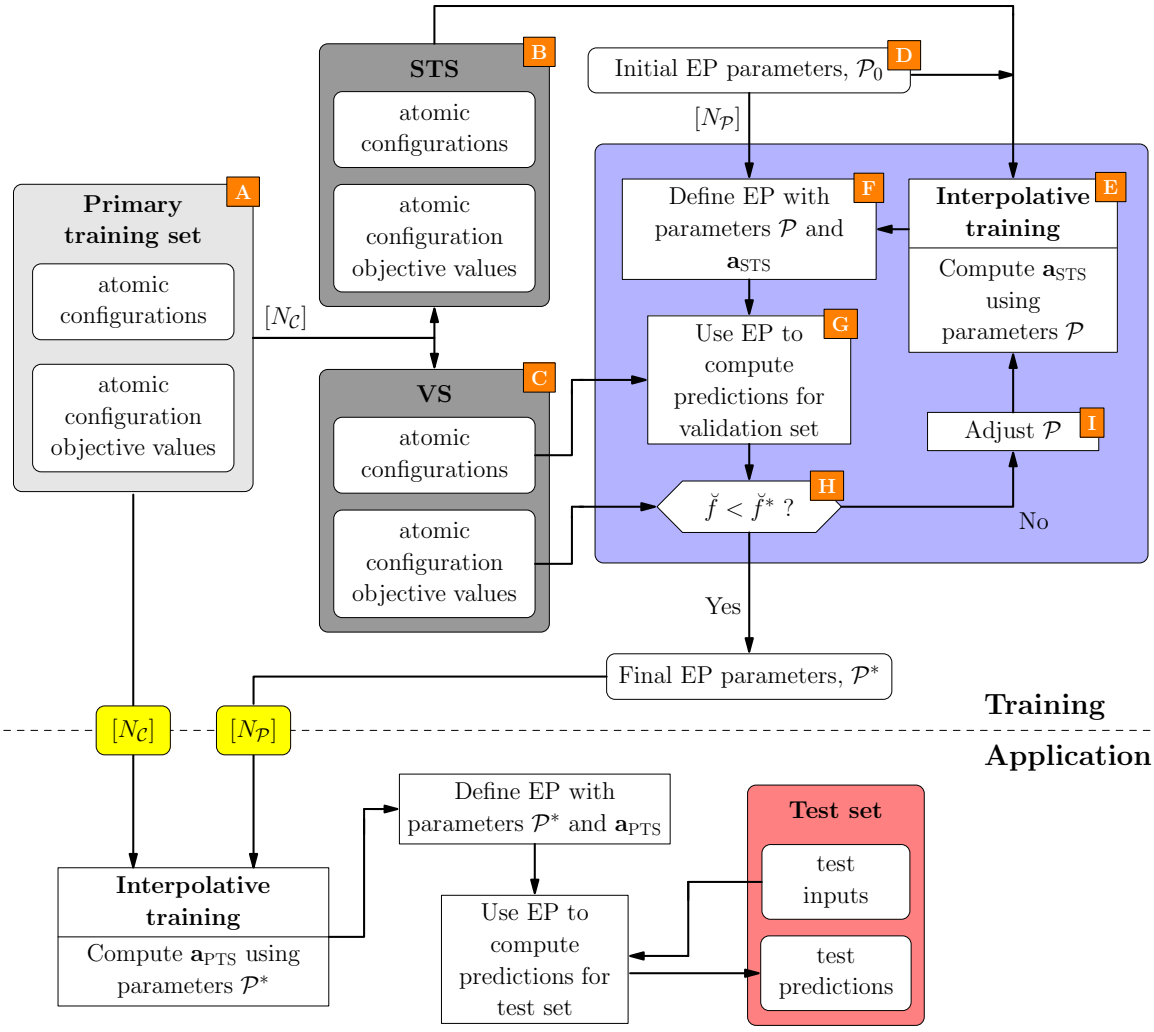


Figure 5.7: Training process of a GPR nonparametric potential and how it is used to make predictions. The blocks labeled STS and VS correspond to the secondary training set and validation set, respectively. Both the final parameter set $\mathcal{P}^*$ of size $N_\mathcal{P}$ and the primary training set of size $N_\mathcal{C}$ are retained for making predictions in application. The vector $\mathbf{a}_{\text{STS}}$ referred to in block E has the value $[\mathcal{K}(\boldsymbol{\mathcal{C}}_{\text{STS}}, \boldsymbol{\mathcal{C}}_{\text{STS}}) + \sigma_n^2\mathbf{I}]^{-1}\mathbf{y}_{\text{STS}}$, while the vector $\mathbf{a}_{\text{PTS}}$ used to make predictions is equal to $[\mathcal{K}(\boldsymbol{\mathcal{C}}_{\text{PTS}}, \boldsymbol{\mathcal{C}}_{\text{PTS}}) + \sigma_n^2\mathbf{I}]^{-1}\mathbf{y}_{\text{PTS}}$.

rion is not satisfied, adjustments are made to the parameters $\mathcal{P}$ (block I) and the process continues until a parameter set $\mathcal{P}^*$ is found for which the fitness function value $\breve{f}$ is sufficiently small. At this point, the primary training set is retained and used to calculate $\mathbf{a}_{\text{PTS}} = [\mathcal{K}(\boldsymbol{\mathcal{C}}_{\text{PTS}}, \boldsymbol{\mathcal{C}}_{\text{PTS}}) + \sigma_n^2\mathbf{I}]^{-1}\mathbf{y}_{\text{PTS}}$, so that predictions for a test set of configurations $\boldsymbol{\mathcal{C}}_{\text{TS}}$ can be made by computing $K(\boldsymbol{\mathcal{C}}_{\text{TS}}, \boldsymbol{\mathcal{C}}_{\text{PTS}})\mathbf{a}_{\text{PTS}}$.

The GPR technique was first applied as an EP in the *Gaussian Approximation Potential* (GAP) of Bartók *et al.* [241–243], which was originally based upon using a squared exponential covariance function with the four-dimensional bispectrum of a Dirac δ-based atomic

neighborhood density. The SOAP kernel discussed in Section 4.3.1 was later introduced by the same authors in [62], where it was shown to demonstrate promising environment reconstruction capability and generally improve the results of GAP, and has therefore become the current standard in distributions of the GAP software package [19] and its applications [96, 97]. However, while GAP has undergone considerable improvements since its inception and has proven itself a useful tool in investigating atomistics, it does still carry its own unique problems. First, the descriptor and covariance function, including their parameters, must be selected with diligence equal to that used in selecting the functional form of a parametric EP. Second, it must be cautioned that it is still a mathematical potential, and thus fundamentally lacks transferability. Some authors [244] have exploited the Bayesian nature of GPR in an attempt to combat this deficiency by using an online learning method in which atomic configurations encountered in simulation which possess a high variance (cf. (5.42)) are subjected to DFT calculations and added to the training set before the simulation is continued. That is, the potential never entirely leaves its training phase.[31] However, this solution still presents a considerable expenditure of resources compared to parametric EPs, which are at least an order of magnitude faster than nonparametric EPs even when no additional DFT calculations are performed. Therefore, we propose in the final chapter of this work a method which leverages the method of GPR but which, instead of directly constructing an EP, seeks to analyze the transferability of existing parametric EPs so that they may be appropriately selected for a specific application. We furthermore propose that, through a combination of manual and automated examination, this device can be used to develop parametric EPs with improved transferability characteristics. For additional background and contemporary work related to NNPs and nonparametric EPs, we urge the reader to consult the recent article of Rupp [245], as well as the collection of articles assembled in [246].

[31] In fact, this type of strategy has been introduced in the context of NNPs by Behler, who proposed the simultaneous calculation of a given configuration by multiple NNPs in parallel, with the degree of variation between the corresponding predictions indicating whether or not they are valid.

Chapter 6

Regression Algorithm for Transferability Estimation (RATE)

6.1 Introduction

In the last several decades, much attention has been paid to EPs owing to their ability to expeditiously compute technologically relevant physical properties and provide new insight into complex material behavior that remains unattainable with first-principles methods. The first class of EPs surveyed in the preceding chapter, parametric empirical potentials, are the most widely used because of their unrivaled computational efficiency, which stems from the fact that they are predicated on the use of fixed functional forms selected according to physical knowledge of a given material. However, while parametric EPs continue to find manifold application in the field of atomistics and have demonstrated surprising transferability in many cases, constructing them is an onerous and time-consuming enterprise. Even if the functional form of a parametric EP has been diligently selected, fitting the associated parameters to a small set of canonical material properties is often exceedingly difficult due to the indeterminate nature of the corresponding optimization problem. In response, developers began to supplement the training sets used to fit parametric EPs with point observations, enabling the use of a wider scope of algorithms which help to automate the fitting process (see Section 5.1.2). Aside from providing a more systematic procedure of parameter fitting, these methods aid EP developers in avoiding hypersensitivity of their models to small changes in their parameters, a problem symptomatic of overfitting. Valuable as they may be in some respects, however, these approaches do not convey a sense of the cumulative transferability of the potentials they fit, i.e. they do not indicate the ac-

curacy of a potential for quantities which it was not fitted to reproduce. This obscures the underlying mechanisms which dominate the behavior of the material(s) in question to any subsequent analysis performed using potentials derived therefrom.

A further attempt at circumventing the difficulty of potential development discussed in the previous chapter is the use of alternate approaches to the problem of selecting functional forms for EPs. These methods manifest themselves as either semiparametric or nonparametric EPs. Tabulated EPs constitute the first departure from the philosophy of parametric potentials, retaining an overarching form while using nonparametric regression to generate tabulated functional subforms. A more pronounced step in this direction was seen in neural network potentials, which have a functional form derived from an extensible set of basis functions which can be expanded and convolved at will to reproduce nearly any desired training set of data. Finally, a similar stance is assumed by nonparametric EPs, which employ a collection of basis functions whose size scales directly with the size of their training set (cf. (5.33)).

In contrast to TEPs, an important commonality of neural network EPs and nonparametric EPs alike is the absence of an overarching physical form which constrains the complexity of the resulting model. This lack of inductive bias (which, in accordance with the bias–variance dilemma, implies a high variance) leads both of these classes of potentials to be best understood as mathematical, rather than physical, in nature. However, despite their lack of physical motivation, mathematical EPs are capable of producing remarkably accurate predictions for many material properties. This is made possible by interpolating between training set observations by means of an additive, dimensionally invariant descriptor of atomic environments which concordantly defines a landscape of atomic energies. Thus, a mathematical potential may be expected to yield accurate predictions for atomic environments which are similar to those contained in its training set, so long as the definition of its descriptor is well-founded. The quandary which accompanies this exceptional versatility, as was previously discussed, is that they are incapable of accurately extrapolating outside of their training set. Therefore, whereas the transferability of any EP (including parametric potentials) will inevitably be limited in practice, mathematical potentials are *nontransferable* by definition and must be used with this knowledge in mind.

While mathematical potentials themselves are not transferable, their operational paradigm may nonetheless be used to effect a measure of the transferability of parametric and tabulated EPs. In this chapter, we exploit the capability of mathematical potentials to infer from

a collection of atomic configurations[1] and their total energies a partition which attributes a partial, atomic energy to each environment present across all of the configurations in a self-consistent manner. However, in addition to assigning an energy to individual atomic environments, we use the framework of a mathematical potential to decompose the total energy error of an EP[2] for each configuration in the training set, creating in the process an atomic energy error function for the potential. The end result, which we term the *Regression Algorithm for Transferability Estimation* (RATE), can be used in the context of KIM (Chapter 3) to select a Model for use in studying a specific set of material properties by yielding a quantitative estimate, including uncertainty bounds, of the accuracy of a set of candidate Models for a collection of representative atomic configurations. In the present work, we apply the RATE method to the eight silicon potentials given in Appendix C for a training set consisting of bulk lattices, surfaces, nanostructures, and clusters. Further, in order to better understand the energy error function rendered for each EP, we implement a novel visualization method for the space of atomic configurations and atomic environments contained in this training set, and we proceed to convey the energy error over this domain.

We begin our agenda by briefly summarizing the first-principles calculations carried out and the EPs which will be considered, as well as delineating the regression technique and descriptor used in RATE and an explicit description of how it is applied. Next, we introduce the multidimensional scaling and isomap algorithms which are used to construct the visualizations of atomic configurations and environments which comprise much of the chapter. Section 6.3 outlines the training set of atomic configurations to which the RATE algorithm is applied and how it was generated. Continuing, we leverage the aforementioned visualization method to illustrate the final inputs used in RATE which represent each of these configurations, and examine the corresponding first-principles total energies and the total energy errors of each EP. The same visualization technique is then used to interrogate the first-principles atomic energy function and the atomic energy error functions for each EP learned from the training set by RATE. Finally, the chapter concludes with a discussion of the interpolative and extrapolative capability of these functions.

[1] Recall from Chapter 4 that by using the term *atomic configuration*, we refer to a complete collection of atoms, irrespective of any notion of locality. Thus, an atomic configuration containing N atoms is said to contain N local atomic environments, which need not be distinct.

[2] In Section 6.2.3, we define the total energy error of an EP to be the arithmetic difference of the total energy predicted by an EP and the total energy predicted by first principles.

6.2 Computational methodology

6.2.1 First-principles calculations and potentials

The first-principles calculations of the total potential energies for each configuration in the training set were performed using the Vienna Ab initio Simulation Package (VASP) [247–250], a plane-wave implementation of DFT. The Generalized Gradient Approximation exchange-correlation functional of Perdew, Burke, and Ernzerhof (GGA-PBE) [251, 252] was followed within the Projector Augmented-Wave (PAW) [253, 254] formalism. The maximum energy of the plane waves used to represent the electronic wave function was approximately 319 eV,[3] while the electronic relaxation itself was performed using the algorithm of Kosugi [255] with a convergence tolerance of 1e-5 eV. Near the Fermi level, Gaussian smearing with a broadening of 0.1 eV was used to define partial band occupancies in order to accelerate convergence of the relaxation. Discrete sampling of the Brillouin zone was done using converged gamma-centered Monkhorst–Pack meshes. As with most plane-wave codes, VASP supports only periodic boundary conditions; in cases where there existed one or more aperiodic directions, a large unit cell length was used along these directions and only the gamma point was retained.

All of the calculations in this chapter pertain to elemental silicon. The potentials studied consist of the eight EPs given in Appendix C, many of which are already implemented as KIM Models. For convenience, they are repeated in Table 6.1 along with their literary references, KIM citations where applicable, and the abbreviations we use in the diagrams to follow. The reader is reminded that, aside from the SW and SWS1 potentials, which are three-body cluster potentials, all of the EPs we consider are three-body cluster functionals (see Section 2.2).

6.2.2 Regression technique and descriptor

If the regression technique used within RATE is to be applicable for an arbitrary EP, it is imperative that it be extremely flexible so as to be able to reproduce the diverse spectrum of physical biases inherent to various EPs. As mentioned above, this leads us to consider the regression methods underlying the mathematical potentials we have encountered thus far: neural networks (NNs) and Gaussian process regression (GPR). In the previous chapter, it was shown that NNs present an appealing prospect for atomistic modeling due to their hierarchical structure, which admits the possibility of capturing convoluted features of the Born–Oppenheimer PES. Deciphering the meaning of these features is also facilitated be-

[3] The 'PREC=High' setting in VASP was used to avoid wrap-around errors in the FFT mesh.

Empirical Potential (EP)	Abbreviation	Citations	Cutoff (Å)
Erhart–Albe	EA	[147, 256, 257]	3.05
Environment-Dependent Interatomic Potential	EDIP	[258–260]	3.12
Lenosky *et al.* MEAM spline	LSA	[179]	4.5
Stillinger–Weber	SW	[43, 261–263]	3.77
Stillinger–Weber silicene 1	SWS1	[44, 145, 263]	3.56
Tersoff 2	T2	[124, 257, 264]	3.2
Tersoff 3	T3	[144, 257, 265]	3.0
Modified Tersoff	TMOD	[266]	3.3

Table 6.1: Empirical potentials whose accuracy will be studied for the training set using RATE.

cause the outputs at each successive layer can be examined explicitly. However, while NNs are usually straightforward to implement, their adaptable structure also gives rise to arbitrary degrees of freedom such as selecting the network architecture and choosing a robust method to optimize the network weights. Further, if an online training procedure is used, as is often the case, the precise order in which the training set entries are input to the network creates an additional source of complication. In practical terms, these particular choices can have a marked impact on the quantitative and qualitative performance of the resulting network and can introduce undesirable sensitivities to the inputs, as well as increase the time required for the training process [191].

Albeit GPR is similar to neural network regression in a limited sense,[4] its intricacies are largely subsumed by its theoretical underpinnings, reducing the number of free parameters to the regularization parameter σ_n and those which define the covariance function (or "kernel"). The modest dimensionality of the explicit parameter spaces typically used in GPR makes it a less direct means of discovering hidden features in data compared to NNs, but also amounts to a simplification of the training process. Furthermore, because GPR is intrinsically Bayesian, incorporation of prior knowledge of the problem at hand into the kernel allows much of the richness that would otherwise be wrought by using a NN to be

[4]It has been shown [267] that for some covariance functions, GPR is equivalent to performing regression using a NN with a single hidden layer which contains an infinite number hidden neurons. See also [268].

regained,[5] and each prediction is accompanied by an estimate of its uncertainty. While Bayesian frameworks have been established for neural networks [269, 270], they are not analytically tractable without additional assumptions, requiring computational investigation which necessitates careful oversight [271]. Finally, we note that while GPR requires the inversion of the (symmetric) covariance matrix, a process whose time complexity is $\mathcal{O}(N^3)$ for N training samples, the cost is still favorable compared to the usual training process carried out for neural networks. In [213], the authors carry out a comparative study of different regression methods to predict molecular atomization energies using the same descriptor space—the Coulomb matrix mentioned in Chapters 4 and 5; not only was it determined in this study that GPR performed at least as well as the neural network considered, but its training process was considerably faster. Moreover, the resulting inverse covariance matrix of GPR need be computed only when training (or retraining) is desired, and can otherwise be stored offline and merely retrieved when predictions are needed. Post-training, making M predictions scales as $\mathcal{O}(NM)$, with the calculation of the variance of each prediction scaling as $\mathcal{O}(N^2M)$.

Having selected GPR to conduct the regression, we are left with choosing a descriptor and covariance function in order to perform calculations. For this purpose, we select the SOAP kernel detailed in Section 4.3.1 with $n = 2$, which we recall corresponds to using the power spectrum of a Gaussian-based atomic neighborhood density as a descriptor and subsequently employing a polynomial kernel function. In addition to its superior performance in the numerical reconstruction experiments of Bartók *et al.* [62], it contains only a small number of parameters, making it the kernel of choice in the most recent GAP potentials. In fact, it was found in the present study that the SOAP kernel gave reasonable performance for a range of these parameters, and in the work to follow, we use a fixed set of kernel parameters for all calculations, which we summarize in Table 6.2 (the significance of these parameters can be found in Appendix B).

6.2.3 Explicit strategy

In this chapter, two separate regression procedures are performed in parallel and their results are shown alongside one another. Before proceeding, we repeat the GPR equations (5.41) and (5.42) below for convenience:

$$\bar{\mathbf{f}}(\boldsymbol{\mathcal{C}}_*) = \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}) \left[\mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathbf{y}, \tag{6.1}$$

$$\text{cov}(\mathbf{f}(\boldsymbol{\mathcal{C}}_*)) = \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}_*) - \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}) \left[\mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}) + \sigma_n^2 \mathbf{I}\right]^{-1} \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}})^T. \tag{6.2}$$

[5] The reader is referred to pg. 167 of [234] and references within for comments on this fascinating topic.

Parameter	Value
r_{cut}	5.0 Å
r_{trans}	1.9 Å
$n_{\max}$	20
$\ell_{\max}$	6
σ_{atom}	0.5 Å
ζ	4
$r_{\text{cut}}^{\text{rad}}$	8.39 Å

Table 6.2: Parameters used in the SOAP kernel within RATE.

Recall that, in the above, the covariance between entire atomic configurations was given in terms of the covariance between individual environments by

$$\begin{aligned} \mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}}) &= LK(\mathbf{X}, \mathbf{X})L^T, \\ \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}) &= L_*K(\mathbf{X}_*, \mathbf{X})L^T, \\ \mathcal{K}(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}_*) &= L_*K(\mathbf{X}_*, \mathbf{X}_*)L_*^T, \end{aligned} \tag{6.3}$$

where the matrices L and L_* consist of the values 0 and 1. The first type of regression performed in this chapter is carried out using the total DFT energy $\mathcal{V}_{\mathcal{C}}^{\text{DFT}}$ of each atomic configuration $\mathcal{C}$ in the training set, i.e. the total DFT energies play the role of the regression observations $\mathbf{y}$ in the GPR equations above. Note that this is essentially the prescription followed by the GAP potentials [97, 241, 242] mentioned in the previous chapter,[6] and thus defines an empirical potential energy surface over the descriptor space being used. Contrarily, in the second mode of regression we use the *total energy errors* of each of the eight EPs listed in Table 6.1 as the training objectives $\mathbf{y}$, which we define for a given EP and configuration $\mathcal{C}$ as

$$\mathcal{V}_{\mathcal{C}}^{\Delta} \triangleq \mathcal{V}_{\mathcal{C}}^{\text{EP}} - \mathcal{V}_{\mathcal{C}}^{\text{DFT}}, \tag{6.4}$$

where $\mathcal{V}_{\mathcal{C}}^{\text{EP}}$ is the total energy of configuration $\mathcal{C}$ as predicted by the EP and $\mathcal{V}_{\mathcal{C}}^{\text{DFT}}$ is the total energy of the configuration computed by DFT. Finally, we note that, in addition to the SOAP parameters of Table 6.2 which fully specify the kernel between individual environments, i.e.

$$\begin{aligned} [\,K(\mathbf{X}, \mathbf{X})\,]_{ij} &\triangleq k_{\text{SOAP}}(x^i, x^j), && i, j = 1, \ldots, N_{\mathcal{A}} \\ [\,K(\mathbf{X}_*, \mathbf{X})\,]_{ij} &\triangleq k_{\text{SOAP}}(x_*^i, x^j), && i = 1, \ldots, N_{\mathcal{A}_*}, \quad j = 1, \ldots, N_{\mathcal{A}} \end{aligned}$$

[6] We use the term "essentially" because, in the GAP publications, atomic forces and stresses are present in the training set in addition to total energies.

$$[\, K(\mathbf{X}_*, \mathbf{X}_*) \,]_{ij} \triangleq k_{\mathrm{SOAP}}(x_*^i, x_*^j), \qquad i, j = 1, \ldots, N_{\mathcal{A}_*}$$

the regularization parameter σ_n was taken to have a value of 0.01 in all calculations.

The latter regression in which the energy error is fit rather than the energy has, in fact, been utilized in some of the aforementioned mathematical potentials. The purpose in each case is to use a "baseline" or "core" potential to compute predictions for atomic configurations which correspond to regions of descriptor space where little or no training data is available. This approach was first undertaken by Szlachta in [96],[7] where a GAP potential for tungsten was posed in the form

$$\begin{aligned} \mathcal{V}^{\mathrm{GAP}} &= \mathcal{V}^{\mathrm{EP}} + (\mathcal{V}^{\mathrm{DFT}} - \mathcal{V}^{\mathrm{EP}}) \\ &= \mathcal{V}^{\mathrm{EP}} - \mathcal{V}^{\Delta}, \end{aligned} \tag{6.5}$$

and the core EP was taken to be a pair functional of the Finnis–Sinclair variety. Unexpectedly, Szlachta's results indicated that using this core potential actually reduced the accuracy of the resulting EP compared to GAP potentials fitted directly to DFT data (without the use of a core potential). However, a more successful application of the core potential method was the SNAP potential for tantalum,[8] where the energy is built upon the Ziegler–Biersack–Littmark (ZBL) pair potential [273] in order to account for Pauli exclusion at small nuclear separations. The many-body components of the potential are then incorporated by means of an additive descriptor which is used as the input space of a least-squares linear regression procedure which features the energy, force, and stress errors of the ZBL potential relative to DFT as its objective values. Finally, the SNAP energy is obtained by subtracting these errors from the original ZBL potential in the manner of (6.5). While the above results seem to imply that the prospect of adding incremental corrections to a core potential is only viable when the latter is of an elementary nature, the full capability of the technique has yet to be exhaustively explored, and we believe it to be an important foreseeable application of the RATE algorithm described in this chapter.

6.2.4 Manifold learning algorithms

The primary purpose of RATE is to provide a numerical indication of the probable transferability (or lack thereof) of a given EP to the computation of a specific material property. However, it is also desirable to make the numerical quantities which underlie RATE percep-

[7] A somewhat related approach was also presented earlier in [272], although in the context of parametric EPs rather than mathematical potentials.

[8] The SNAP potential was mentioned in Section 5.2.2 as a type of NNP.

tible in an intuitive way in order to ensure they are consistent with physical expectations, and so that higher level patterns related to the construction of different EPs and their relative performance can be more easily recognized. For this purpose, we enlist the aid of two algorithms commonly used in exploratory data analysis: *multidimensional scaling* (MDS) and *isometric feature mapping* (isomap) [274].

The aim of MDS (more specifically, "metric MDS") is to generate from a given set of pairwise dissimilarities Ξ_{ij} between N objects an embedding of them in a Euclidean space $\mathbb{R}^p$ so that

$$f(\Xi_{ij}) \approx d_{ij}, \qquad i,j = 1,\ldots,N, \tag{6.6}$$

where f is a specific function chosen according to prior beliefs, and d_{ij} are the Euclidean distances computed between the points in $\mathbb{R}^p$. The final embedding obtained is found by minimizing the *raw stress* S_r, defined by

$$S_r \triangleq \sqrt{\sum_{i<j}(f(\Xi_{ij}) - d_{ij})^2}\ , \tag{6.7}$$

where the optimization itself is performed using the SMaCoF majorization method [275]. In conveying the quality of a MDS embedding, it is also useful to remove scale dependence by normalizing the raw stress S_r by the square root of the sum of squared embedding distances to define the *Kruskal stress* S_K:

$$S_K \triangleq \frac{S_r}{\sqrt{\sum_{i<j} d_{ij}^2}} = \sqrt{\frac{\sum_{i<j}(f(\Xi_{ij}) - d_{ij})^2}{\sum_{i<j} d_{ij}^2}}\ . \tag{6.8}$$

In the simplest case, MDS is used as a means of dimensionality reduction: Given a set of N data objects in $\mathbb{R}^q$, an embedding of the data into the lower dimensional space $\mathbb{R}^p$ $(p < q)$ is sought. In this application of MDS, one takes the dissimilarities Ξ_{ij} to be equal to the Euclidean distances D_{ij} calculated in the original q-dimensional space,

$$\Xi_{ij} = D_{ij}, \qquad i,j = 1,\ldots,N, \tag{6.9}$$

and the function f in (6.6) is taken to be the identity, so that the embedding sought is that which optimizes the likeness of the distances in each space:

$$D_{ij} \approx d_{ij}, \qquad i,j = 1,\ldots,N. \tag{6.10}$$

Note that, while the distances D_{ij} in the original space are known, MDS still requires the

use of initial embedding configurations as the starting point of the iterative stress minimization process. Accordingly, the MDS should be repeated for numerous such initial configurations in order to account for the arbitrariness of these selections. An illustration of this procedure is shown in Figures 6.1 and 6.2, where the MDS routine of the Scikit-learn [276] Python library has been used to perform a distance-preserving mapping from $\mathbb{R}^3$ to $\mathbb{R}^2$ for a collection of 2000 data points based on 40 random initial embedded configurations. We can see that the "S" shape of the data in the original space is maintained in the two-dimensional embedding found to possess the lowest final stress.[9] In the present work, our application of MDS is motivated by a somewhat more obscure scenario. Using the SOAP kernel yields a correlation between any two atomic environments, as well as between any two atomic configurations via (6.3); however, we do not possess any rigorous quantitative knowledge which can be used to represent these environments and configurations as points in particular vector spaces—a perceptual tool which could potentially prove valuable in understanding EPs, as well as RATE. Thus, as we will see, our intention here is to use MDS as a method of procuring such a representation (of the lowest acceptable dimensionality) from only a set of dissimilarities derived using the SOAP kernel. A detailed exposition of MDS may be found in the comprehensive text by Borg and Groenen [277], while for an abbreviated introduction we refer the reader to [278].

Similar to the MDS S-curve example above, the purpose of isomap is to produce from a specific configuration of N data points in $\mathbb{R}^q$ a low-dimensional embedding in $\mathbb{R}^p$ which preserves a set of distances between each pair of data points in the original space. However, in isomap, the pairwise distances used in the original space are not the respective Euclidean distances, but rather the *geodesic distances* within a manifold upon which the data are presumed to reside. The geodesic distance g_{ij} between two points i and j on a manifold is defined as the length of the shortest path connecting the two which lies entirely within the manifold itself. In Figure 6.1, for example, the Euclidean distance between one of the purple data points and one of the green data points is considerably shorter than the length of the geodesic path between the two, which must necessarily traverse the space near the blue and turquoise data points along the S-shaped manifold. As a result, points which are nearby in the (Euclidean) metric of the original space $\mathbb{R}^q$ may lay far apart from one another in the (Euclidean) metric of the embedding space $\mathbb{R}^p$. As isomap is generally applicable to any data set with manifold geometry, the precise form of the manifold underlying the data is not explicitly encoded as an input to the algorithm. Instead, the geodesic distances are derived empirically using a locality rule to associate a set of neighbors with each data

[9] This example was excerpted from the Scikit-learn documentation content found at **http://scikit-learn.org/stable/auto_examples/manifold/plot_compare_methods.html**.

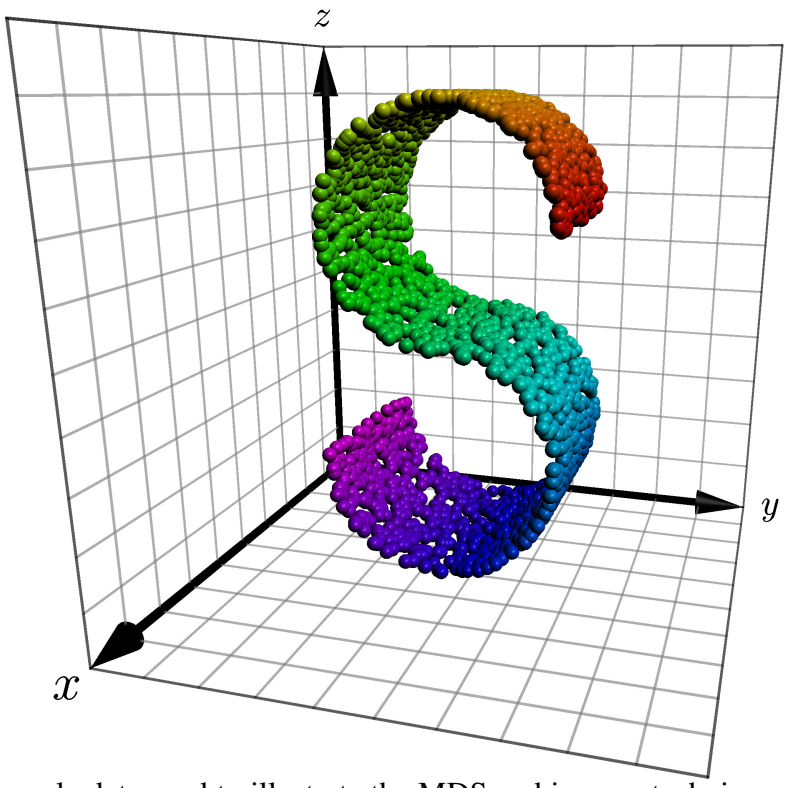


Figure 6.1: Example data used to illustrate the MDS and isomap techniques which fall on an S-shaped manifold in $\mathbb{R}^3$.

point. In practice, this rule takes either the form of a cutoff distance r_{cut} or an integer $N_{\text{neigh}}^{\text{Isomap}}$ which defines the nearest $N_{\text{neigh}}^{\text{Isomap}}$ data points to a given member of the data set to be its neighbors. Once the neighbors of each of the data points are defined, a graph $G(V, E)$ is constructed within which each data point is a vertex V and edges E are drawn to connect each of them to their neighbors. Each edge is assigned a "weight" corresponding to the Euclidean distance between the two connected data points in the original space. The length of a given path between two vertices in the graph G is then defined as the sum of the weights of the edges contained in the path, and the geodesic distance between two arbitrary data points is taken to be the distance of the shortest path which connects the two corresponding vertices in the graph.[10] Finally, the original coordinates of the data in $\mathbb{R}^q$ are projected onto the p most significant eigenvectors of the dissimilarity matrix formed by taking $\Xi_{ij} = g_{ij}$ $(i, j = 1, \ldots, N)$ to arrive at the final embedding coordinates. The application of the isomap algorithm (also from Scikit-learn) to the S-curve data using $N_{\text{neigh}}^{\text{Isomap}} = 10$ is shown in Figure 6.2, which clearly identifies the two-dimensional manifold on which the data lies.

[10]The determination of shortest path between two vertices of a graph is most often performed using *Dijkstra's algorithm* [279].

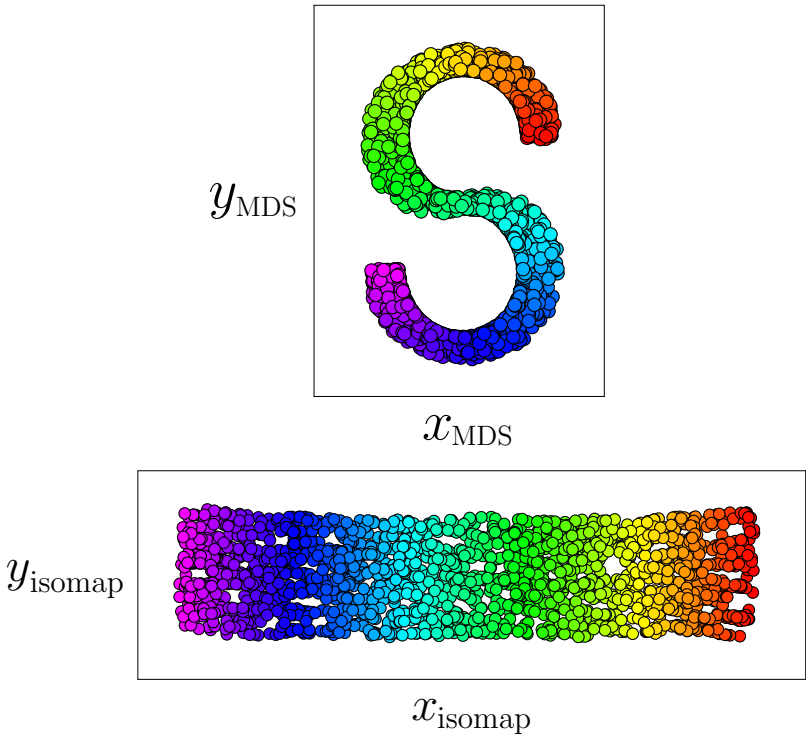


Figure 6.2: MDS and isomap embeddings of the example data shown in Figure 6.1.

6.3 Training set

The primary training set featured in the remainder of this chapter consists of monoatomic bulk lattices, unrelaxed surfaces, nanostructures, and clusters composed of elemental silicon. Altogether, it includes 2110 atomic configurations comprised of 15721 atoms. Below, we detail the construction of each group of the training set.

6.3.1 Bulk structures

The bulk configurations of the training set were based on ten ideal crystal structures: β-Sn, bc8, body-centered cubic (bcc), diamond, face-centered cubic (fcc), graphite, hexagonal close-packed (hcp), hexagonal diamond (hd), simple cubic (sc), and simple hexagonal (sh). The conventional unit cell and corresponding geometric parameters of each lattice (which were arbitrarily taken from the values quoted in Table IV of [261]) can be found in Appendix D along with their primitive descriptions. For each of the ten ideal bulk lattices in the training set, 50 perturbed versions were additionally included for a total of 510 bulk configurations. The perturbed structures were generated by first constructing an eight-atom unit cell from the primitive unit cell of each ideal lattice. Each of the lattice vectors of the resulting non-primitive unit cell were displaced in a random direction by a magnitude

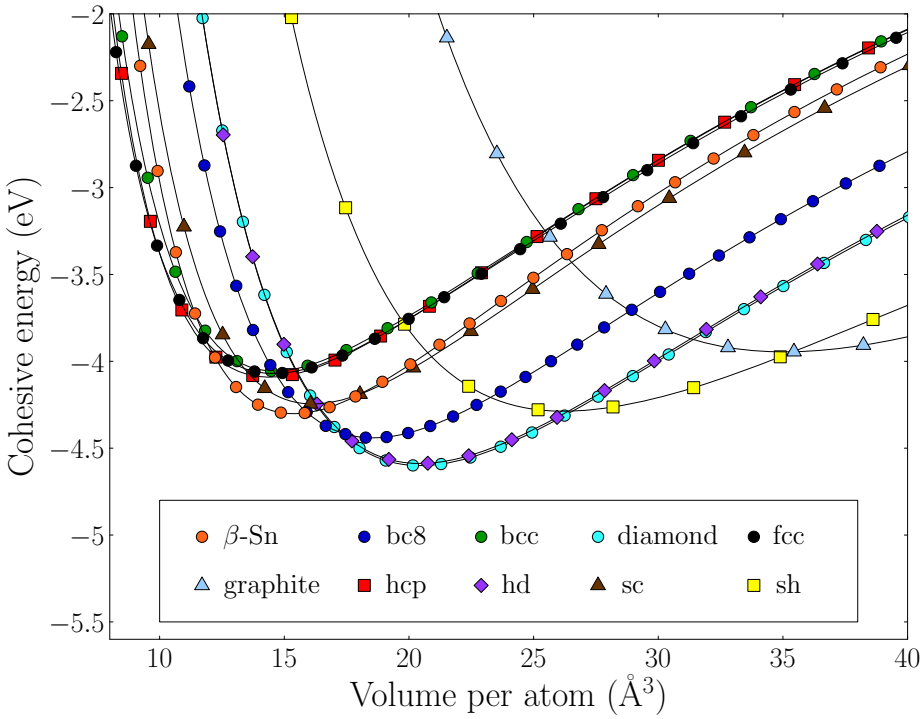


Figure 6.3: Cohesive energy curves of the ideal bulk structures as a function of average volume per atom as calculated by DFT. In cases where multiple lattice parameters are required the define the structure (β-Sn, bc8, graphite, hcp, hd, and sh), all quantities other than the primary lattice parameter a were held fixed.

uniformly sampled between five and ten percent of its magnitude, and each of the eight basis atoms was also displaced in a random direction by a magnitude uniformly sampled between 0 and 0.5 Å.

For comparison with results later in the chapter, Figure 6.3 shows the cohesive energy of each ideal bulk lattice as a function of volume per atom, as calculated by DFT. Lattices whose geometry is fully determined by a single primary lattice parameter a were computed at a range of its values, while lattices which require specification of additional lattice parameters (β-Sn, bc8, graphite, hcp, hd, and sh) had these parameters fixed and only a was varied. The diamond and hexagonal diamond structures are shown to be the most energetically favorable, followed by bc8, β-Sn, sh, sc, hcp, fcc, bcc, and graphite. Similar to Figure 6.3, the cohesive energy as a function of lattice constant is plotted in Figure 6.4 for each of these ten structures for the eight EPs listed in Table 6.1.

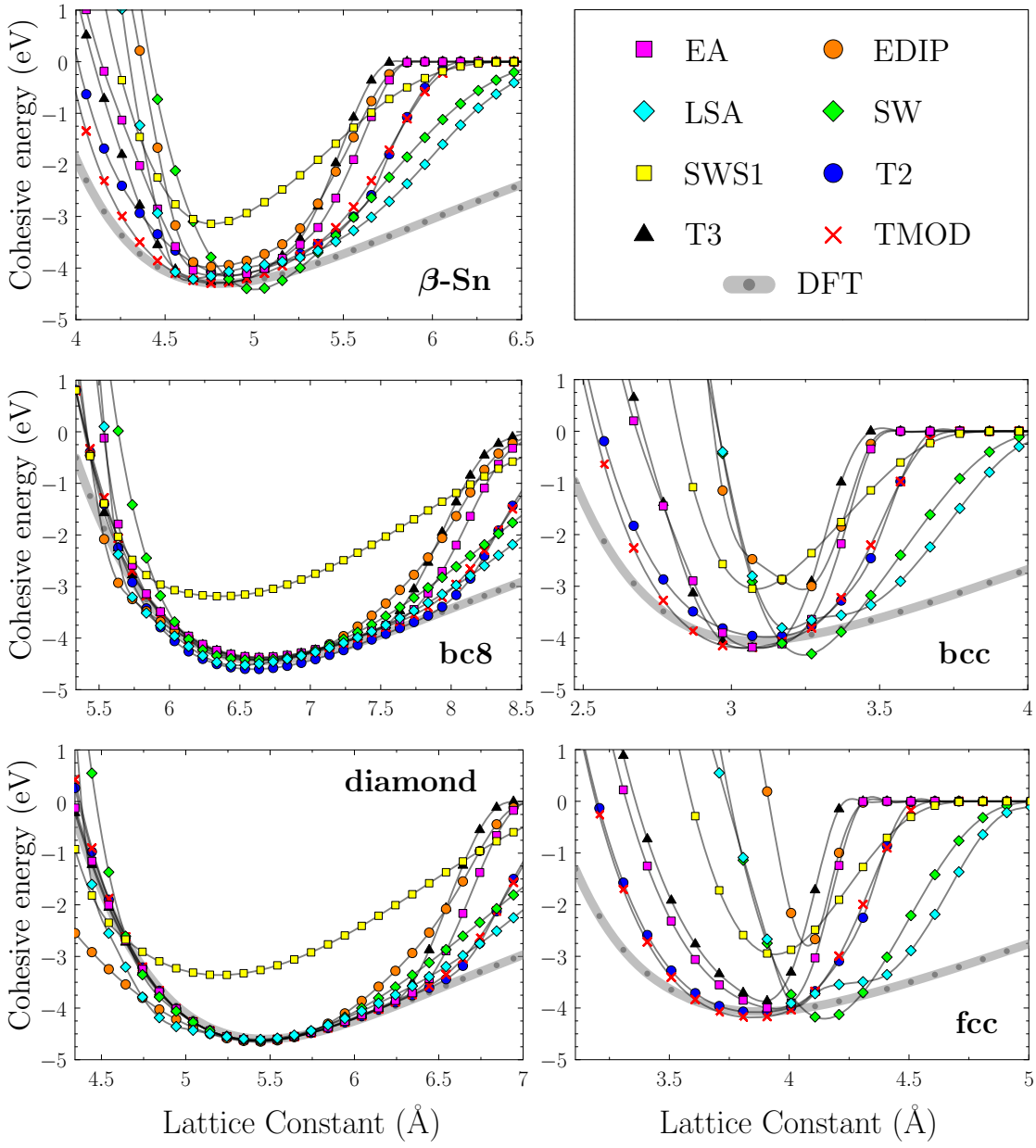


Figure 6.4 (Continued on next page)

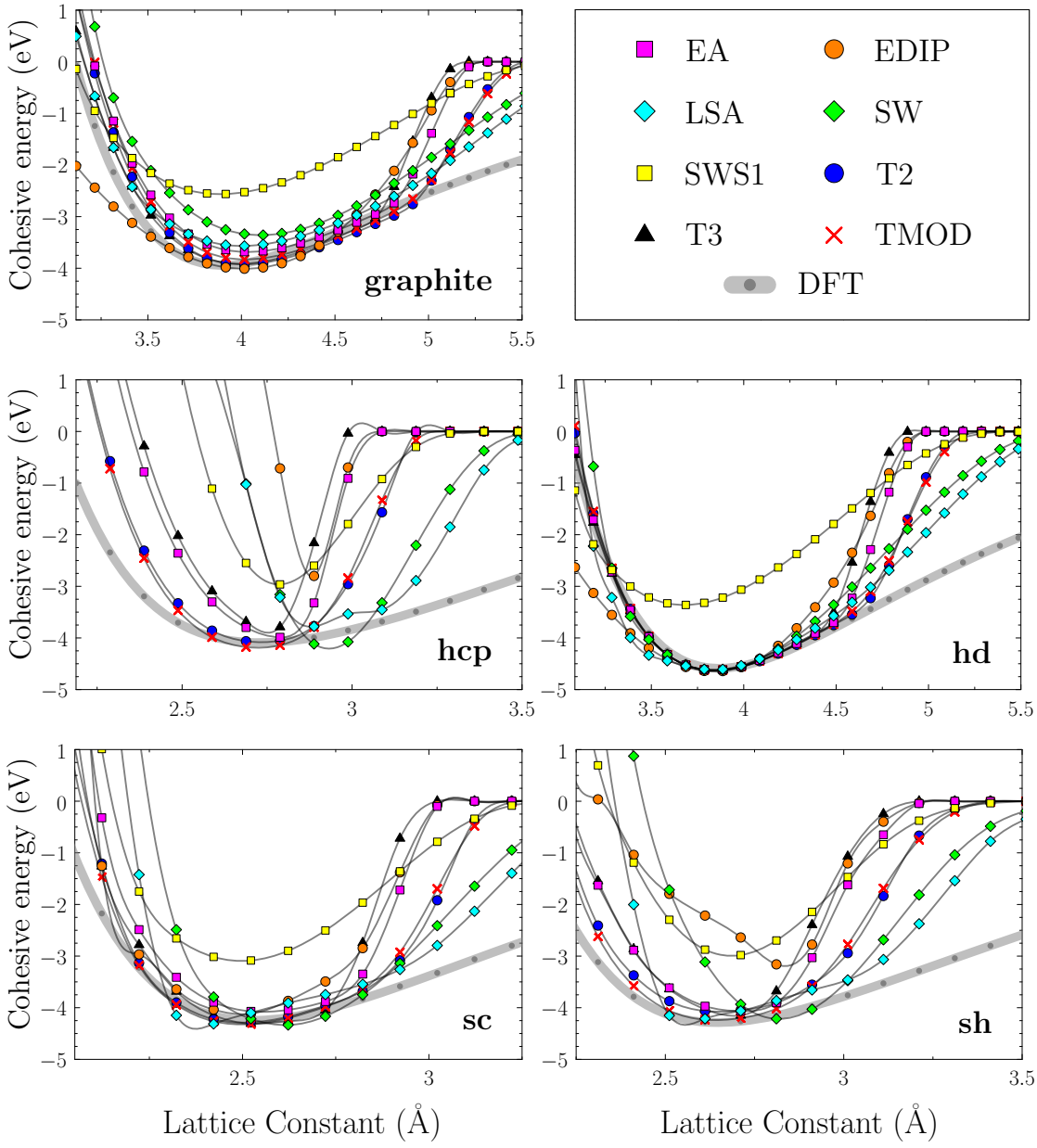
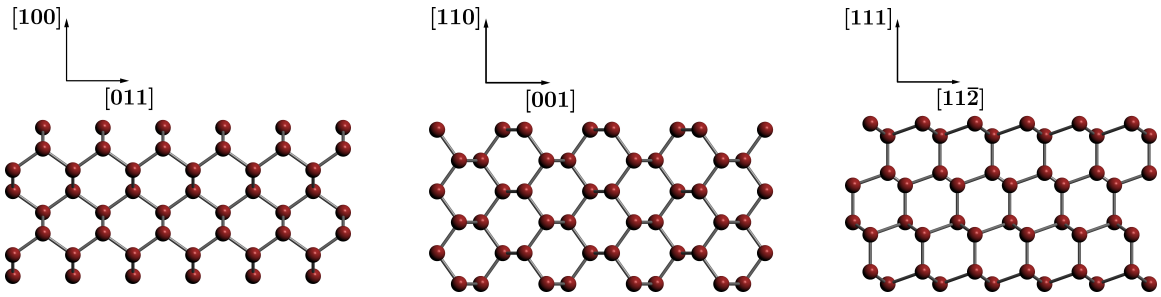


Figure 6.4: Cohesive energy curves of the ideal bulk structures as a function of lattice constant for each of the eight EPs listed in Table 6.1. In cases where multiple lattice parameters are required the define the structure (β-Sn, bc8, graphite, hexagonal close-packed, hexagonal diamond, hexagonal), all quantities other than the primary lattice constant a were held fixed.



(a) (100) surface (b) (110) surface (c) (111) surface

Figure 6.5: Unrelaxed surfaces of the ideal diamond lattice included in the training set.

6.3.2 Surfaces

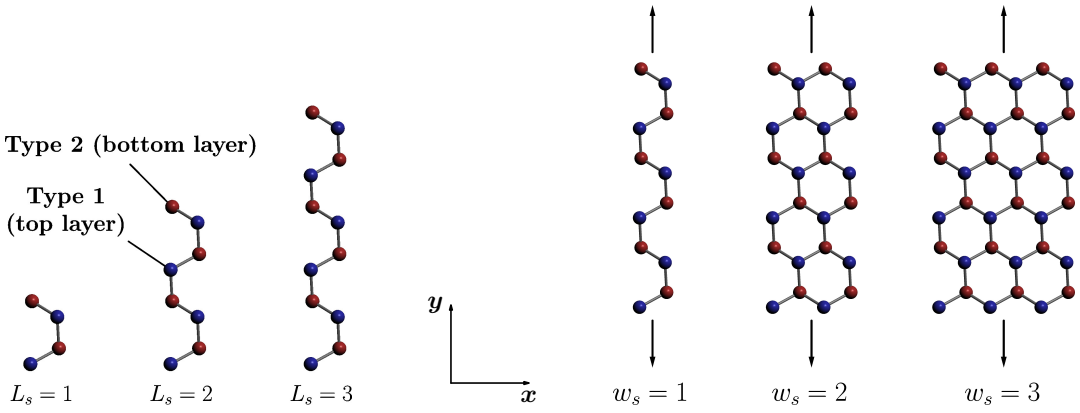
Three different unrelaxed surfaces of the ideal diamond lattice were considered. In terms of the conventional cubic unit cell of diamond, these are denoted the (100), (110), and (111) surfaces, and are shown in Figure 6.5.

6.3.3 Nanostructures

Graphite is a crystal lattice which is almost invariably thought of in terms of the layers of atoms which may be found stacked in AB sequence along its [001] direction. These layers, commonly known as *graphene* sheets, are fast becoming a staple of nanomechanics due to their enormous strength and electrical conductivity, and even garnered the 2010 Nobel Prize in Physics for Andre Geim and Konstantin Novoselov. Although graphene ordinarily consists of carbon atoms, we consider here graphene structures composed of silicon. Similar to the graphite lattice, the ground state diamond lattice of silicon can be thought of as being formed by a stacking of atomic layers along its [111] direction and, moreover, these layers bear a remarkable structural resemblance to graphene. However, the *silicene* layers found in diamond are found by performing alternating out-of-plane displacements of the atoms in a sheet of graphene, and are thus often said to be "buckled." Furthermore, unlike the AB stacking which forms graphite from graphene monolayers, the diamond structure is formed by an ABC stacking sequence. Below, we detail various nanostructures included in the training set which are built upon graphene and silicene, and include illustrations of these structures in the latter case.

6.3.3.1 Nanoribbons

As they are both periodic structures, unit cells of both graphene and silicene can be identified; the unit cell of graphene is two-dimensional since it is strictly planar, while the unit



(a) Finite-length (b) Infinite-length

Figure 6.6: A subset of the silicene nanoribbons included in the training set. The upper layer of atoms (larger z coordinate) is colored blue, while the lower layer of atoms is colored red. In (b), the black arrows indicate periodic boundary conditions.

cell of silicene is three-dimensional to account for its out-of-plane displacements relative to graphene. The first type of nanostructures in our training set are finite-width systems formed from these unit cells which are referred to as *nanoribbons*. In Figure 6.6a, we show atomic configurations produced by taking a single silicene unit cell and repeating it L_s times along the vertical direction to form finite-length silicene nanoribbons. In our training set, we include finite-length silicene nanoribbons of lengths $L_s = [1, 9]$ which have geometry consistent with the corresponding layers found in the ideal diamond lattice of the training set. Similarly, we also include finite-length graphene nanoribbons with lengths $L_g = [1, 9]$ and geometry consistent with the corresponding layers in the ideal graphite lattice previously mentioned. Making the above nanoribbons periodic along their length yields infinite-length nanoribbons, as shown in Figure 6.6b. In this case, the structural parameter of interest is the width of each nanoribbon, and the infinite-length nanoribbons included in the training set possess widths $w_s = [1, 9]$ and $w_g = [1, 9]$ for the silicene and graphene variants, respectively.

6.3.3.2 Nanosheets

Taking both in-plane directions of the nanoribbons to be periodic recovers the original silicene and graphene structures from which they were created, the former of which is shown to the left in Figure 6.7. Stacking layers of silicene and graphene, which we refer to as *nanosheets*, leads us to another portion of our training set. In the case of silicene, the layers are stacked in ABC sequence, as shown in the right of Figure 6.7, with an interlayer separation corresponding to that found in the ideal diamond lattice of Section 6.3.1. Thus,

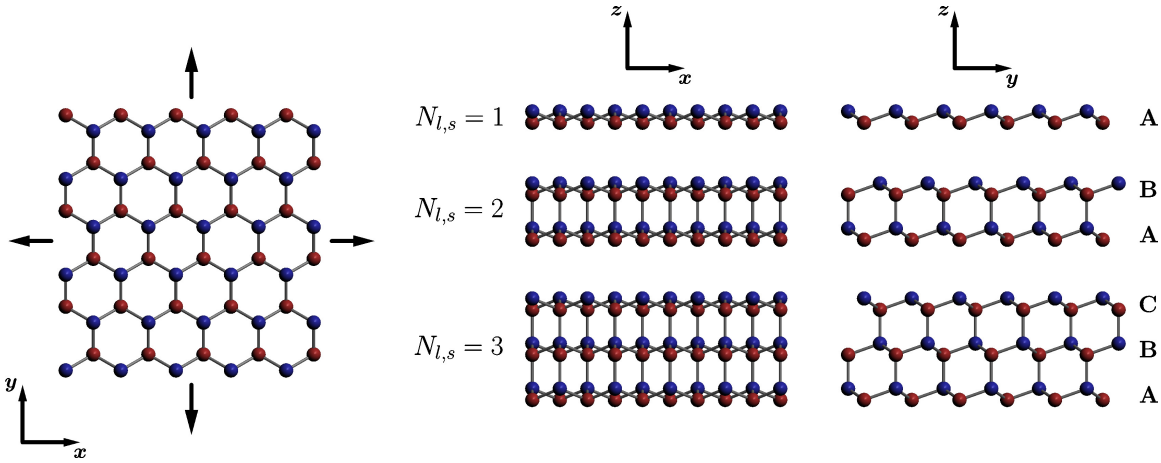


Figure 6.7: A subset of the silicene nanosheets in the training set. Atoms in the upper and lower layers of each silicene nanosheet are colored consistently with Figure 6.6.

while we include only silicene nanosheet stacks of heights $N_{l,s} = [1, 9]$ in the training set, it may be noted that the limit as the number of layers $N_{l,s} \to \infty$ would exactly reproduce the ideal diamond structure. Notice, as well, that for a large but finite $N_{l,s}$, the upper and lower surface terminations are approximately equivalent to the (111) surface of diamond shown in Figure 6.5c. The graphene nanosheets of the training set, which similarly have heights $N_{l,g} = [1, 9]$, are stacked in AB sequence so that the limit $N_{l,g} \to \infty$ identically recovers the ideal graphite structure.

6.3.3.3 Nanotubes

Rather than making the infinite-length silicene nanoribbons periodic along the x axis in Figure 6.6 to form a nanosheet, one can instead imagine rolling them around the y axis in order to join the finite boundaries on each side. Here, we specifically choose this transformation such that the plane which sits halfway between the upper (blue) and lower (red) planes of atoms is deformed isometrically in the process, i.e. the length of all lines which reside in this plane remain constant during the transformation.[11] This procedure results in silicene *nanotubes*, three of which are pictured in Figure 6.8. Each nanotube is denoted according to its circumference C_s, which is defined to be equal to the width w_s of the nanoribbon used to create it. However, because an infinite-length silicene nanoribbon of width $w_s = 1$ or 2 would lead to a nanotube with excessive energy, we limit the silicene nanotubes in the training set to circumferences $C_s = [3, 11]$. We similarly include graphene nanotubes of circumferences $C_g = [3, 11]$.

[11] This implies that the atoms in the upper plane are situated closer to one another in the nanotube than in the corresponding nanoribbon, while atoms in the lower plane are spread further apart in nanotube form.

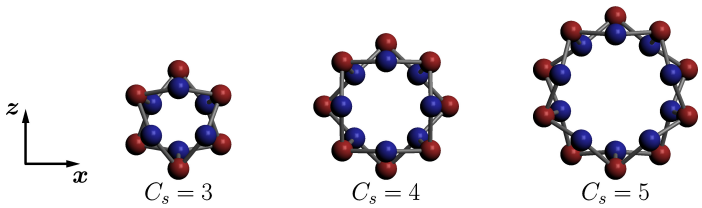


Figure 6.8: Three of the silicene nanotubes included in the training set. The structures are periodic along the y direction (perpendicular to the page).

6.3.4 Clusters

Beyond the bulk crystal lattices, surfaces, and nanostructures, a total of 1525 clusters containing between two and ten atoms were also a part of the training set. Included among these clusters were a total of 75 dimers with separations ranging from 1.5 Å to 8.0 Å in equal increments. The three- to eight-atom clusters fell into two groups according to the algorithms shown in Figure 6.9. The first algorithm, which generated *elongated clusters*, begins by placing a single atom at the origin. Next, a distance r is sampled from a normal distribution $\mathcal{N}(\mu = 2.35 \text{ Å}, \sigma^2 = 0.01 \text{ Å}^2)$[12]—if the sampled distance is greater than 1.8 Å, spherical angles θ and ϕ are sampled from uniform distributions unif$(0, \pi)$ and unif$(0, 2\pi)$, respectively, and the atom is placed at the resulting spherical coordinates (r, θ, ϕ). The sampling process repeats, and the coordinates of the next atom are placed at a position displaced from that of the previously added atom by the vector (r, θ, ϕ). Throughout the sampling procedure, any atom which is to be added to the system is required not to fall within 1.8 Å of any previously added atoms. The second algorithm, which was used to create *compact clusters*, is identical to the first other than the additional constraint that any candidate atom to be added to the system must fall within 3 Å of at least two previously added atoms. Unlike the former algorithm, this restriction results in relatively smaller, more convoluted clusters. While the nine- and ten-atom clusters were all generated using the compact cluster algorithm, the number of elongated and compact clusters containing three to eight atoms can be found in Table 6.4. Samples of elongated and compact clusters drawn from the training set are shown in Figure 6.10.

[12]The mean distance 2.35 Å was chosen to approximately match the nearest-neighbor distance of the ideal diamond lattice, while a variance of $\sigma^2 = 0.01$ Å^2 was used so that approximately 95% of distances sampled would fall within the range of 2.15 Å to 2.55 Å.

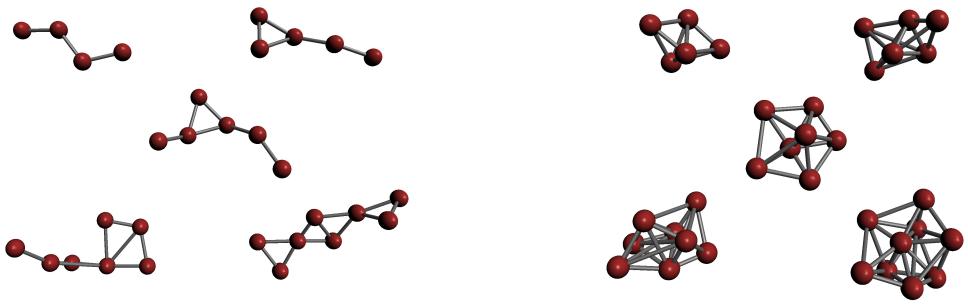
Elongated

Place atom 1 at origin
$N = 1$
Sample $r \sim \mathcal{N}(2.35, 0.01)$
while $r \leq 1.8$ Angstroms:
    Sample $r \sim \mathcal{N}(2.35, 0.01)$
Sample $\theta \sim \text{unif}(0, \pi)$, $\phi \sim \text{unif}(0, 2\pi)$
Place atom 2 at $\mathbf{r_2} = (r, \theta, \phi)$
$N = 2$
while $N < \mathcal{A}$:
    Sample $r \sim \mathcal{N}(2.35, 0.01)$
    Sample $\theta \sim \text{unif}(0, \pi)$, $\phi \sim \text{unif}(0, 2\pi)$
    $\mathbf{r_{N+1}} = \mathbf{r_N} + (r, \theta, \phi)$
    if $\mathbf{r_{N+1}}$ falls within 1.8 Angstroms of any existing atoms:
        pass
    else:
        Place atom $N + 1$ at $\mathbf{r_{N+1}}$
        $N = N + 1$

Compact

Place atom 1 at origin
$N = 1$
Sample $r \sim \mathcal{N}(2.35, 0.01)$
while $r \leq 1.8$ Angstroms:
    Sample $r \sim \mathcal{N}(2.35, 0.01)$
Sample $\theta \sim \text{unif}(0, \pi)$, $\phi \sim \text{unif}(0, 2\pi)$
Place atom 2 at $\mathbf{r_2} = (r, \theta, \phi)$
$N = 2$
while $N < \mathcal{A}$:
    Sample $r \sim \mathcal{N}(2.35, 0.01)$
    Sample $\theta \sim \text{unif}(0, \pi)$, $\phi \sim \text{unif}(0, 2\pi)$
    $\mathbf{r_{N+1}} = \mathbf{r_N} + (r, \theta, \phi)$
    if $\mathbf{r_{N+1}}$ falls within 1.8 Angstroms of any existing atoms **or** is not within 3 Angstroms of at least two existing atoms:
        pass
    else:
        Place atom $N + 1$ at $\mathbf{r_{N+1}}$
        $N = N + 1$

Figure 6.9: Pseudocode describing how the elongated and compact cluster configurations of the training set were generated. The notation $\mathcal{N}(\mu, \sigma^2)$ represents a univariate normal distribution with mean μ and variance σ^2. The notation $\text{unif}(a, b)$ represents a uniform distribution over the interval (a, b). All positions are specified in spherical coordinates, where r is in units of Å, while the polar and azimuthal angles θ and ϕ are in radians. The variable $\mathcal{A}$ represents the total number of desired atoms in the cluster.



(a) Elongated

(b) Compact

Figure 6.10: Sample cluster configurations in the training set.

6.3.5 Summary

The entire training set are summarized in Tables 6.3 and 6.4 below, which also indicate the number of atomic environments, i.e. the number of atoms, present across each subdivision of configurations. For example, there were 50 perturbed β-Sn configurations in the training set, each of which contained eight atoms, so there are 400 atomic environments across this set of configurations.

Structure category	Subcategory	Subtype	Number of training configurations	Number of atomic environments
Bulks	β-Sn	ideal	1	8
		perturbed	50	400
	bc8	ideal	1	8
		perturbed	50	400
	bcc	ideal	1	8
		perturbed	50	400
	diamond	ideal	1	8
		perturbed	50	400
	fcc	ideal	1	8
		perturbed	50	400
	graphite	ideal	1	8
		perturbed	50	400
	hcp	ideal	1	8
		perturbed	50	400
	hexagonal diamond	ideal	1	8
		perturbed	50	400
	sc	ideal	1	8
		perturbed	50	400
	sh	ideal	1	8
		perturbed	50	400
	Total counts:		**510**	**4080**
Surfaces	(100)	ideal	1	48
	(110)	ideal	1	48
	(111)	ideal	1	36
	Total counts:		**3**	**132**

Table 6.3: Bulk and surface configurations present in the training set. See text for details.

Structure category	Subcategory	Subtype	Number of training configurations	Number of atomic environments
Nanostructures	nanoribbons	graphene	18	360
		silicene	18	360
	nanosheets	graphene	9	180
		silicene	9	180
	nanotubes	graphene	9	252
		silicene	9	252
	Total counts:		**72**	**1584**
Clusters	dimer	N/A	75	150
	trimer	elongated	50	150
		compact	100	300
	4-atom	elongated	50	200
		compact	100	400
	5-atom	elongated	75	375
		compact	100	500
	6-atom	elongated	75	450
		compact	100	600
	7-atom	elongated	100	700
		compact	100	700
	8-atom	elongated	100	800
		compact	100	800
	9-atom	elongated	0	0
		compact	200	1800
	10-atom	elongated	0	0
		compact	200	2000
	Total counts:		**1525**	**9925**

Table 6.4: Cluster and nanostructure configurations present in the training set. See text for details.

6.4 Results and analysis

6.4.1 Atomic configurations

The prediction surfaces which are defined by nonparametric regression methods are usually considered to be, to a large extent, inscrutable to human intuition because they have no closed form representation independent of the size of their training set. However, because the covariance function is the fundamental object which defines a Gaussian process, it is natural to inquire whether it is possible to use it to visualize the internal anatomy of the predictive equations (6.1) and (6.2) in order to glean a better understanding of the end regression which is performed. We commence our study of these relations by first examining the apparent quantities which are explicitly used to perform inference. These consist of the target values $\mathbf{y}$ and the configuration covariance matrices $\mathcal{K}$, where we take the test set $\boldsymbol{\mathcal{C}}_*$ to be equal to the training set $\boldsymbol{\mathcal{C}}$ so that both covariance matrices present are equal to $\mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}})$. Although in RATE we take the values $\mathbf{y}$ to be the total energy errors $\mathcal{V}_{\mathcal{C}}^{\Delta}$ of EPs for various atomic configurations, we also let the first-principles total energies $\mathcal{V}_{\mathcal{C}}^{\mathrm{DFT}}$ play the role of the target observations during our investigation so that they might serve as a baseline for comparison.

The entries of the configuration covariance $\mathcal{K}(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}})$ are generally unbounded, as there is no limit to the number of atoms that any of the training configurations $\mathcal{C}$ may contain (cf. (6.3)). By normalizing these quantities, however, it is possible to define a dissimilarity measure between configurations $\mathcal{C}_i$ and $\mathcal{C}_j$ according to

$$\Xi(\mathcal{C}_i, \mathcal{C}_j) \triangleq 1 - \frac{\mathcal{K}_{ij}}{\sqrt{\mathcal{K}_{ii}}\,\sqrt{\mathcal{K}_{jj}}}, \tag{6.11}$$

similar to what was done in the SOAP kernel in (4.97). Note that we have used the notation $\mathcal{K}_{ij} \triangleq \mathcal{K}(\mathcal{C}_i, \mathcal{C}_j)$, and that the resulting dissimilarity $\Xi(\mathcal{C}_i, \mathcal{C}_j)$ is always confined to the unit interval. These dissimilarities can be used with the MDS algorithm outlined in Section 6.2.4 to construct a low-dimensional embedding in which each data point corresponds to an atomic configuration and their relative distances reflect the covariance between them which defines their similarity in RATE. Here, the MDS procedure was repeated for embedding dimensionalities ranging from two to nine, where the function f in (6.6) is taken to be the identity. For each dimensionality, 40 initial configuration seeds were used and relaxation was continued until either the stress differed by less than 1e-6 between successive steps or 15000 total relaxation steps were reached. The lowest resulting final Kruskal stress across the 40 optimizations performed for each dimensionality is shown in Figure 6.11. As

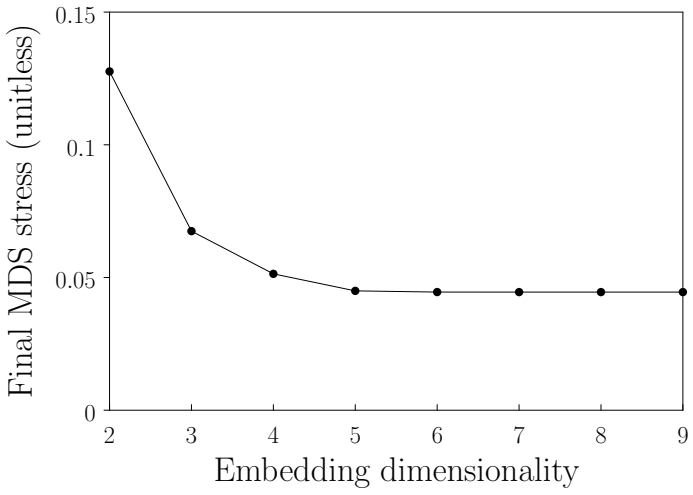


Figure 6.11: Kruskal stress of the MDS procedure for the total atomic configurations of the entire training set as a function of the embedding dimensionality.

expected, allowing higher dimensionality of the embedding gives the algorithm more flexibility in determining a consistent set of coordinates and results in a smaller stress. Although a monotonic decrease in the final stress is observed with increasing dimensionality, using three dimensions already reduces it to a level near the converged value, and should thus provide a reasonably faithful representation of the data.

In Figures 6.12 – 6.14, we plot the resulting MDS coordinates of the training set configurations in $\mathbb{R}^3$ from three different perspectives. The varied shading depicted in these figures is selected to highlight the location of different subgroups of the training set. The small groups of points on the right of Figure 6.12 correspond to the ideal and perturbed bulk configurations, while the large, grey mass to the left corresponds to the cluster configurations. The transparent spheres, cubes, tetrahedra, and cylinders represent the MDS positions of the finite-length nanoribbons, infinite-length nanoribbons, nanosheets, and nanotubes, respectively. The two shadings of these objects (green and pink) correspond to the graphene and silicene variants of each, respectively, and will be discussed shortly. The three transparent octahedra, which all fall at nearly the same position, represent the surface configurations.

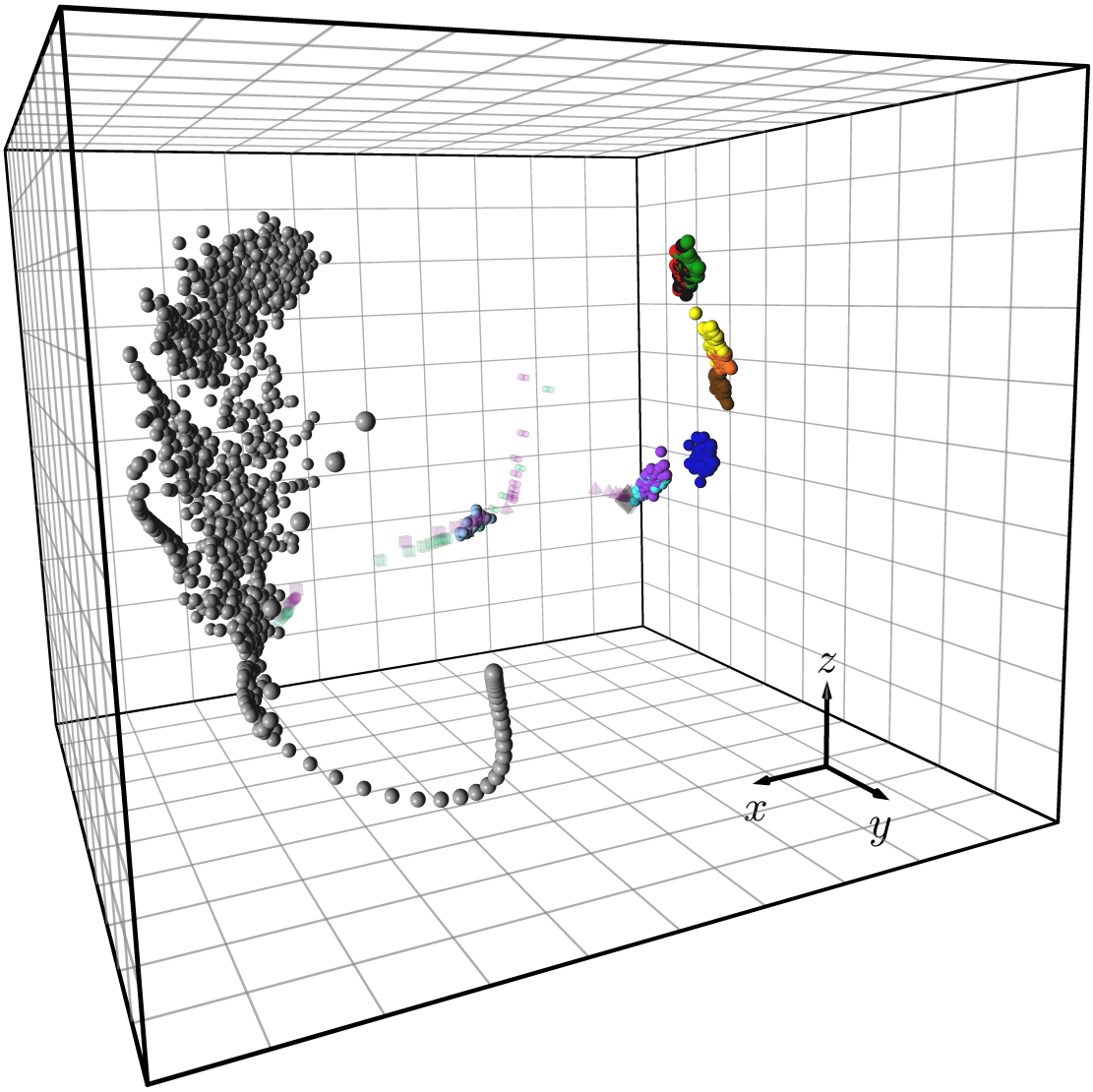


Figure 6.12: Three-dimensional MDS embedding of the 2110 atomic configurations of the training set. Figures 6.13 and 6.14 depict alternative perspectives of the same data.

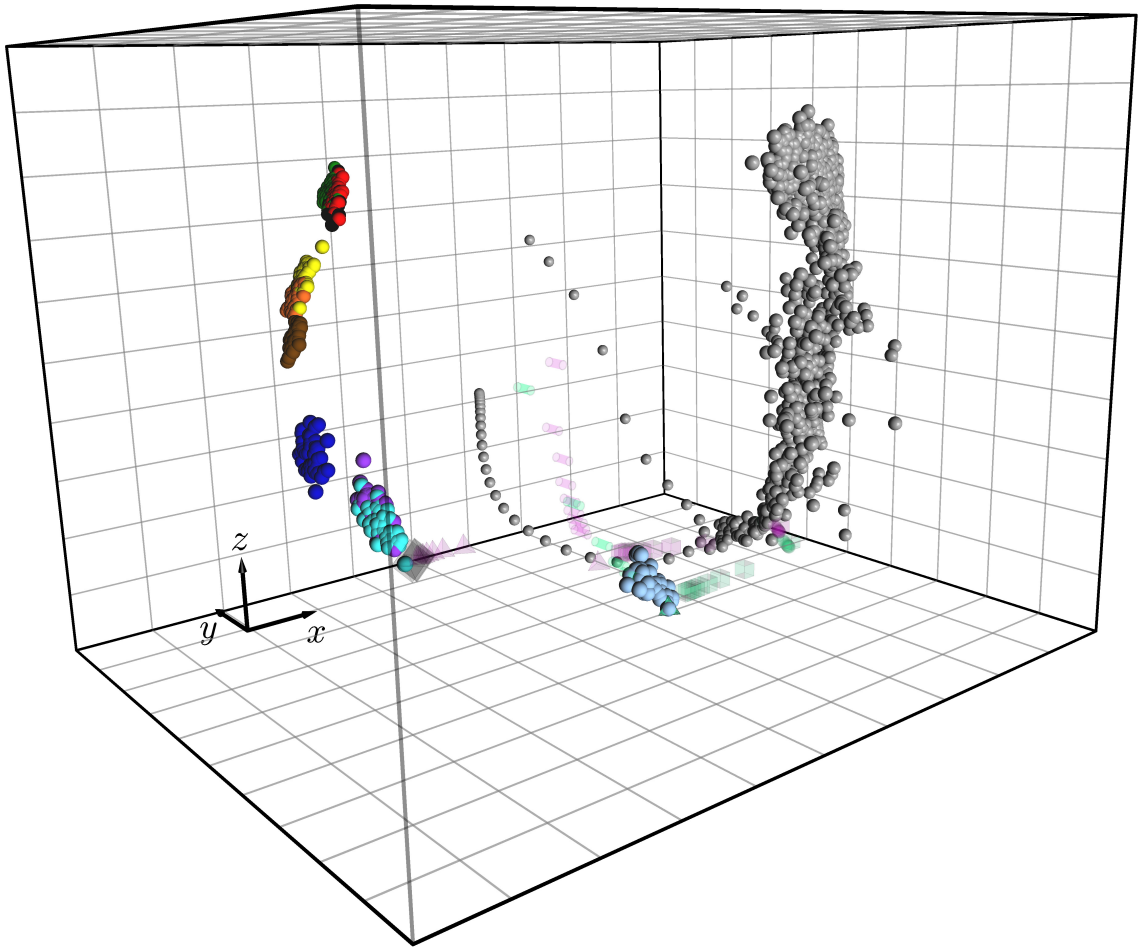


Figure 6.13: Three-dimensional MDS embedding of the 2110 atomic configurations of the training set. Figures 6.12 and 6.14 depict alternative perspectives of the same data.

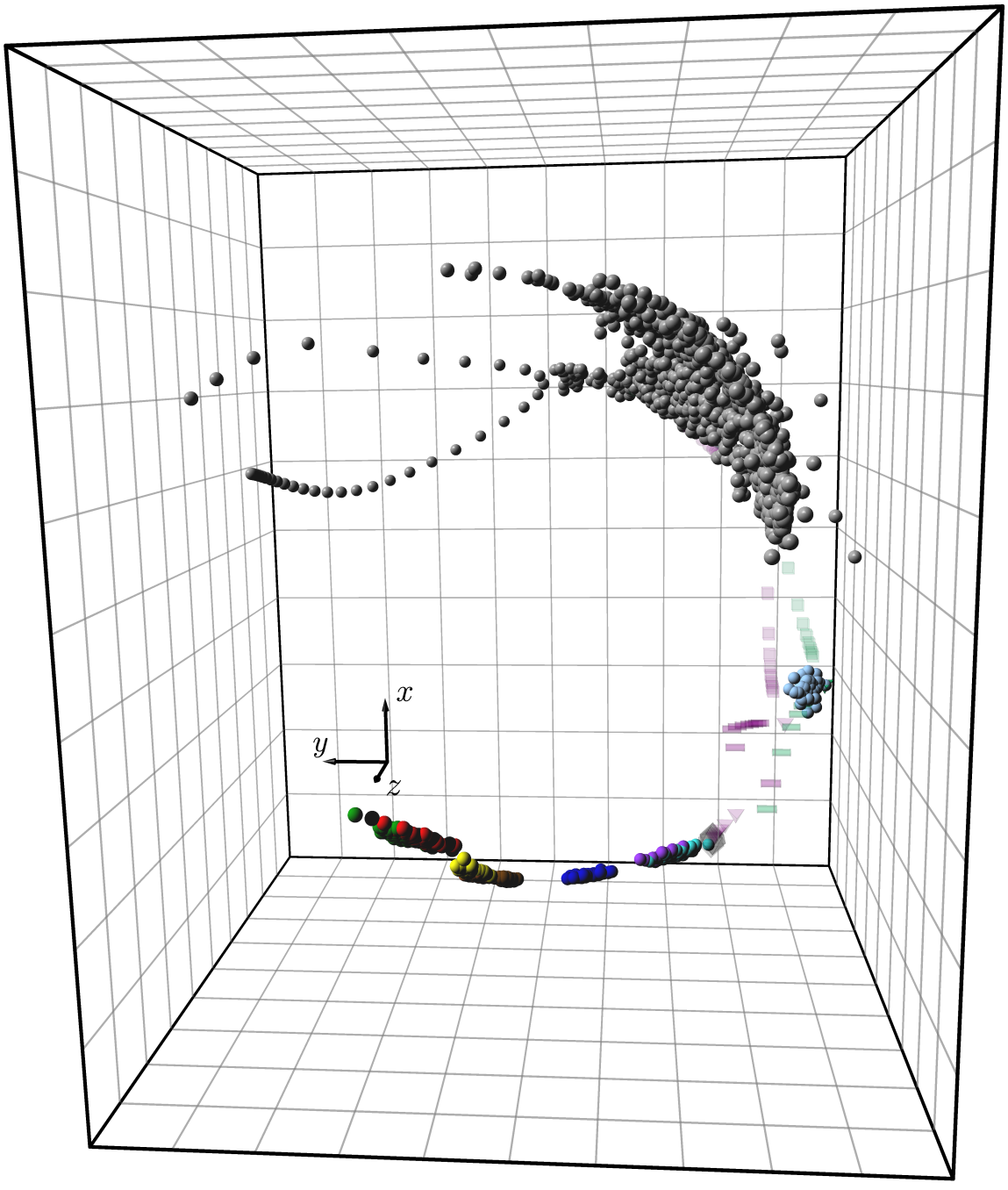


Figure 6.14: Three-dimensional MDS embedding of the 2110 atomic configurations of the training set. Figures 6.12 and 6.13 depict alternative perspectives of the same data.

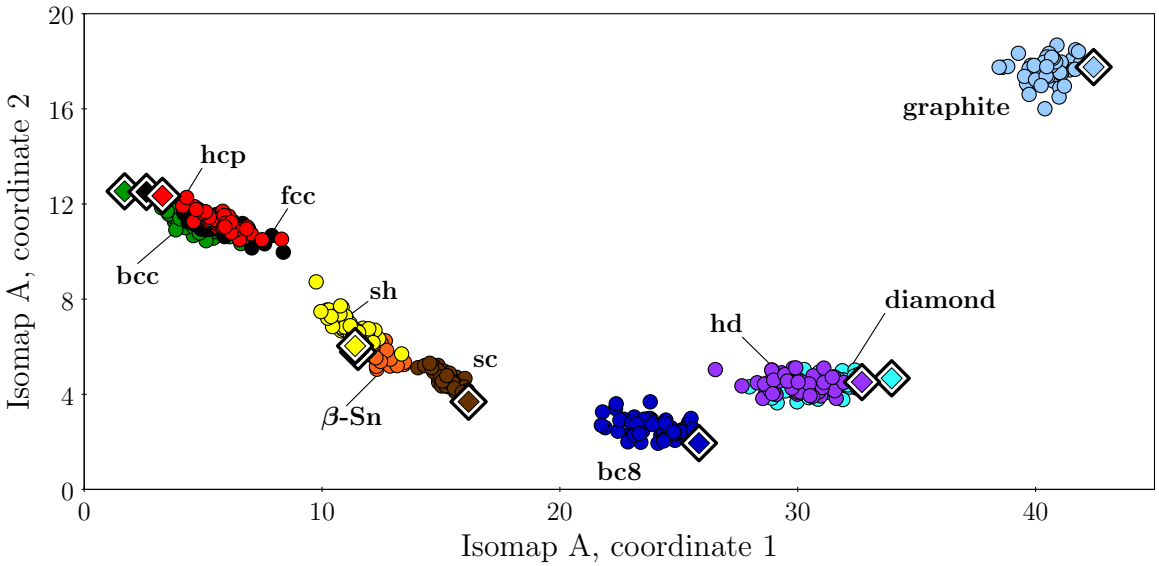


Figure 6.15: Isomap embedding of three-dimensional MDS coordinates of the bulk configurations. Each type of bulk structure is indicated with a different color, and the diamond-shaped symbols indicate the location of the ideal bulk structures.

In order to further simplify the visualization of the MDS coordinates, we may attempt to reduce the dimensionality once more by appealing to the isomap algorithm of Section 6.2.4. We first apply the algorithm to the collection of 510 bulk configurations, which we term 'Isomap A'. A tolerance of 1e-6 was used with automatic eigensolver and shortest-path method determination in the isomap implementation of the Scikit-learn library. Because the overall manifold on which these points lie is only mildly deformed from being planar, the results of the isomap procedure were found to depend only weakly on the number of neighbors $N_{\text{neigh}}^{\text{Isomap}}$ chosen, and selecting the full complement of neighbors ($N_{\text{neigh}}^{\text{Isomap}} = 509$) produced a satisfactory embedding (more or less equivalent to a direct two-dimensional projection). The coordinates determined from this isomap are shown in Figure 6.15. First, we note that the coincidence of the diamond and hexagonal diamond structures is expected, as the former is composed of two interpenetrating fcc lattices while the latter is composed of two interpenetrating hcp lattices, giving the two nearly identical densities and cohesive energies (cf. Figure 6.3).[13] Moreover, the arrangement of configurations in Figure 6.15 is in keeping with the experimentally observed pressure dependence of silicon bulk phases quoted in [261] and [280]. Starting from standard temperature and pressure, increasing pressure results in the following sequence of phase transformations: diamond$\rightarrow\beta$-Sn$\rightarrow$sh$\rightarrow$hcp$\rightarrow$fcc. The bc8 structure, meanwhile, is obtained by starting in the β-Sn phase and decreasing the pressure. The fact that graphite sits apart from the other structures in

[13]The fcc and hcp lattices differ only in their stacking order (ABC and AB, respectively).

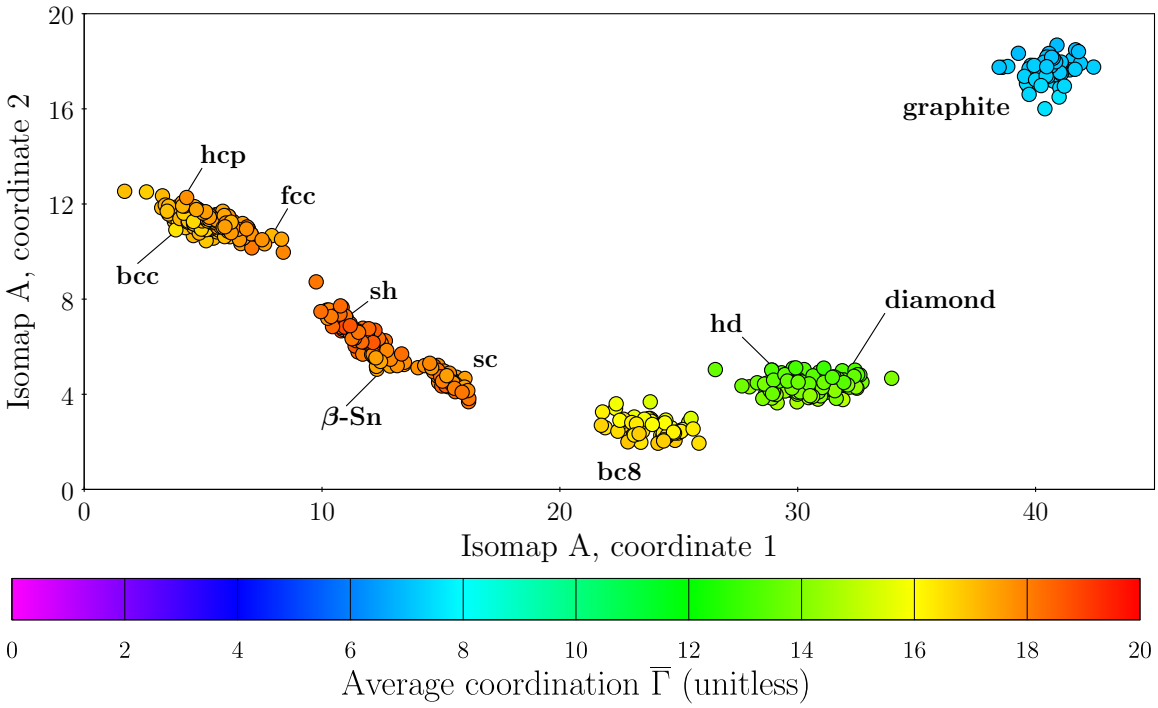


Figure 6.16: Isomap A embedding of the MDS coordinates of the bulk configurations, shaded according to the values of the average coordination of the atoms in each configuration as given in (6.13).

Figure 6.15, being closest to diamond, is plausible; while there is evidence of direct phase transitions from graphite to diamond in carbon [281–283], it requires somewhat drastic structural rearrangement. On the other hand, the close proximity of bcc, fcc, and hcp is also sensible considering their similar atomic packing factors (approximately 0.68 for bcc and 0.74 for fcc and hcp) and their coupled role in the martensitic transformations of iron [284–286]. A simple pattern consistent with the above observations can be revealed by defining a coordination function Γ for an arbitrary atom α with N_{neigh} neighbors indexed by β as

$$\Gamma(\alpha) \triangleq \sum_{\beta=1}^{N_{\text{neigh}}} f_{\text{cut}}(r_{\alpha\beta}), \tag{6.12}$$

where f_{cut} is the cutoff function of (4.84) used in the SOAP descriptor with the parameters listed in Table 6.2. Further, define the average of this function for a configuration $\mathcal{C}$ containing $\mathcal{A}_\mathcal{C}$ atoms as

$$\overline{\Gamma}_\mathcal{C} \triangleq \frac{1}{\mathcal{A}_\mathcal{C}} \sum_{\alpha=1}^{\mathcal{A}_\mathcal{C}} \Gamma(\alpha). \tag{6.13}$$

Plotting the latter quantity over the Isomap A coordinates in Figure 6.16, we see a smooth variation across the structures, with the highest $\overline{\Gamma}$ occurring for the perturbed sh and β-Sn

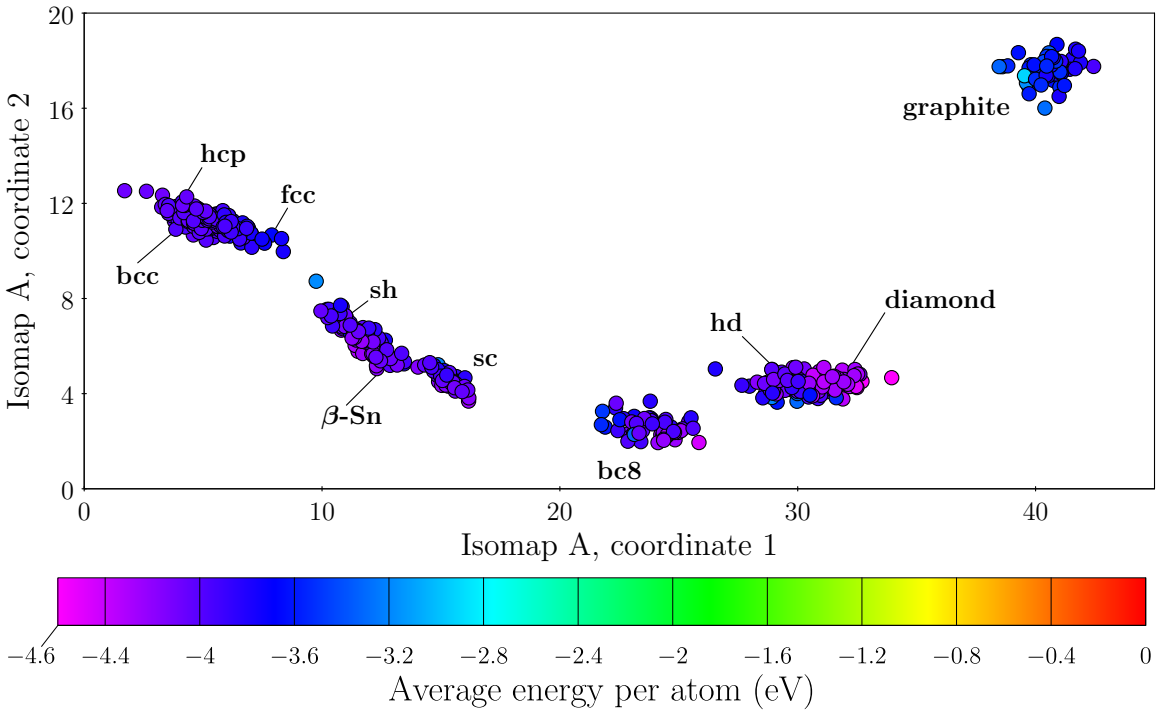


Figure 6.17: Isomap A embedding of the MDS coordinates of the bulk configurations, shaded according to the average first-principles energy per atom in each configuration.

structures[14] and the lowest values occurring for the undercoordinated perturbed graphite structures. Because $\overline{\Gamma}$ has no angular dependence, its smooth variation in Figure 6.16 does not necessarily guarantee that the energy will vary smoothly over the Isomap A coordinates, as well. However, plotting the average bulk energies (found by dividing the total energy output from the periodic DFT calculations by the number of atoms in each unit cell) in Figure 6.17 reveals a fairly modest variation, with nearly all average energies falling between -3.6 eV and -4.6 eV. Although the curves plotted in Figure 6.3 correspond to uniform lattice dilations rather than the random perturbations of the bulk lattices found in the training set, the results shown in Figure 6.17 indicate that the vast majority of the perturbed bulk structures possess average energies near the global minima of each corresponding curve in the former. Unsurprisingly, the perturbed diamond and hexagonal diamond structures are still the most energetically preferable, followed roughly in sequence by bc8, β-Sn, sh, and sc. The perturbed hcp, fcc, bcc, and graphite structures possess the highest average energies.

Next, we plot the average energy error (the total energy error $\mathcal{V}_{\mathcal{C}}^{\Delta}$ of (6.4) divided by the number of atoms) for the eight EPs of Table 6.1 over the Isomap A coordinates in Figures 6.18 and 6.19. We pause to emphasize that, as for the first-principles energies in

[14]The average coordination is higher for many of the sh and β-Sn structures than for the fcc and hcp structures because of the specific lattice parameters used, which can be found in Appendix D.

Figure 6.17, the energy errors shown in Figures 6.18 and 6.19 are not predicted by RATE, but are rather the quantities $\mathbf{y}$ which are used to train RATE to learn an atomic energy error function for each EP using (6.1).[15] As pointed out above for the case of average energy, the lattices used to construct the curves in Figure 6.4 are obtained by merely varying the primary lattice constant, as opposed to randomly displacing the lattice vectors and basis atoms as was done to generate the perturbed bulk configurations represented in Figures 6.18 and 6.19. Nevertheless, there similarly exists a reasonable correspondence between these two sets of results. Other than the SWS1 potential, which was fitted specifically to model monolayer silicene, all of the potentials accurately predict the average energies of the perturbed diamond and hexagonal diamond structures, as would be expected given that these types of structures are strongly emphasized during their design and parametrization. However, transferability to the other bulk phases is inconsistent, with the EPs generally tending to overpredict the average energy. That is, the EPs tend to predict an average energy closer to zero than the corresponding first-principles result, particularly in the cases of the fcc, hcp, and sh structures.

One may observe in Figure 6.18 that the average energy error of SW (Section C.4) grows larger for configurations which are farther away from diamond and hexagonal diamond in the isomap coordinates. This is to be expected from the fact that it possesses a significant bias toward the tetrahedral bonding found in diamond, as indicated by the corresponding angular dependence in Figure C.5. Thus, SW provides an inaccurate description of the perturbed graphite structures because the intralayer bonding within is roughly hexagonal. More egregious, however, are its average energy errors for the compressed bulk phases: sc, β-Sn, sh, bcc, fcc, and hcp (the nominal coordinations of which are 6, 6, 8, 8, 12, and 12, respectively). Initially, this behavior may seemingly be owed to the fact that there are many non-tetrahedral bond angles in the compressed bulk geometries. However, there is a more fundamental explanation for the failures of SW in this regard. While bonding in the low coordination phases (diamond, graphite, bc8) is covalent and, thus, energetically sensitive to angular distortions, bonding in the compressed phases is metallic in nature.[16] Because there is no strong notion of bond directionality in metals, the physical assumptions inherent to SW become invalid in these cases and preclude it from making accurate predictions.

[15] In these diagrams, and in all diagrams to follow in which variable shading is used, the left end of the color scale encompasses all values which are less than or equal to the value displayed (-0.5 eV in the present cases), and the right end of the color scale represents all values greater than or equal to the value displayed (2 eV in the present cases).

[16] In fact, the qualitative discrepancy which exists in the bonding of silicon is especially pathological from a modeling standpoint because all of its bulk phases have relatively similar energies; Tersoff [124] referred to this as the "polymorphous perversity" of silicon.

The SWS1 potential (Section C.5) experiences similar problems in Figure 6.19, but also appears to overpredict all energies by approximately 1.25 eV/atom. It is somewhat counter to expectation that it is not more accurate for the graphite structures, given the resemblance between individual graphene layers and silicene. However, this is accounted for by the fact that SWS1 was only fitted to reproduce the buckled geometry and thermal transport properties of silicene rather than its explicit energetics.

In contrast to SW, the T2 potential (Section C.6) previously discussed in Section 5.1 possesses only weak dependence on bond angle, which lends it accuracy for the metallic polytypes which is superior to nearly all of the other EPs. However, despite the robustness of T2 which might be implied by Figure 6.19, an important attribute not reflected in these results is the prediction of the elastic constants of diamond. Due to its diminished angular sensitivity, T2 woefully underpredicts C_{44}, leaving the diamond lattice far less stable to shear stress than indicated by experiment (Table C.1). An attempt at rectifying this shortcoming was made in the form of the T3 potential (Section C.7), where angular forces similar to those of SW are introduced. Unfortunately, while T3 does markedly improve on the description of elastic constants compared to T2, it is met with diminished fidelity for the metallic phases in Figure 6.19, as one would expect from the discussion above. An alternative compromise of radial and angular bond sensitivity in the Tersoff framework is found in the EA potential, which provides more accurate average energies for the metallic bulks while simultaneously also yielding elastic behavior in better agreement with experiment, albeit it is less accurate for graphite than T3 according to our results. The last Tersoff-type potential of our study, TMOD, is anomalous in the sense that it features some of the strongest three-body contributions among the EPs we consider and closely reproduces the elastic properties of diamond (and even, ostensibly, the melting point) shown in Table C.1, yet still offers a characterization of compressed polytypes which rivals that of T2. Although TMOD possesses a relatively deep two-body energy function which evidently suffices to compensate for its three-body interactions in the case of metallic phases,[17] this outcome is quite unexpected given that physical intuition dictates that only pairwise interactions are relevant in metals. Indeed, it is a reminder of the complexity intrinsic to many-body potentials, particularly cluster functionals, which oftentimes feature a delicate interplay of their various terms (two-body, three-body, etc.) in order to attain transferability.

Beyond the Tersoff potentials, EDIP (Section C.2) is another example of a bond order po-

[17] With modest manipulation, the general Tersoff form can be divided into two logical components. The first of these functions can be interpreted a two-body energy which depends only on bond length, while the second completely encapsulates the contributions due to bond order, and may thus be interpreted as a "three-body energy." See Section C.9 for details.

tential. However, unlike the four Tersoff potentials above, the bond order in EDIP has no angular dependence. Rather, the central quantity of EDIP is a coordination parameter Z_α which is calculated for a given atom α according to a pairwise summation over its neighbors of a function which resembles the cutoff functions used in the Tersoff potentials (Figure C.2). This quantity then determines the bond order, which augments the two-body energy as shown in Figure C.3. Concurrently, Z_α also independently serves as input to the three-body energy term of EDIP, altering the sensitivity of its angular dependence as well as its equilibrium bond angle. This progression is demonstrated in Figure C.5, where it can be seen that the three-body energy is minimized at the hexagonal angle for $Z_\alpha = 3$ (corresponding to graphite) with high bond stiffness, resembles the three-body energy of SW for $Z_\alpha = 4$ (diamond), and gradually softens until it is nearly zero at $Z_\alpha = 12$ (hcp, fcc). Because of its adaptive philosophy, EDIP manages to produce reasonably good estimates of the diamond elastic constants while still appropriately modeling graphite. However, the error pattern of EDIP shown in Figure 6.18 reveals that, in spite of the suppression of the three-body energy of EDIP at high coordination, it is no more transferable to the metallic phases than SW. From Figure C.3, it appears that the bond order dependence of the two-body energy is to blame for this behavior, as it becomes unphysically high (and even monotonic) as coordinations near $Z_\alpha = 12$ are approached.

Finally, we consider the LSA cluster functional (Section C.3) of Lenosky et al. [179]. As discussed at the end of Section 5.2.1, it is composed entirely of spline functions, some of which clearly lack physical interpretation. Unlike the aforementioned EPs, LSA is based upon the MEAM formalism, consisting of a two-body energy term and an embedding energy $U[\varrho^{\mathrm{CF}}]$ which incorporates both two-body and three-body summations.[18] Because of this discrepancy, it is difficult to make direct comparisons between LSA and the other EPs of our study. While the three-body energy is always non-negative in the EPs examined thus far, the embedding energy of LSA (which may be loosely construed as its three-body energy) can be either positive or negative. However, the fact that U is approximately linear in ϱ^{CF} with positive slope allows for at least a qualitative comparison to be made between ϱ^{CF} and the three-body energies of the other potentials. In particular, the form of ϱ^{CF} mimics that of SW in the sense that both explicitly penalize deviations from the geometry of the ideal diamond lattice. Accordingly, it can be seen that like SW, LSA develops greater average energy errors for configurations furthest from diamond in Figure 6.18. However, it does perform significantly better for the metallic polytypes with nominal coordinations of eight or less, i.e. all of them other than hcp and fcc.

[18] The notation ϱ^{CF} is used to indicate that the embedding density has angular dependence, and thus belongs to a cluster functional (CF). See Section 2.2 for details.

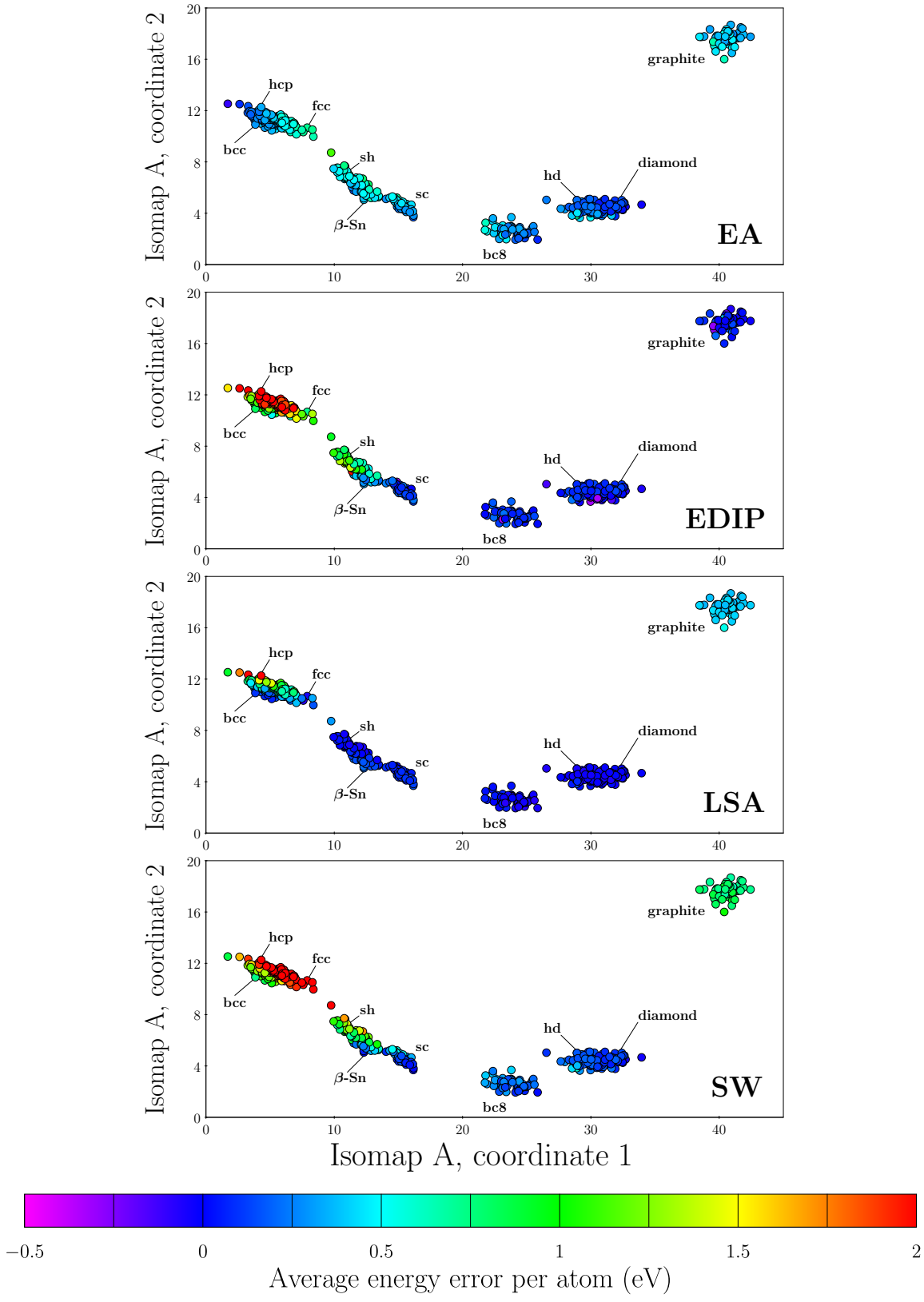


Figure 6.18: Average energy error per atom of the EA, EDIP, LSA, and SW potentials over the Isomap A coordinates.

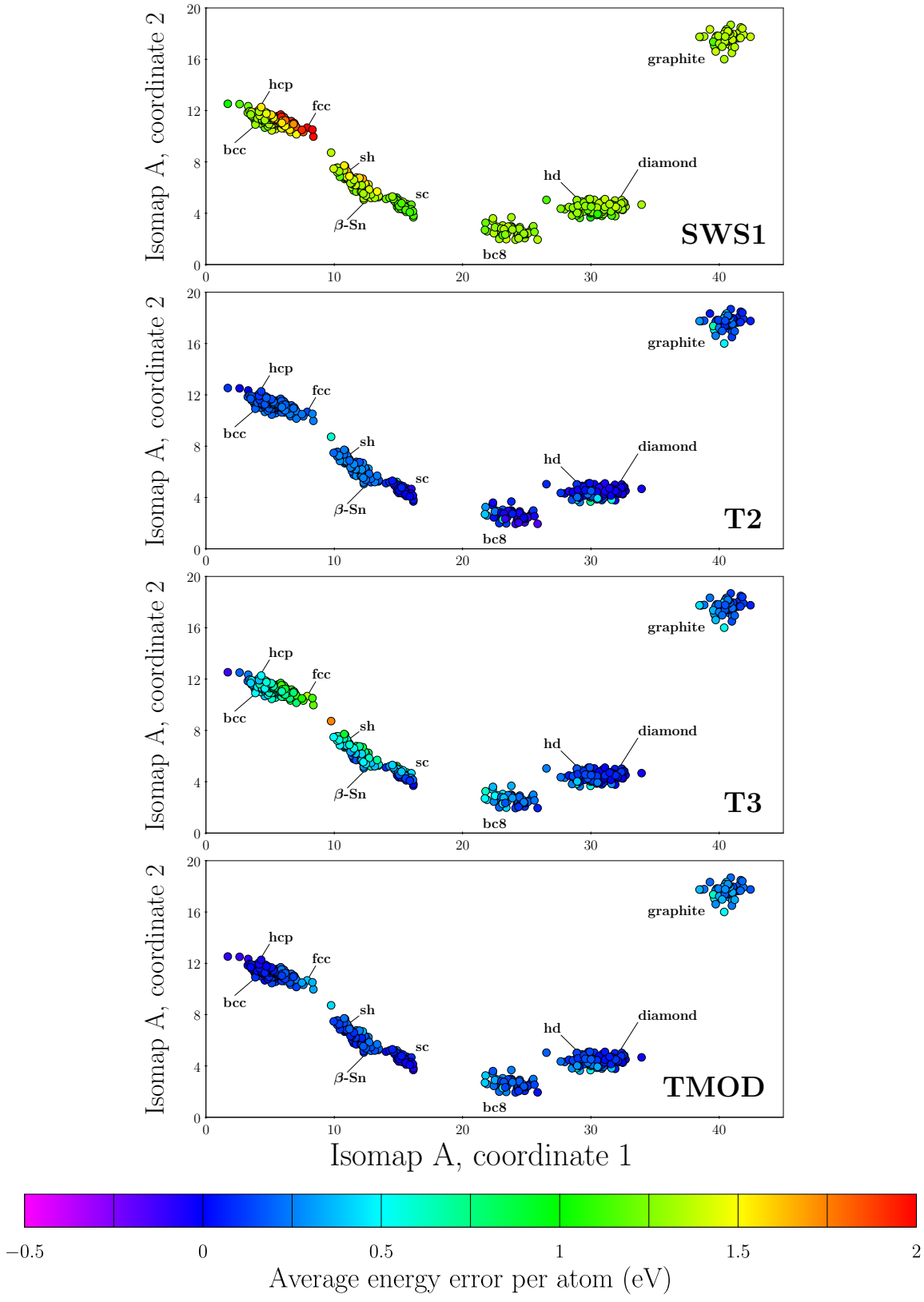


Figure 6.19: Average energy error per atom of the SWS1, T2, T3, and TMOD potentials displayed over the Isomap A coordinates.

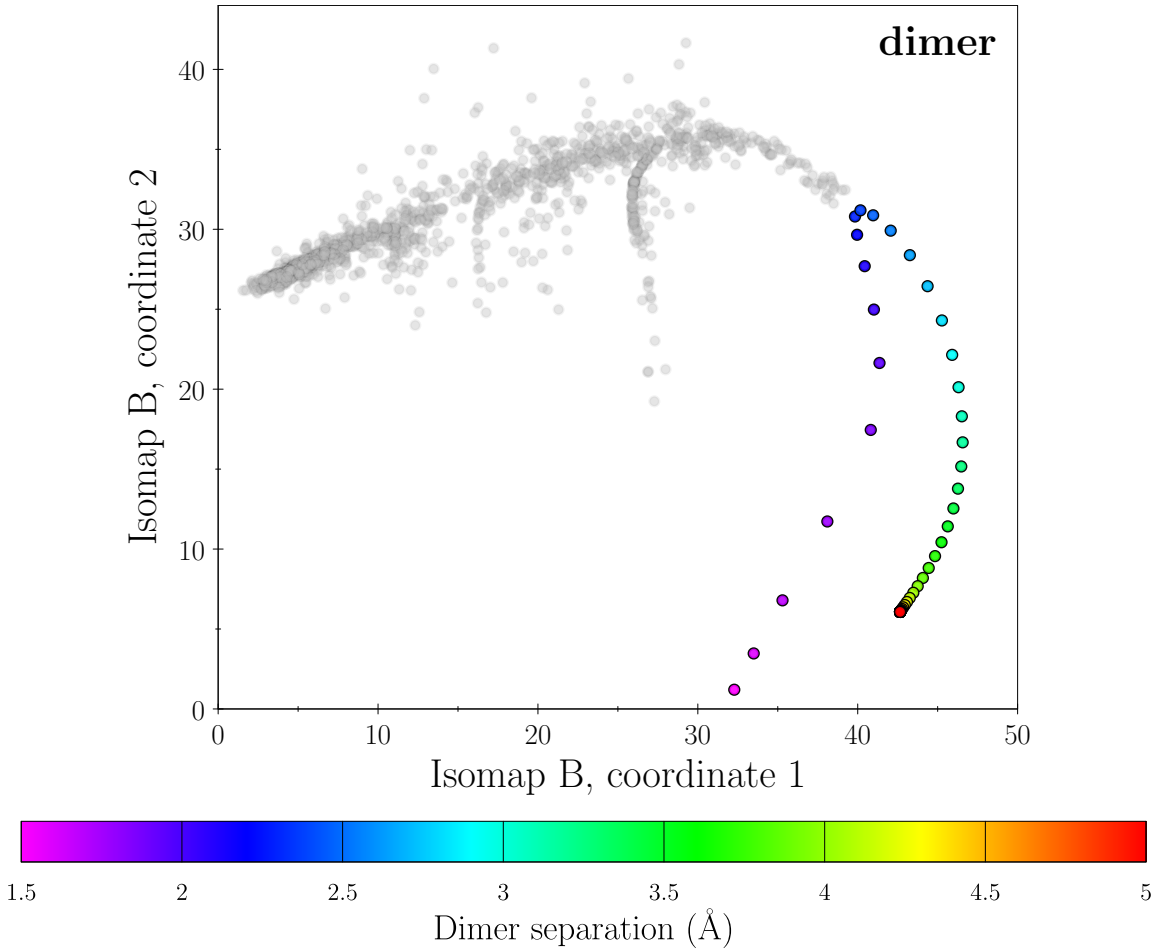


Figure 6.20: Isomap B embedding of the MDS coordinates of the cluster configurations highlighting the dimer configurations, which are shaded according to bond length.

Having examined the bulk MDS coordinates, we turn to those of the cluster configurations. Because these coordinates appear to approximately fall on a two-dimensional manifold in Figures 6.12 – 6.14, we perform a second isomap procedure, 'Isomap B', similar to what was done with the bulk configuration MDS coordinates, only now we exclude all configurations other than the clusters. Figure 6.20 highlights the positions of the dimer configurations over Isomap B, which show a smooth transition from dimers which are only narrowly separated to those which have a separation which meets or exceeds the SOAP cutoff used (5 Å). Dimers which have separations of 2.35 Å (approximately equal to the first nearest-neighbor bond distance found in the ideal diamond configuration of the training set) fall closest to the primary cloud of cluster configurations. One feature of the cluster MDS coordinates which is not entirely faithfully represented in Isomap B is the fairly large separation of the two prongs formed by the dimer configurations, which flare apart from one another in Figures 6.12 – 6.14.

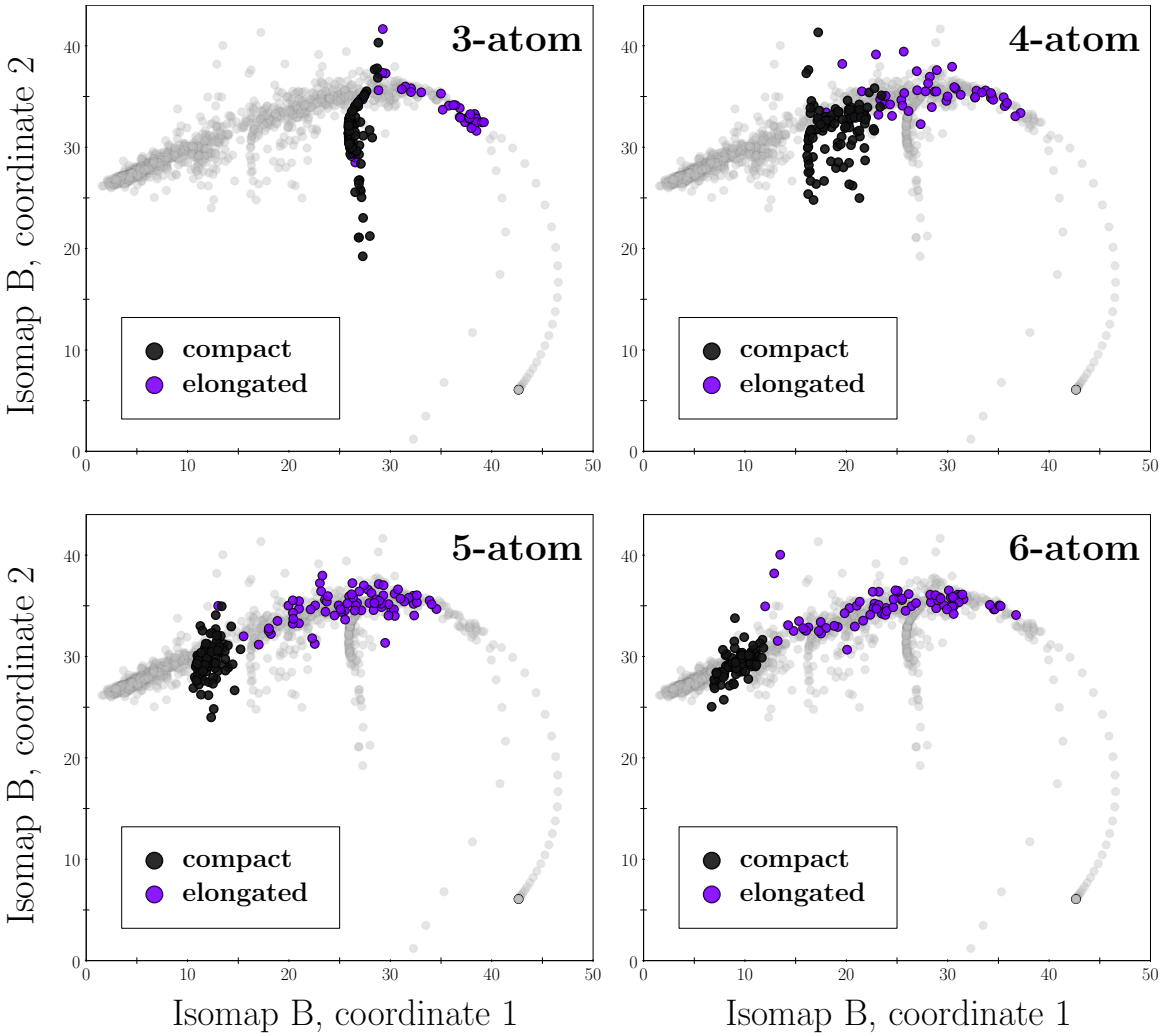


Figure 6.21: Isomap B embeddings of the MDS coordinates of the cluster configurations, highlighting those clusters which contain three, four, five, and six atoms. In each subfigure, the elongated and compact clusters are shaded differently.

Aside from the dimer configurations, the relative positions of the remaining clusters in Isomap B are shown in Figures 6.21 and 6.22, where subfigures illustrate the distribution of the cluster configurations according to how many atoms they contain. The first clear pattern borne out in these plots is that, for a given atom count, the elongated clusters always lay to the right of the compact clusters. This is in keeping with the second trend we observe: the groups of compact clusters fall farther to the left as the number of atoms present in the cluster increases. However, note that the elongated clusters still stretch broadly across the spine of the data for every atom count. Of course, one aspect of these plots which obscures comparison is the fact there are only 150 three- and four-atom clusters in the training set compared to 175 five- and six-atom clusters and 200 clusters corresponding to each of the remaining atom counts. Nonetheless, it is reasonable to posit that, since the elongated clus-

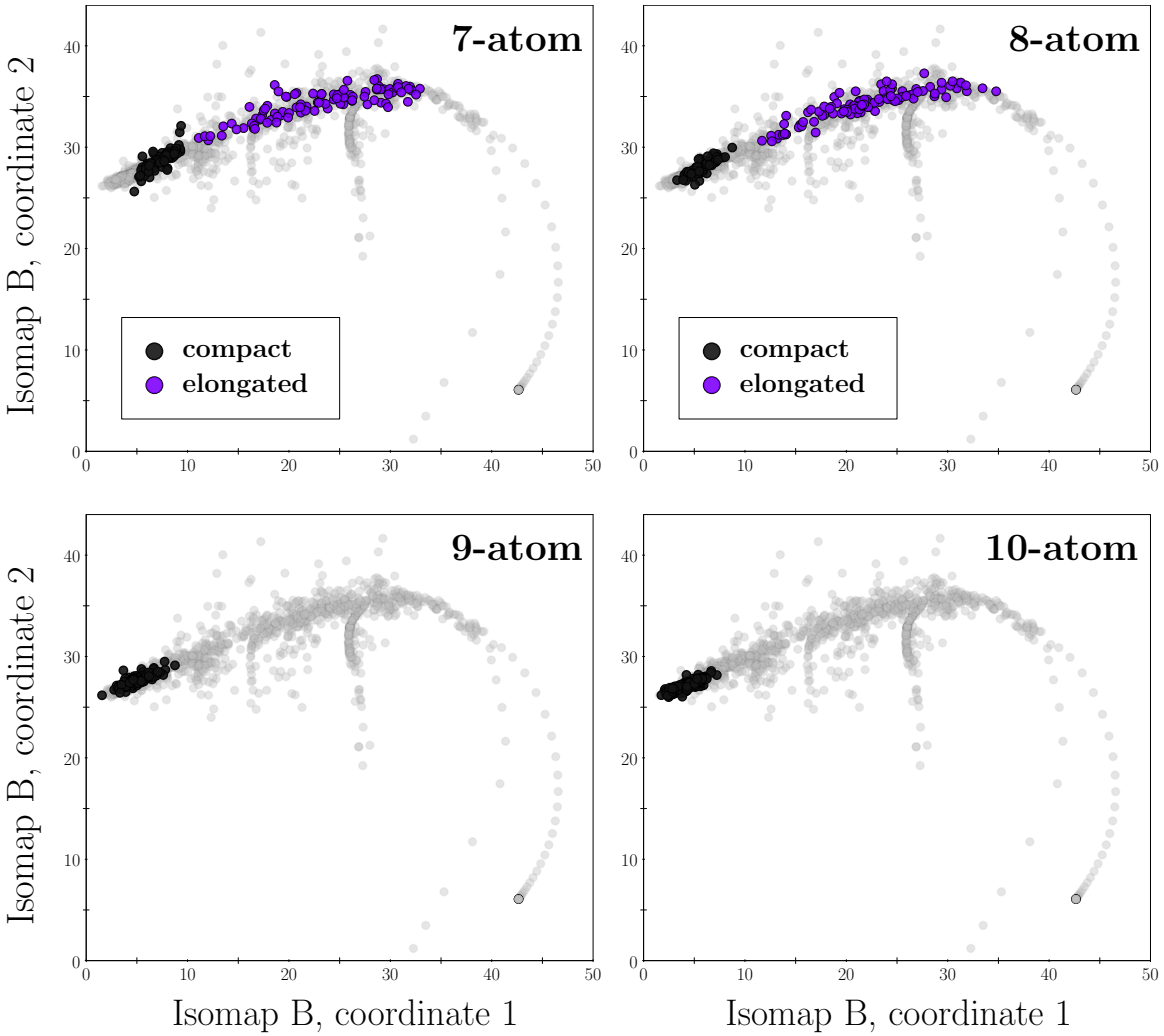


Figure 6.22: Isomap B embeddings of the MDS coordinates of the cluster configurations, highlighting those clusters which contain seven, eight, nine, and ten atoms. In each subfigure, the elongated and compact clusters are shaded differently (recall that there were no elongated nine- or ten-atom clusters).

ters have a lower average coordination than the compact clusters, the underlying quantity which best explains the arrangement of points on the isomap is the coordination itself. This may be confirmed by once again plotting the average coordination function $\overline{\Gamma}_C$ defined in (6.13) for the cluster configurations, the result of which is in Figure 6.23. As with the bulk structures, there is a smooth variation in coordination ranging from values near unity at the dimers in the lower right to values near 8.0 for the nine- and ten-atom clusters on the far left.

Our final matter of inquiry of the cluster MDS embedding is the distribution of the average first-principles energy and average energy error over Isomap B. In Figure 6.24, we see

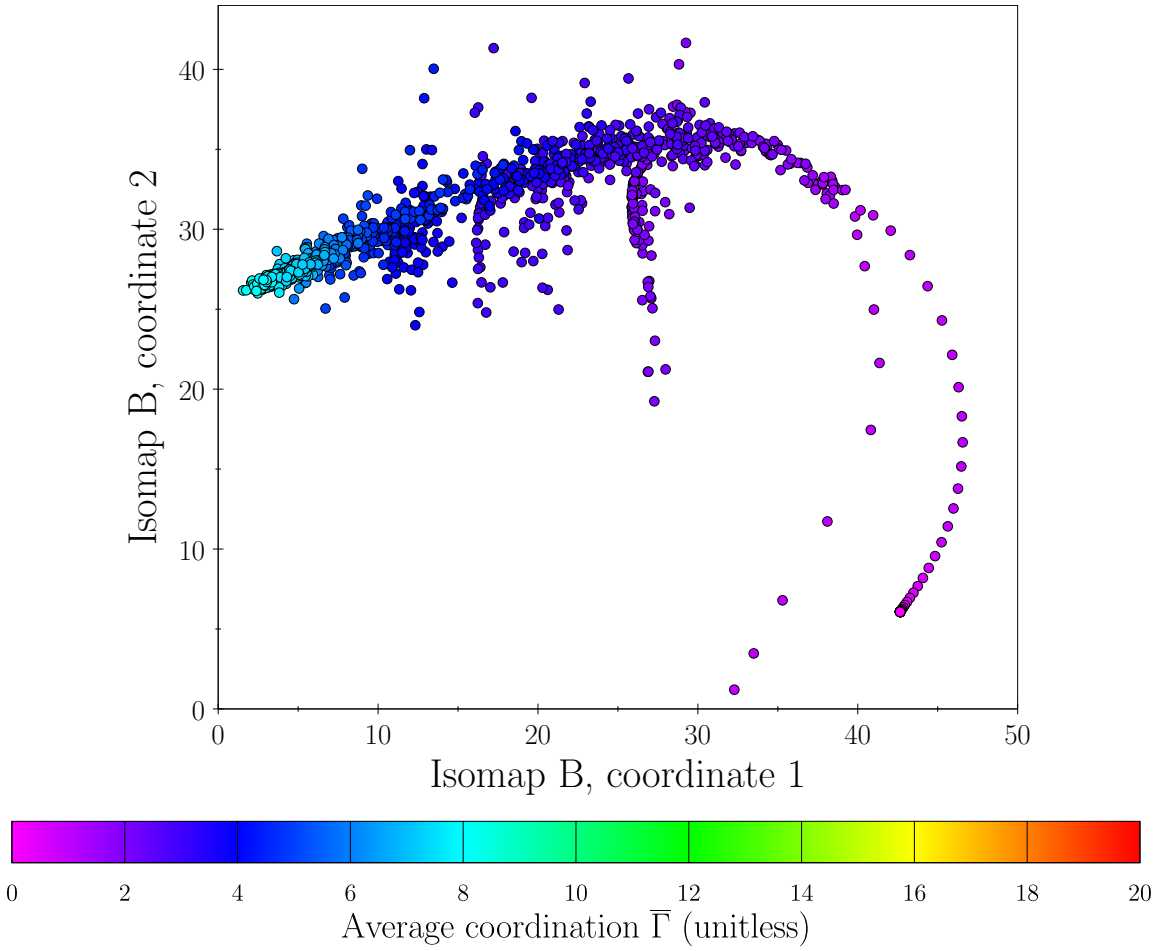


Figure 6.23: Isomap B embedding of the MDS coordinates of the cluster configurations, shaded according to the values of the average coordination of the atoms in each configuration as given in (6.13).

the former, where it appears that highest average energies occur for the atoms in dimers, particularly those of very high or very low bond length. As one moves along the vertical direction in the figure, the energy decreases to intermediate values for compact clusters containing three or more atoms. Finally, on the upper portion of the cloud, where configurations possess an average coordination of $\overline{\Gamma} \approx 4$ (corresponding to tetravalent bonding), the average energy decreases to its minimum value of approximately -3 eV/atom. The average energy error of the EPs of our study are shown in Figures 6.25 and 6.26. Because the primary cutoff of all of the EPs is smaller than the descriptor cutoff of 5.0 Å used, they all assign a zero energy to the dimers which possess bond lengths near this value, just as in the first-principles results. On the other hand, the nine- and ten-atom clusters corresponding to the far left portion of the cloud introduce significant average energy errors for most of the EPs compared to first-principles, reaching as high as 4 eV/atom and beyond in the case

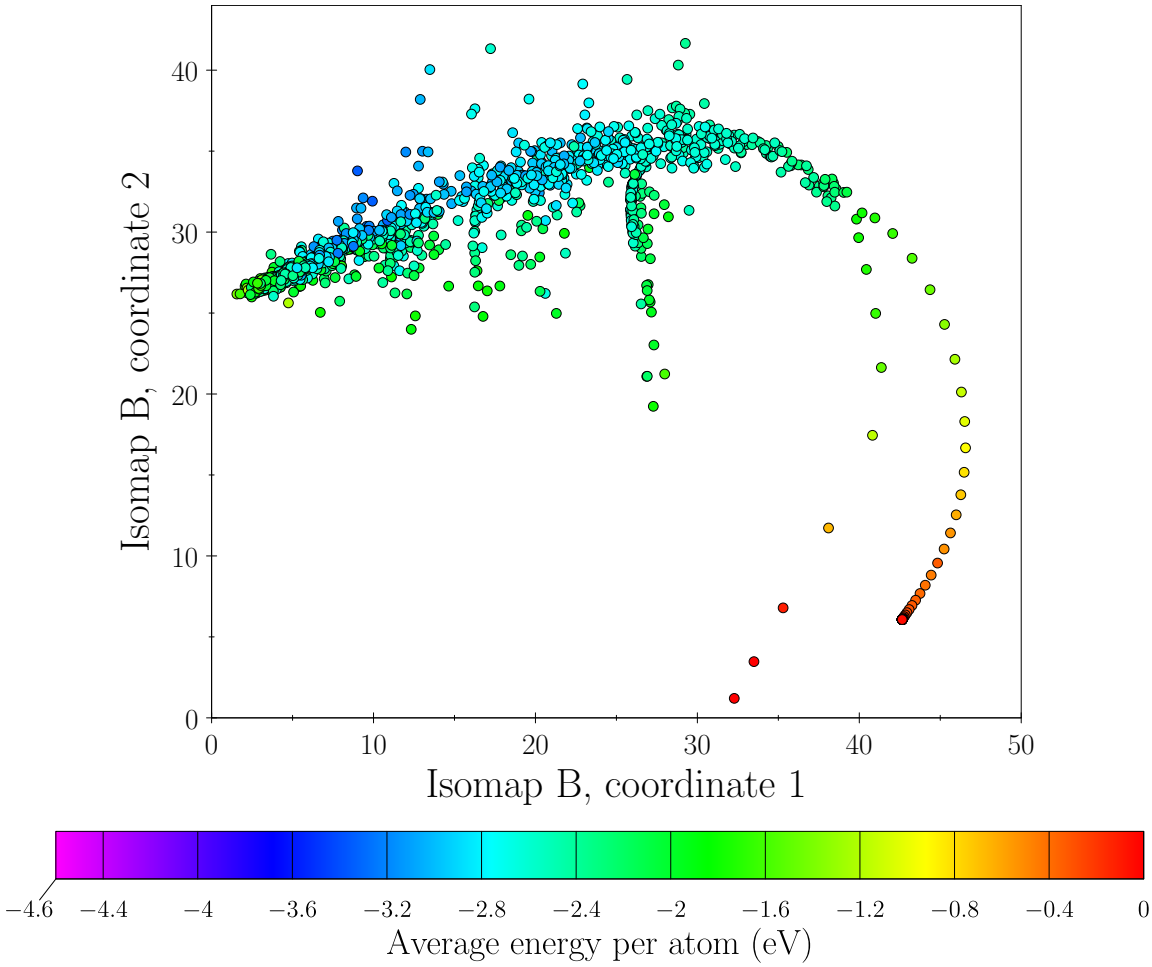


Figure 6.24: Isomap B embedding of the MDS coordinates of the cluster configurations, shaded according to the average first-principles energy per atom in each configuration.

of SW and SWS1. The T3 potential also produces inadequate energy predictions for the majority of the clusters in the left portion of the data. After T3, the EA, EDIP, and TMOD all provide moderate overall accuracy but encounter difficulty for points on the underside of the spine, which are primarily compact clusters containing three to five atoms, as can be seen from Figure 6.21. The highest accuracy is achieved by the LSA and T2 potentials, with T2 providing the best overall performance, as in the case of the bulk structures. Although this is in agreement in previous works [261, 287], it is somewhat unintuitive because of the relative importance of angular contributions in small clusters, where covalent bonds (including π bonds for the very smallest clusters) are formed rather than metallic bonds. However, we note that the angular insensitivity of T2 does lead it to underestimate the average energies of some of the compact four- and five-atom clusters, whereas none of the other EPs underestimate the average energies of any of the clusters with three or more atoms.

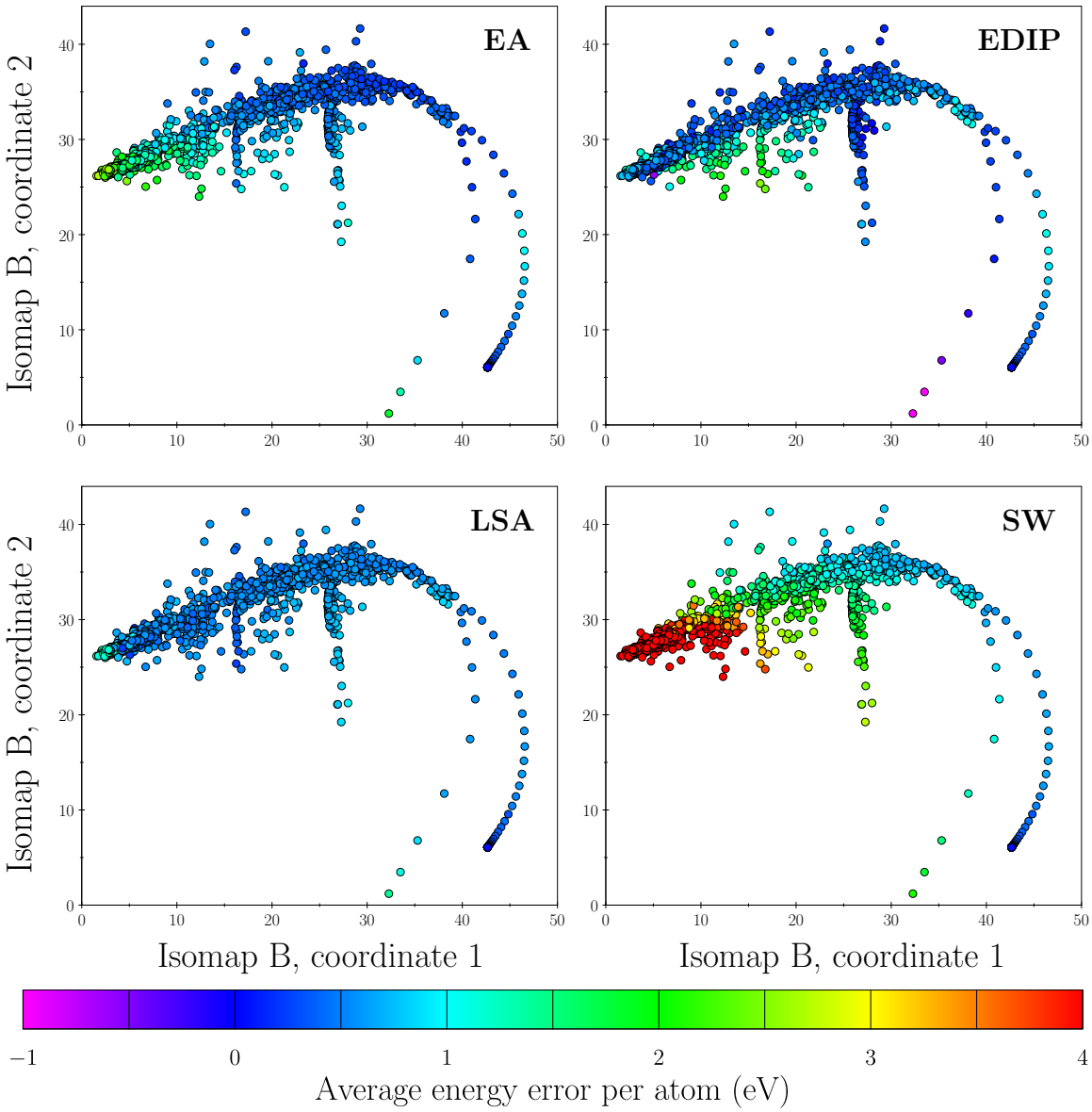


Figure 6.25: Average energy error per atom of the EA, EDIP, LSA, and SW potentials displayed over Isomap B.

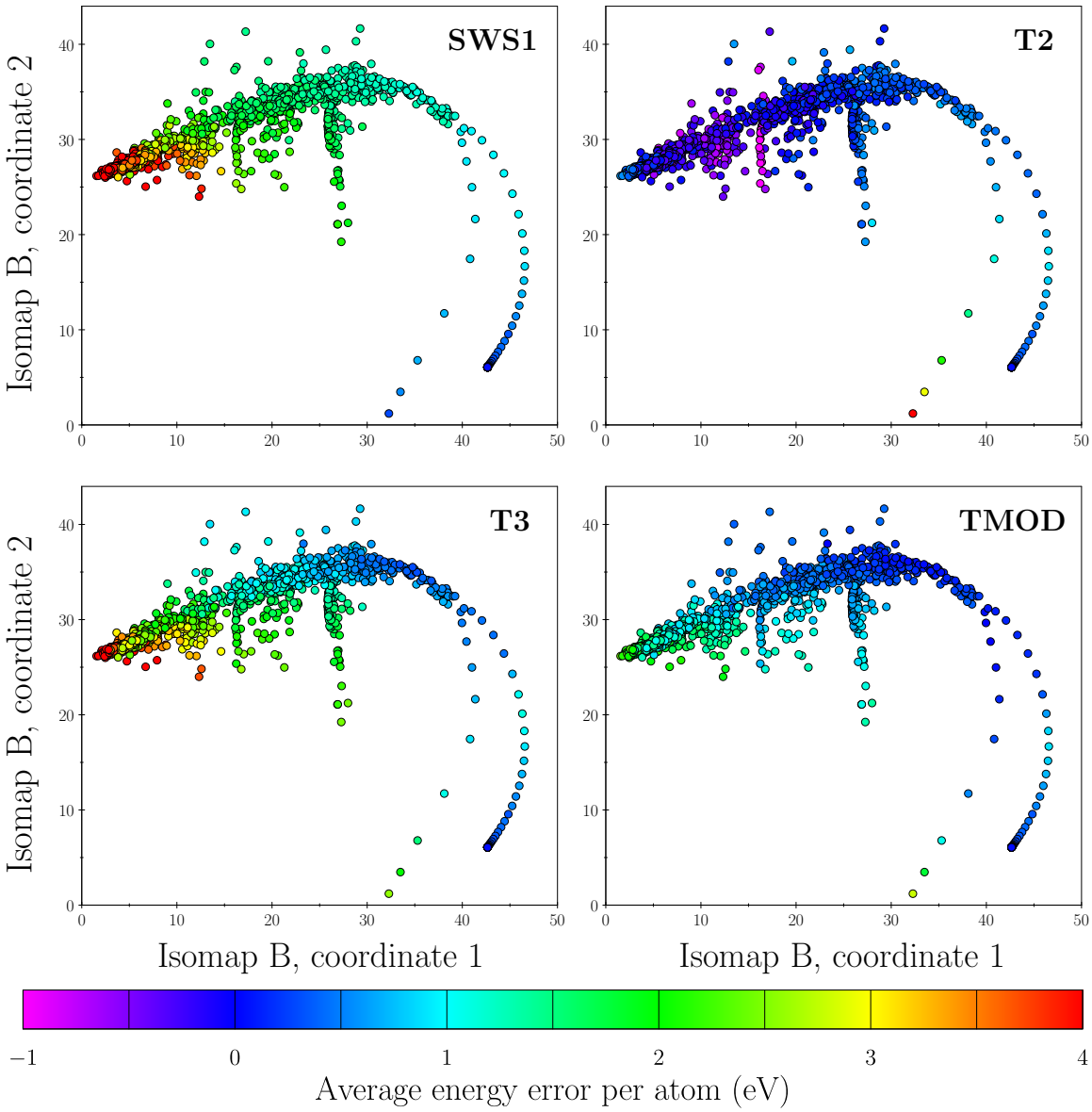


Figure 6.26: Average energy error per atom of the SWS1, T2, T3, and TMOD potentials displayed over Isomap B.

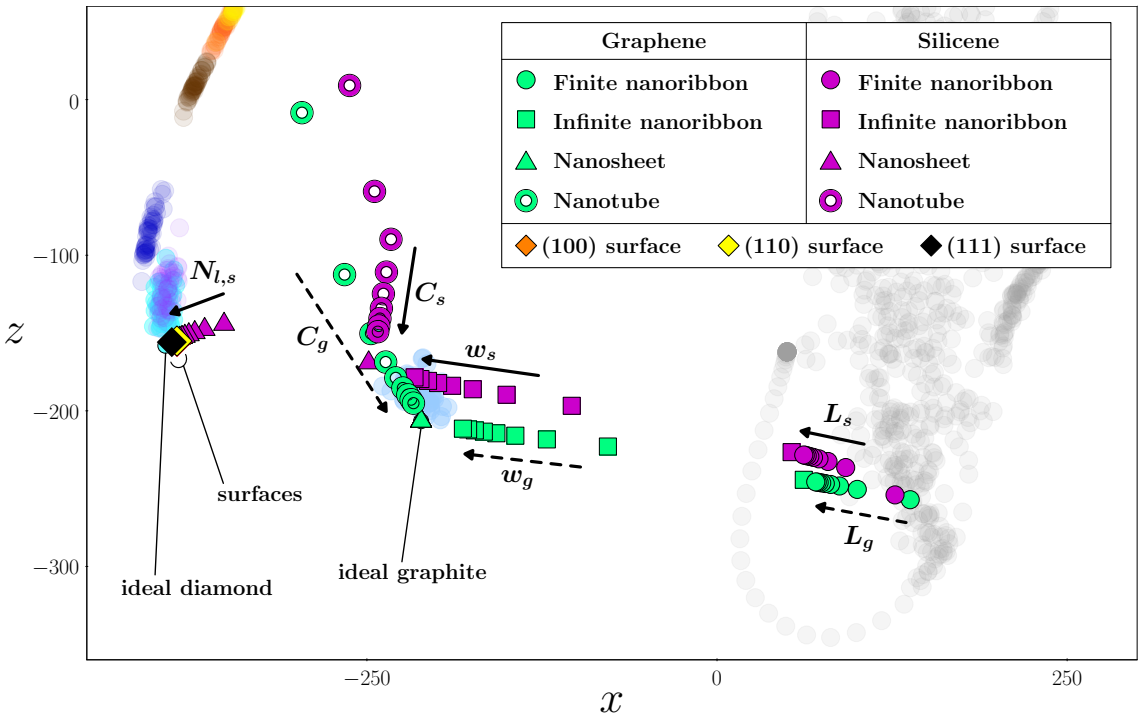


Figure 6.27: Projection of the MDS coordinates of the nanostructure configurations into the xz plane.

We now turn our attention to the nanostructures of our study, which are related to one another by simple limiting processes, as mentioned in Section 6.3.3. By taking incrementally larger lengths of the finite nanoribbons, the structure tends toward the infinite nanoribbons which have width $w = 1$. Increasing the width $w \to \infty$ yields a monolayer nanosheet (which has infinite planar extent, by our definition). Finally, recalling that the nanosheets of our training set are stacked in AB sequence for graphene and ABC sequence for silicene, increasing the number of graphene layers $N_{l,g}$ recovers the ideal graphite structure while increasing the number of silicene layers $N_{l,s}$ recovers the ideal diamond structure. As an independent limiting process, we also note that the environment of a given atom in the nanotube structures becomes increasingly akin to the corresponding atomic environment in a monolayer nanosheet of the same type as the nanotube diameter C is increased. The MDS coordinates of the nanostructures are depicted in Figure 6.27; because these coordinates do not clearly lie on a two-dimensional manifold in Figures 6.12 – 6.14, we have not applied the isomap algorithm, instead opting to project them into the xz plane.[19] As shown by the arrows in the figure (dashed arrows indicate the limits of the structural parameters of the graphene-based nanostructures, while the solid arrows indicate those of the silicene nanostructures), each of the limiting processes outlined above is respected by the MDS embed-

[19]The coordinate axes x and z referred to here are those depicted in Figures 6.12 – 6.14.

ding. However, one curious feature of Figure 6.27 is that all of the graphene nanosheets lay atop the position of ideal graphite, which is a consequence of the normalization carried out in (6.11). The configuration covariance values $\mathcal{K}(\mathcal{C}_{N_{l,g}}, \mathcal{C}_{\text{graphite}})$ themselves between the graphene nanosheets and the ideal graphite increase as the number of layers $N_{l,g}$ increases due to the increasing number of atoms present, according to the construction of the configuration covariance from the individual environment covariances in (6.3). However, the configuration autocovariance $\mathcal{K}(\mathcal{C}_{N_{l,g}}, \mathcal{C}_{N_{l,g}})$ also increases in measure commensurate with the number of atoms present. This is because with the descriptor cutoff of 5.0 Å (Table 6.2), all of the atomic environments in the various graphene nanosheet stacks are identical to the unique type of atomic environment found in graphite, owing to the fact that the interlayer spacing of the stacks (for $N_{l,g} \geq 2$) is approximately 10.6 Å, leaving each atom in these structures with only neighbors which reside in the same layer as them.

Working from the assumption that the total energy is composed of local atomic contributions, one would expect that the average energy per atom of the nanostructures would follow limiting processes which accord with those mentioned above. In Figure 6.28, we plot the average first-principles energy per atom of the finite-length nanoribbons (top left), infinite-length nanoribbons (top right), nanosheets (bottom left), and nanotubes (bottom right) for both silicene and graphene. Consistent with the structural relations discussed above, we see monotonic convergence of the corresponding average energies toward the average energies of the postulated limiting structures (shown in dashed lines), albeit complete convergence does not occur for the particular set of nanostructure configurations in our training set due to their finite size. Because the interlayer spacing of the ideal graphite structure exceeds the cutoff radius used in the descriptor of our study, as mentioned in the preceding paragraph, the average energy of the nanosheet stacks of graphene remain constant and equal to the cohesive energy of the ideal graphite lattice predicted by first principles. In Figures 6.29 and 6.30, we also depict the average energy errors of each EP for the same structures, which demonstrate the same convergence process, athough it is not monotonic in some cases. Once more, the average energy errors of the EPs for the graphene nanosheets are constant and equal to their corresponding ideal graphite values because the cutoff radii of the EPs are smaller than the interlayer spacing. Judging from these figures, the EA and T2 potentials offer the highest accuracy for the nanostructures, with the EDIP and T3 potentials yielding the next best performance. While all of the potentials aside from SWS1 provide high accuracy for the stacks of six or more silicene layers (tending to the ideal diamond structure), it is surprising to note that SWS1 is inaccurate by comparison even for the silicene nanoribbons and nanosheets.

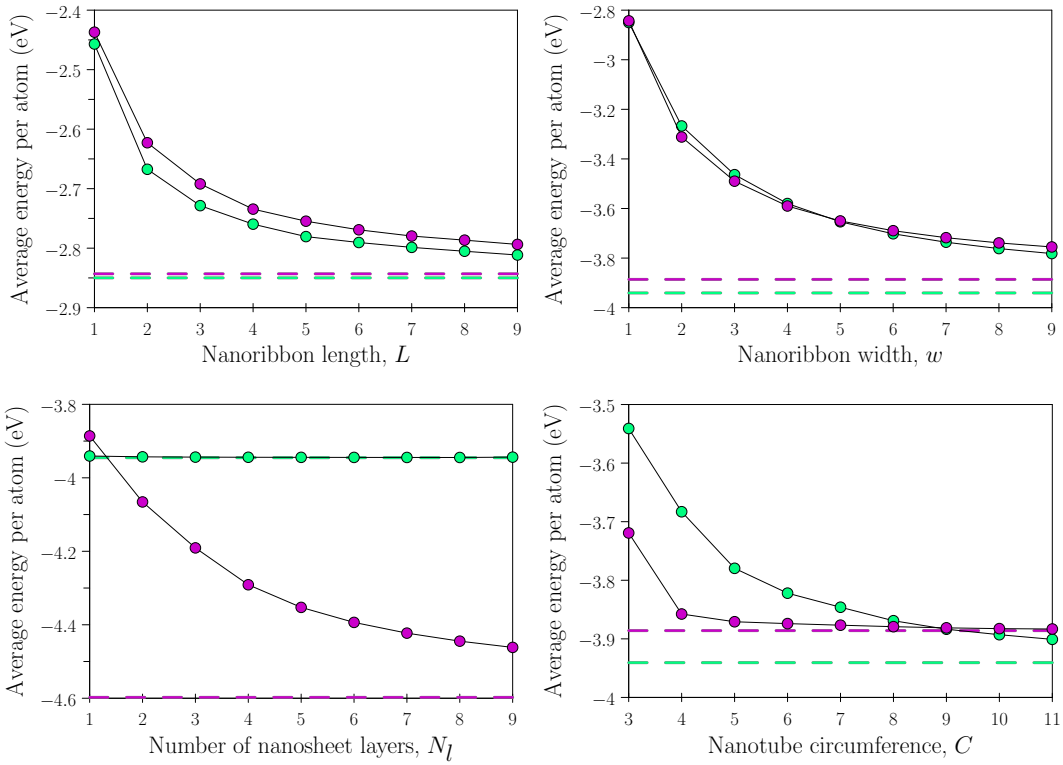


Figure 6.28: Average energy per atom of the nanostructure configurations as a function of their structural parameters: (Top Left) Finite-length nanoribbons. The dashed green line indicates the average energy of an infinite-length graphene nanoribbon of width $w_g = 1$, while the dashed pink line indicates the average energy of an infinite-length silicene nanoribbon of width $w_s = 1$. (Top Right) Infinite-length nanoribbons. The dashed green line corresponds to a single layer of graphene, i.e. a graphene nanosheet stack with $N_{l,g} = 1$, while the dashed pink line corresponds to a single layer of silicene. (Bottom Left) Nanosheet stacks. The dashed green line is the average energy of the ideal graphite structure, while the dashed pink line is that of the ideal diamond structure. (Bottom Right) Nanotubes. The dashed lines are the same as in the top right figure.

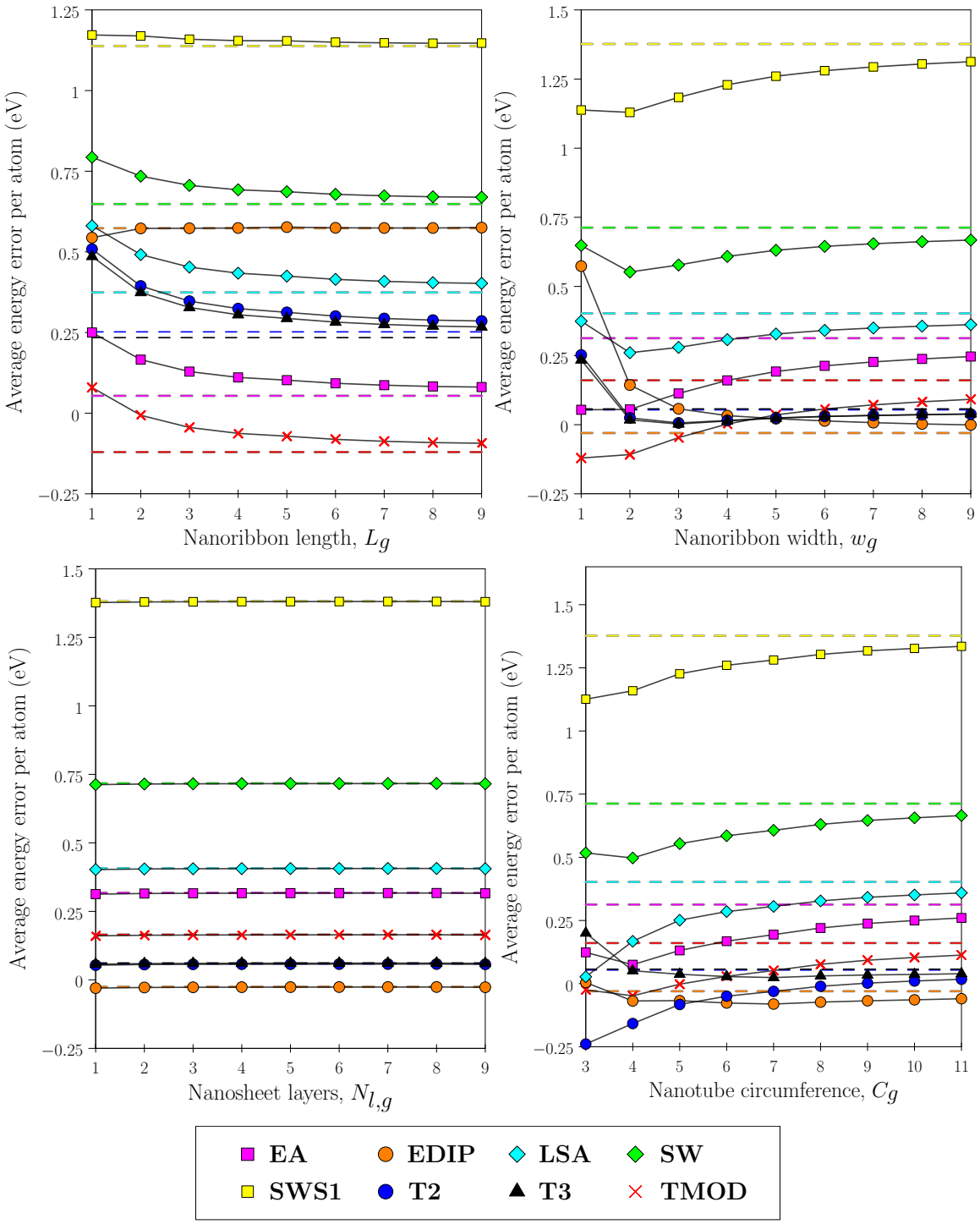


Figure 6.29: Average energy error per atom of the EPs for the finite-length nanoribbons (top left), infinite-length nanoribbons (top right), nanosheets (bottom left), and nanotubes (bottom right) derived from graphene as a function of the structural parameters of each. In the top left plot, the dashed lines correspond to the average energy error of each EP for the infinite-length graphene nanoribbon. In the top right and bottom right frames, they indicate the average energy errors for a monolayer graphene nanosheet. In the bottom left, they represent the average energy errors for the ideal graphite bulk structure.

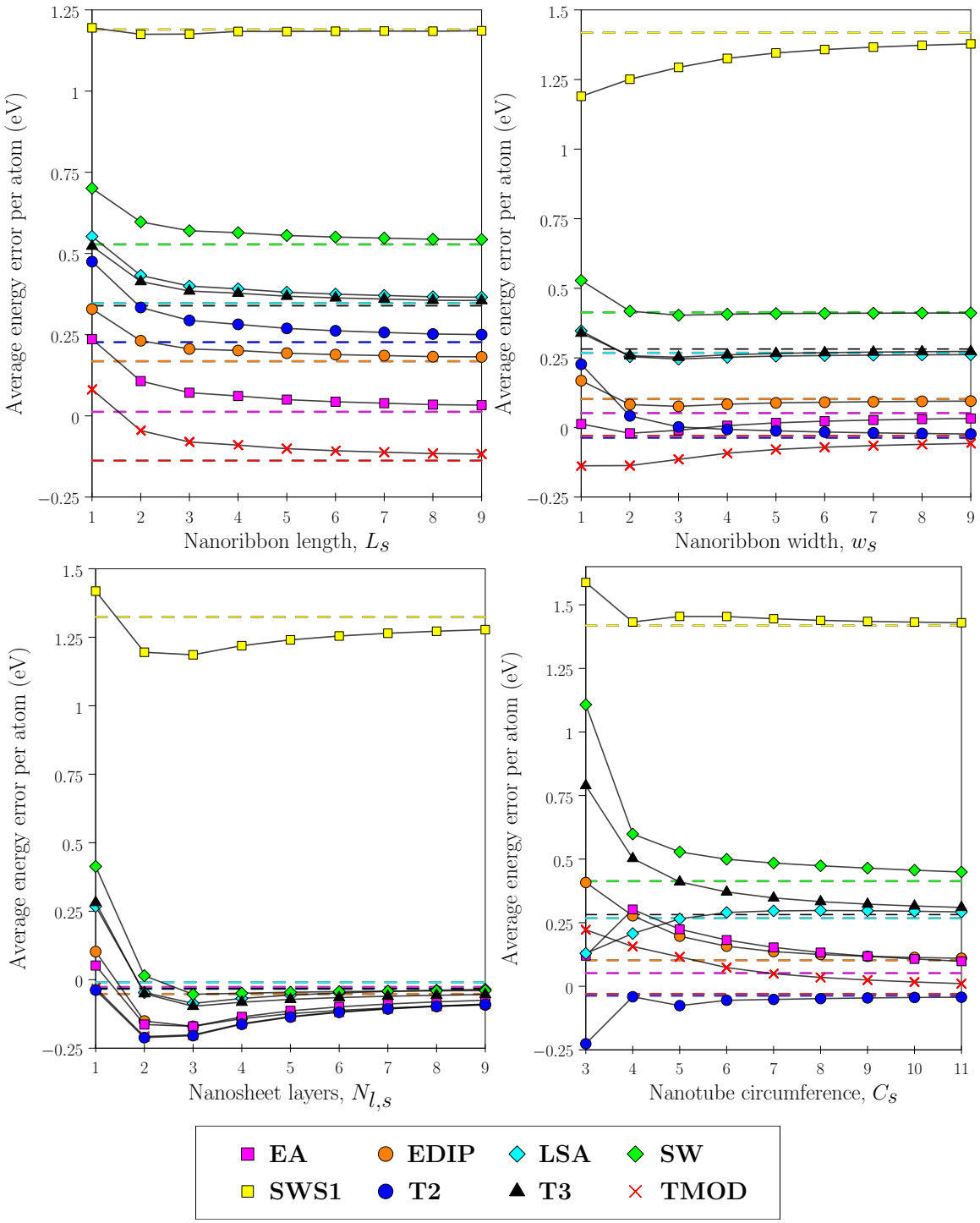


Figure 6.30: Average energy error per atom of the EPs for the finite-length nanoribbons (top left), infinite-length nanoribbons (top right), nanosheets (bottom left), and nanotubes (bottom right) derived from silicene as a function of the structural parameters of each. In the top left plot, the dashed lines correspond to the average energy error of each EP for the infinite-length silicene nanoribbon. In the top right and bottom right frames, they indicate the average energy errors for a monolayer silicene nanosheet. In the bottom left, they represent the average energy errors for the ideal diamond bulk structure.

6.4.2 Atomic environments

In the previous section, we analyzed the GPR predictive equations (6.1) and (6.2) at the topmost level by inspecting the matrix $K(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}})$ of covariances defined between each pair of atomic configurations in the training set, as well as the vector of target observations $\mathbf{y}$ which, by turns, played the role of the first-principles total energies of the training set configurations and the corresponding total energy errors of the eight EPs listed in Table 6.1. By defining dissimilarities between entire configurations of atoms from their covariances used in the GPR predictive equations through (6.11), a three-dimensional embedding in a metric space was obtained using the MDS algorithm. Upon inspecting the coordinates of each configuration in this space, it was found that they were organized into separate groups consisting of the bulk configurations and the cluster configurations, with the nanostructures residing between the two. Using the isomap algorithm to further project the bulk and cluster groups onto two-dimensional spaces, it was revealed that the points were laid out in such a way as to produce smooth gradients in average coordination (as defined in (6.13)), average first-principles energy per atom, and average energy error per atom amongst the eight three-body EPs of Table 6.1.

Although $K(\boldsymbol{\mathcal{C}}, \boldsymbol{\mathcal{C}})$, $K(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}})$, and $K(\boldsymbol{\mathcal{C}}_*, \boldsymbol{\mathcal{C}}_*)$ are ultimately the covariance matrices that are actually used in (6.1) and (6.2), they are still constructed from the covariance matrices between individual environments according to (6.3). As discussed in the introduction of this chapter, the regression performed by (6.1) and (6.2) using the total energy or the total energy error of an EP is accordingly based on an underlying set of atomic energies or atomic energy errors, respectively, which are assigned to each environment in the training and test sets. This is made apparent by substituting (6.3) into these equations to obtain

$$\begin{aligned} \overline{\mathbf{f}}(\boldsymbol{\mathcal{C}}_*) =& L_* K(\mathbf{X}_*, \mathbf{X}) L^T \left[LK(\mathbf{X}, \mathbf{X}) L^T + \sigma_n^2 \mathbf{I} \right]^{-1} \mathbf{y}, && (6.14) \\ \operatorname{cov}(\mathbf{f}(\boldsymbol{\mathcal{C}}_*)) =& L_* K(\mathbf{X}_*, \mathbf{X}_*) L_*^T \\ & - L_* K(\mathbf{X}_*, \mathbf{X}) L^T \left[LK(\mathbf{X}, \mathbf{X}) L^T + \sigma_n^2 \mathbf{I} \right]^{-1} LK(\mathbf{X}, \mathbf{X}_*) L_*^T, && (6.15) \end{aligned}$$

where we identify the energies (or energy errors) assigned to each atomic environment in the test set by writing (6.14) in the form

$$\overline{\mathbf{f}}(\boldsymbol{\mathcal{C}}_*) = L_* \mathbf{y}_*^{\text{atomic}}, \tag{6.16}$$

and we have defined the $N_{\mathcal{A}_*} \times 1$ vector of atomic energies (or energy errors) as

$$\mathbf{y}_*^{\text{atomic}} \triangleq K(\mathbf{X}_*, \mathbf{X}) L^T \left[LK(\mathbf{X}, \mathbf{X}) L^T + \sigma_n^2 \mathbf{I} \right]^{-1} \mathbf{y}. \tag{6.17}$$

When the vector $\mathbf{y}$ of target observations consists of the first-principles total energies of the atomic configurations of the training set, we will denote[20] the atomic energies which comprise the vector $\mathbf{y}_*^{\text{atomic}}$ by $\varepsilon_\alpha^{\text{DFT}}$, while if regression is performed on the total energy errors of an EP, we denote the corresponding atomic energy errors $\check{\varepsilon}_\alpha^{\text{EP}}$. Because L_* is a $N_{\mathcal{C}_*} \times N_{\mathcal{A}_*}$ matrix whose ij-th element is equal to 1 if the atom in the test set corresponding to column j belongs to test configuration i (and zero otherwise), the form of (6.16) intuitively acts to render a total energy or energy error for each configuration in the test set $\boldsymbol{\mathcal{C}}_*$ by summing the individual energies or energy errors assigned to each of its environments. The covariance matrix corresponding to the individual atomic environments is given by

$$\text{cov}(\mathbf{y}_*^{\text{atomic}}) = K(\mathbf{X}_*, \mathbf{X}_*) - K(\mathbf{X}_*, \mathbf{X}) L^T \left[LK(\mathbf{X}, \mathbf{X}) L^T + \sigma_n^2 \mathbf{I} \right]^{-1} LK(\mathbf{X}, \mathbf{X}_*), \quad (6.18)$$

and has diagonal entries which indicate the variance associated with the atomic energies and energy errors inferred by the regression, i.e. the i-th entry along the diagonal contains the variance for environment x_i.

We now repeat the same procedure of the previous section to visualize the internal components of the GPR regression equations, only this time we draw our focus to individual atomic environments rather than entire atomic configurations. Once again, we take the test set to correspond to the training set so that the environments of the two are identical ($\mathbf{X}_* = \mathbf{X}$), and thus the only unique covariance matrix in each of the above equations can be written as $K(\mathbf{X}, \mathbf{X})$. Recall that the ij-th element of $K(\mathbf{X}, \mathbf{X})$ is given by the direct evaluation of the SOAP kernel between environments x_i and x_j:

$$[K(\mathbf{X}, \mathbf{X})]_{ij} = k_{\text{SOAP}}(x_i, x_j). \quad (6.19)$$

For our training set, which contains 2110 configurations comprising a total of 15721 atomic environments (see Tables 6.3 and 6.4), the matrix $K(\mathbf{X}, \mathbf{X})$ is thus symmetric and of dimensions 15721$\times$15721. Since the SOAP kernel is already normalized in (4.97), a dissimilarity measure between atomic environments x_i and x_j may be defined by simply taking

$$\Xi(x_i, x_j) \triangleq 1 - K(x_i, x_j).$$

Because of the prohibitive computational expense required to perform MDS with the entire set of 15721(15721-1)/2 = 123,567,060 dissimilarities of the training set, only ten initial

[20] Using the subscript α is an abuse of notation in the sense that we have hitherto used this symbol to serve as an index which runs over the atoms in a single configuration, whereas here it runs over all of the atoms in the entire test set.

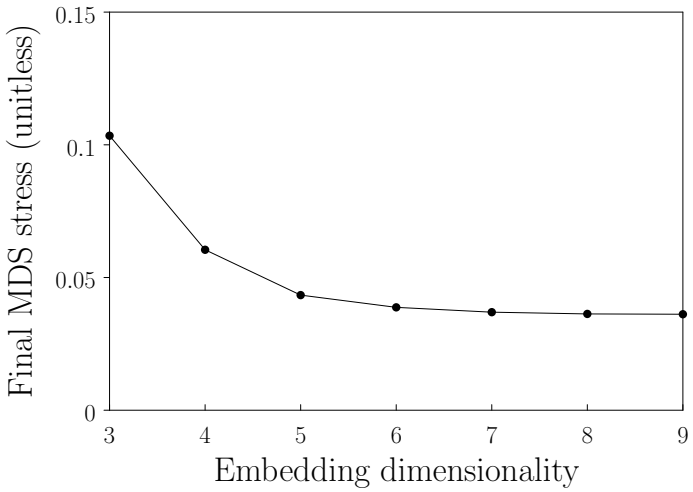


Figure 6.31: Final Kruskal stress of the MDS procedure for all of the individual atomic environments of the entire training set as a function of MDS embedding dimensionality.

configuration seeds were used for each embedding dimensionality, which ranged from three to nine. As in the case of the atomic configurations, convergence of the MDS algorithm was defined according to successive stresses differing by less than 1e-6 or until 15000 iterations were reached. The lowest resulting stresses for each dimensionality are shown in Figure 6.31, where a monotonic decay with dimensionality is observed. Although the Kruskal stress accounts for the fact that a much larger data set was used in these calculations, the stress for dimensionality three (over 0.1) is approximately 45% higher than for the embedding of the atomic configurations (≈ 0.07), which is due in part to incomplete convergence of the algorithm. Furthermore, there is a marked decrease in stress in going to four or more dimensions. However, because the primary goal of our study is to gain an understanding of the atomic energies and energy errors defined by the GPR equations through visualization, we proceed with the result of the three-dimensional embedding.

Despite the higher stress of the embedding, the relative positions of the MDS coordinates of the atomic environments shown in Figures 6.32 – 6.34 more or less mirror those of the atomic configurations in Figures 6.12 – 6.14. However, as opposed to before, there is greater overlap between the coordinates of the atomic environments, and we have accordingly identified four primary groups into which the data fall. Group I, shown in blue, consists of atomic environments belonging to the metallic bulk configurations (sc, β-Sn, sh, bcc, fcc, hcp). Group II (red) consists of environments from the diamond, hexagonal diamond, and bc8 configurations. Group III (green) includes only environments from the graphite structures. Group IV (grey) includes all of the dimer and cluster environments. Finally, the environments belonging to the nanostructures are superposed transparently and

fall in between groups II, III, and IV; spheres, cubes, tetrahedra, and cylinders represent finite-length nanoribbons, infinite-length nanoribbons, nanosheets, and nanotubes, respectively. Green and pink coloring is used to represent graphene- and silicene-based nanostructures, respectively, while octahedra represent environments belonging to the diamond surface configurations.

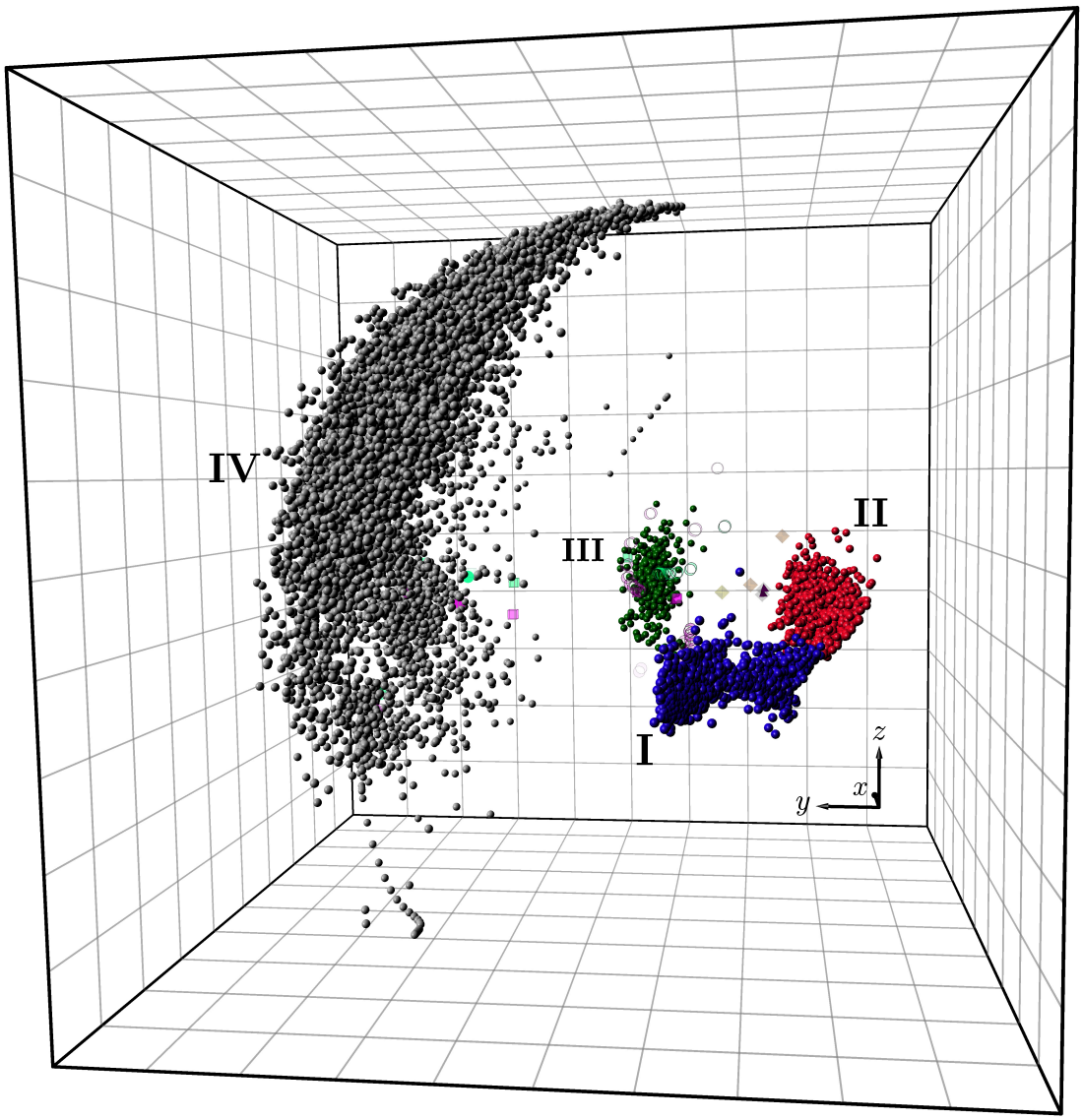


Figure 6.32: MDS embedding of the individual atomic environments of the entire training set. Figures 6.33 and 6.34 depict alternative perspectives of the same data.

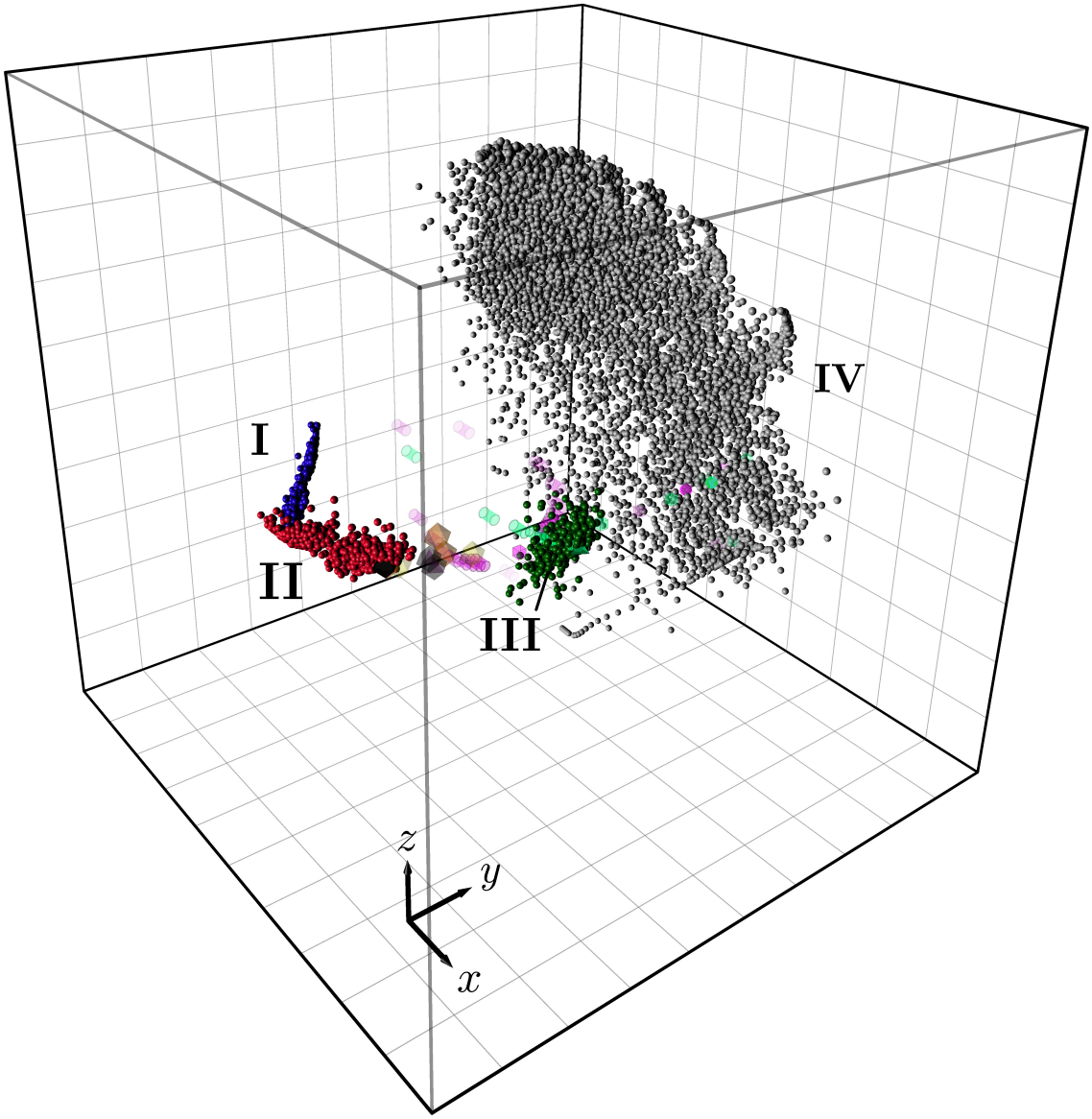


Figure 6.33: MDS embedding of the individual atomic environments of the entire training set. Figures 6.32 and 6.34 depict alternative perspectives of the same data.

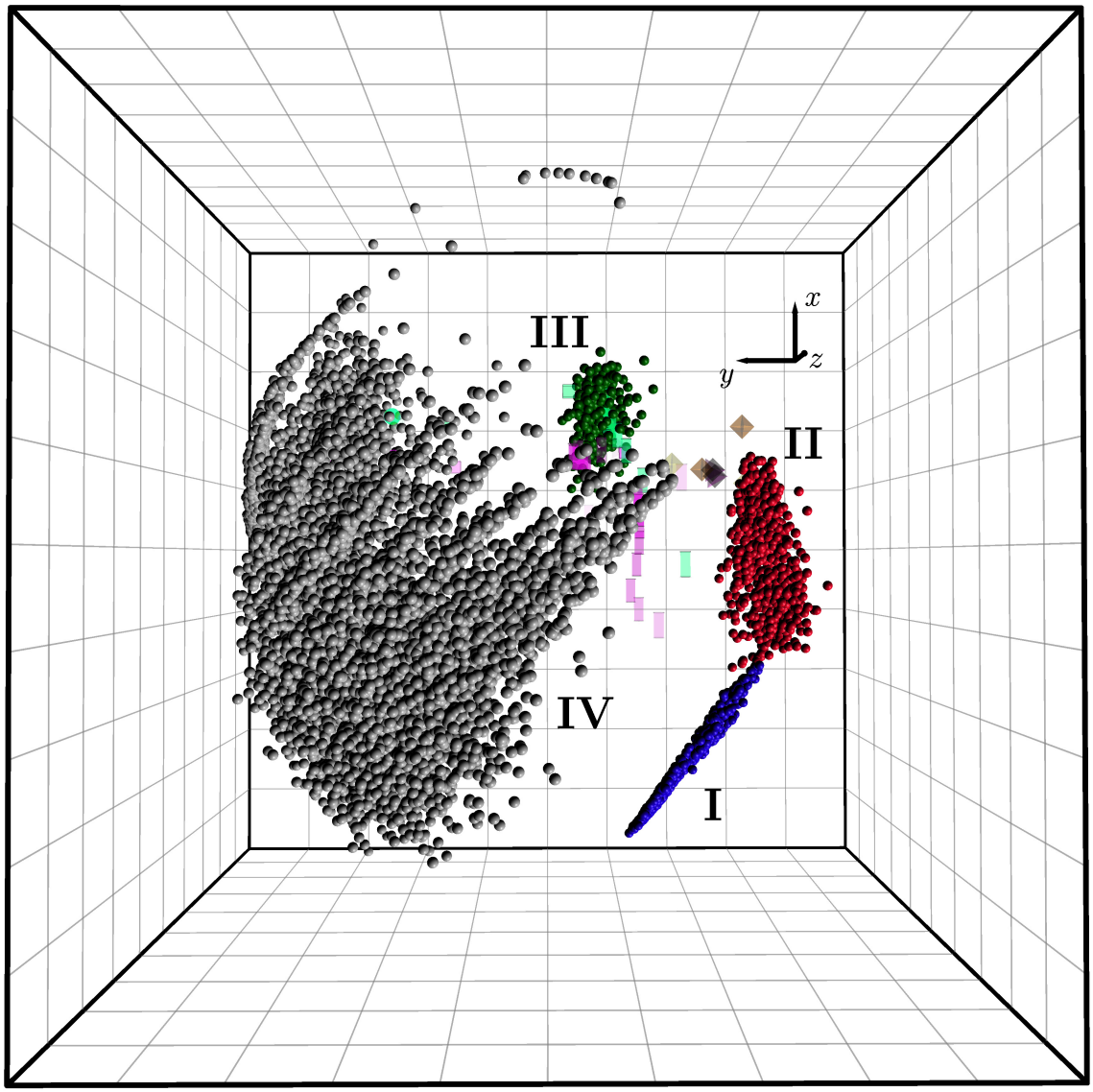


Figure 6.34: MDS embedding of the individual atomic environments of the entire training set. Figures 6.32 and 6.33 depict alternative perspectives of the same data.

We begin our analysis by applying the isomap algorithm to each of the bulk structure groups (I-III), the results of which are presented in Figures 6.35 – 6.37. As with the atomic configurations in Figure 6.15, the atomic environments belonging to the bcc, fcc, and hcp structures lay nearby one another in the isomap I coordinates. Similarly, the β-Sn, sc, and sh environments appear on top of each other in isomap I, and the diamond and hexagonal diamond environments are interspersed in isomap II. In order to understand the specific arrangement of the MDS coordinates, we plot the coordination function $\Gamma(\alpha)$ of (6.12), which we recall is constructed using the same cutoff function as was used in the SOAP kernel (which defines the similarity between atomic environments and, thus, their embedding coordinates). The resulting pattern, shown in Figure 6.38, is nearly identical to the plot of the average coordination $\overline{\Gamma}_{\mathcal{C}}$ shown in Figure 6.16, with the lack of variation between the different environments being a consequence of the relatively large 5.0 Å cutoff used.

The atomic energies $\varepsilon_{\alpha}^{\mathrm{DFT}}$ inferred by the regression for each environment from the first-principles total energies of the training set configurations are shown over isomaps I-III in Figure 6.39. Comparison with Figure 6.17 indicates that many of the atomic energies are not far removed from the average energies of the corresponding configurations. As intuition would suggest, the environments present in the perturbed diamond and hexagonal diamond structures are assigned the lowest energies, which are close to the cohesive energy computed for the ideal diamond lattice (-4.6 eV/atom). Moreover, we note that no atomic energies inferred for any of the environments in the training set, including those of the cluster and nanostructure configurations, fell below this value. Similar to the average energies, the bc8 environments are assigned the next lowest energies, followed by a portion of the β-Sn, sh, and sc environments in group I. However, there are discrepancies with the average energies which indicate that the atomic energies determined by the regression are, indeed, sensitive to the geometric details of each environment. First, one may observe that elevated energies are inferred for some of the environments in each of isomaps I, II, and III. Figure 6.40 reveals that many of these environments contain a neighbor which is within 1.85 Å of the corresponding target atom, and it should thus be expected that they are assigned relatively high energies in order to account for Pauli repulsion. However, even excluding these environments, there are still variations in the atomic energies of the graphite environments in isomap III which deviate significantly from those of the average energies of graphite shown in Figure 6.17. For example, there are atomic energies assigned to some of these environments which are significantly lower than the cohesive energy of the ideal graphite structure (-3.94 eV/atom).

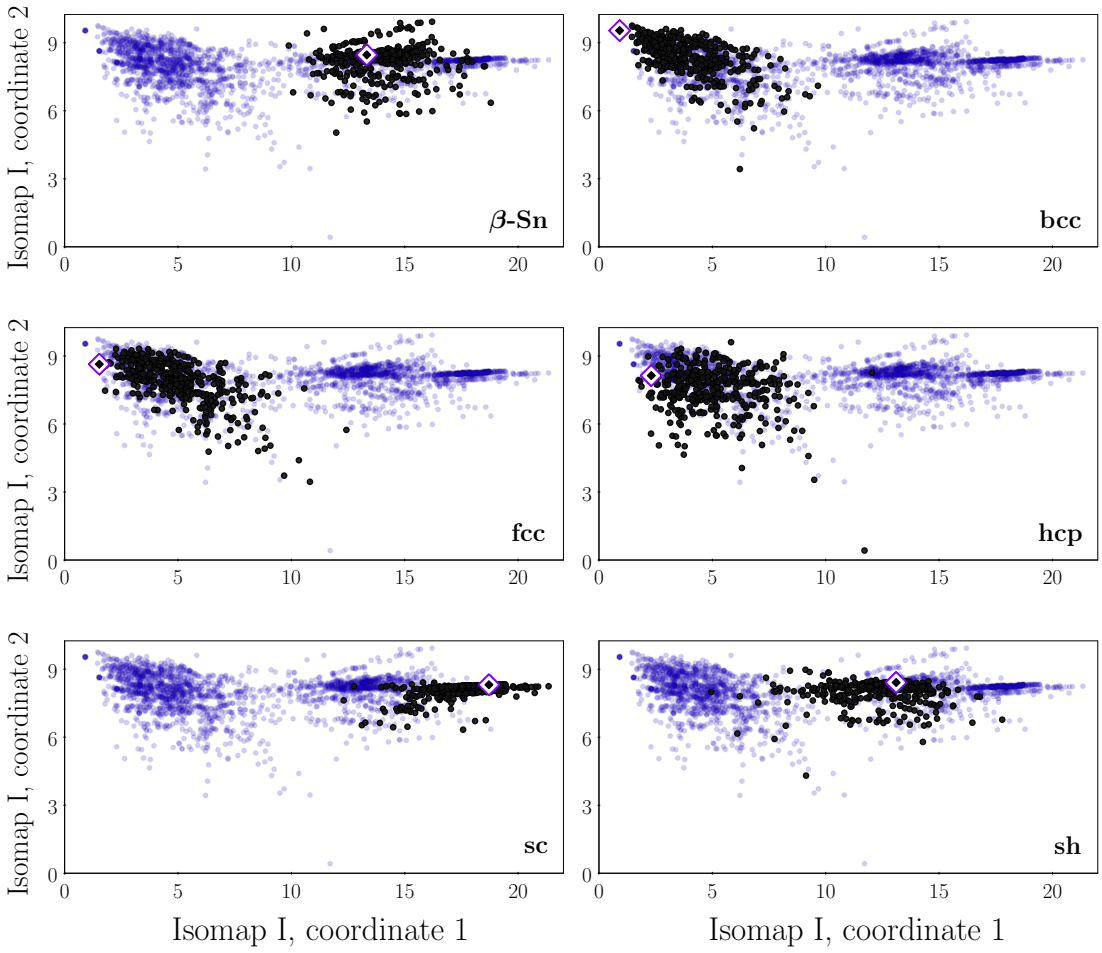


Figure 6.35: Isomap of group I illustrating the location of the atomic environments belonging to the perturbed β-Sn, bcc, fcc, hcp, sc, and sh configurations. In each subfigure, the diamond-shaped symbol indicates the location of the corresponding ideal bulk environment.

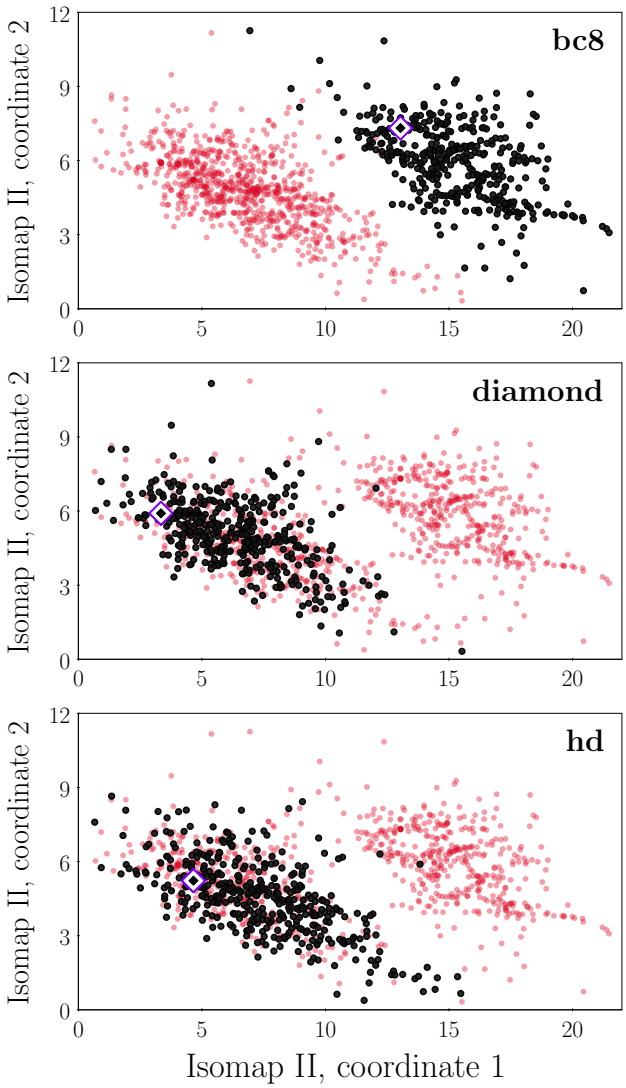


Figure 6.36: Isomap of group II illustrating the locations of the atomic environments belonging to the perturbed bc8, diamond, and hexagonal diamond configurations.

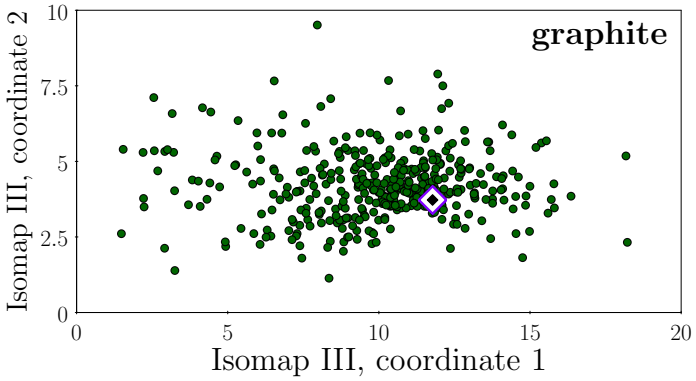


Figure 6.37: Isomap of group III illustrating the locations of the atomic environments belonging to the perturbed graphite configurations.

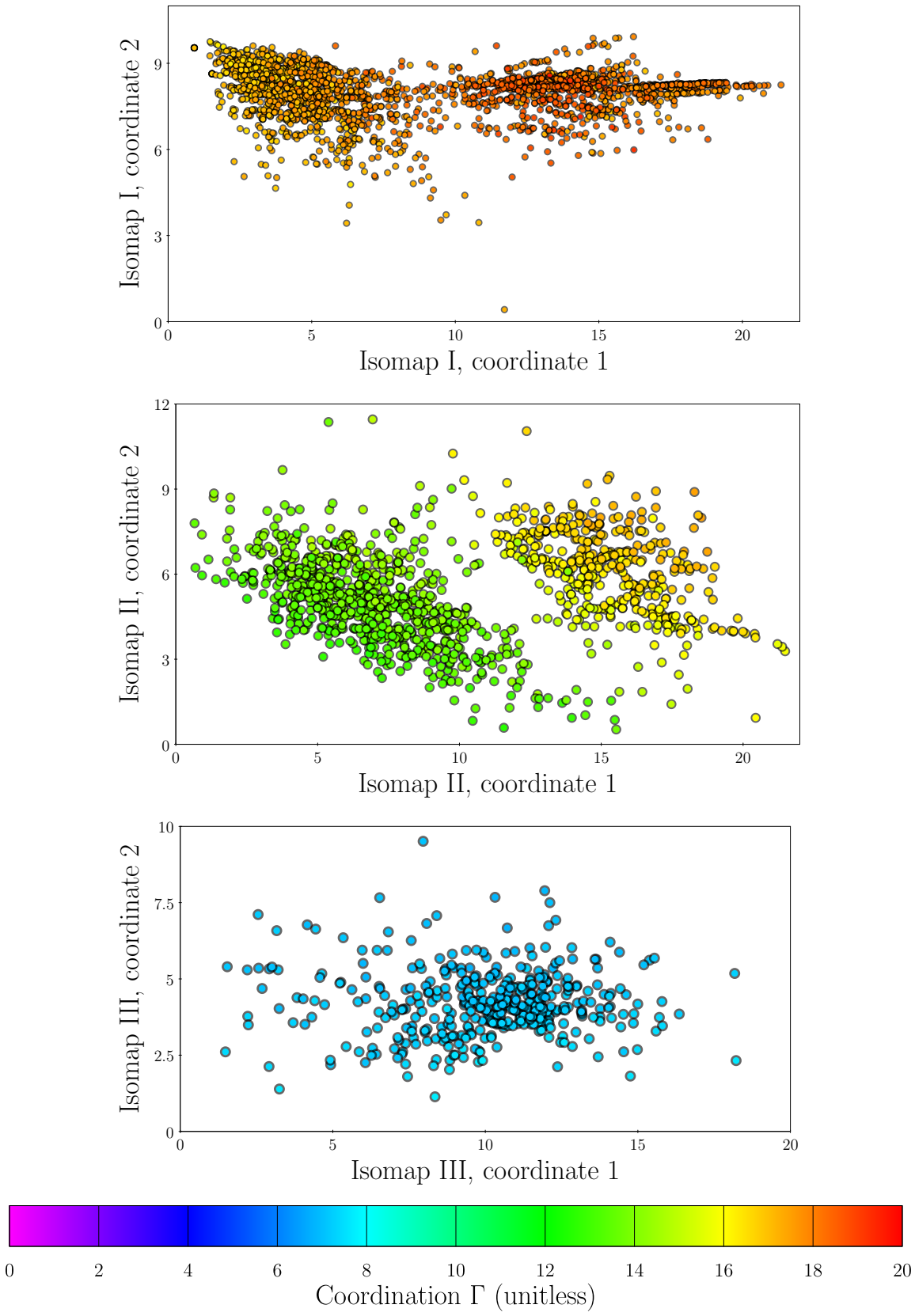


Figure 6.38: Coordination Γ defined in (6.12) of the bulk environments, shown over isomaps I-III.

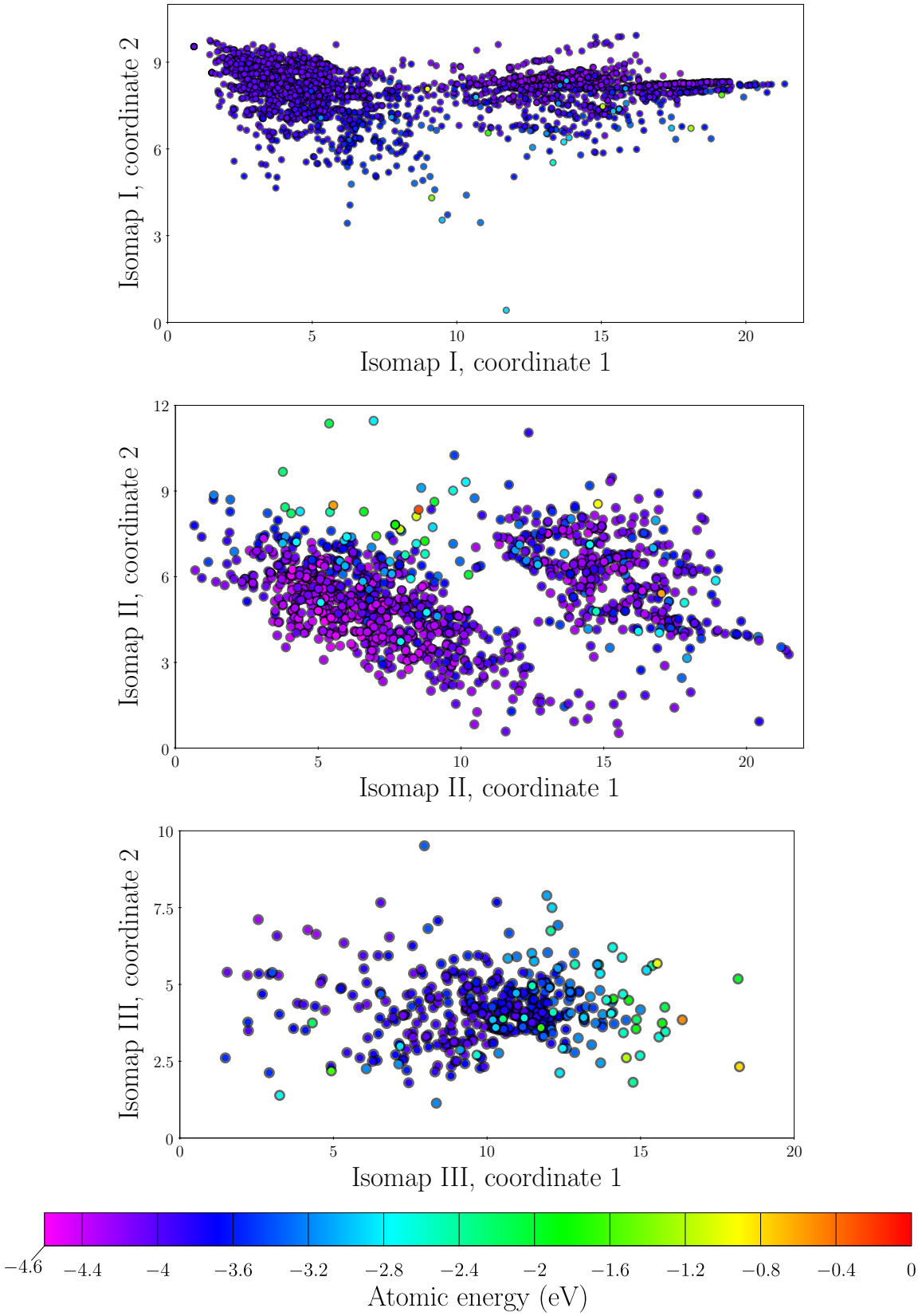


Figure 6.39: Atomic energies $\varepsilon_\alpha^{\mathrm{DFT}}$ learned for the bulk environments from the first-principles total enregies, shown over isomaps I-III.

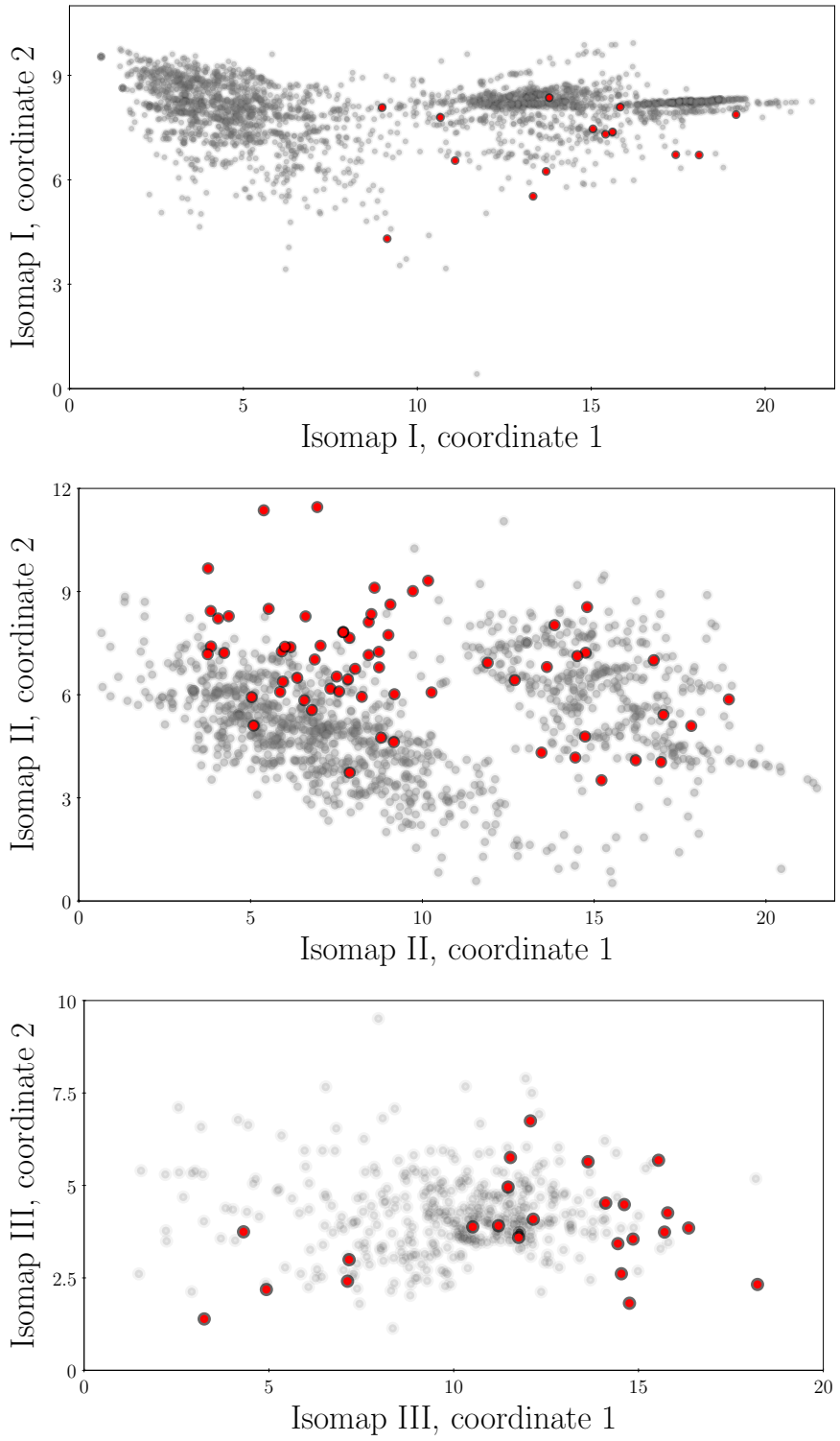


Figure 6.40: Atomic environments of the bulk structures shown over isomaps I-III. Points which are colored red indicate the environments of atoms which possess a neighbor within a distance of 1.85 Å (no more than one such neighbor was present in any of the configurations shown).

As the partition of the first-principles total energies into atomic energies $\varepsilon_\alpha^{\mathrm{DFT}}$ performed by the regression lacks any rigorous theoretical derivation, it cannot necessarily be shown to be unique or have physical significance. However, it is useful for the purpose of heuristic modeling in the sense that, were an EP constructed which defined atomic energies matching the quantities $\varepsilon_\alpha^{\mathrm{DFT}}$, it would be certain to reproduce the total energies of the training set configurations to the same accuracy as the first-principles calculations themselves.[21] In the remainder of this chapter, we proceed by making the assumption that the atomic energies $\varepsilon_\alpha^{\mathrm{DFT}}$ are unique and that an EP must, therefore, necessarily reproduce them in order to provide accurate total energies for the training set configurations. Bearing this requirement in mind, we next examine the atomic energy errors determined by RATE for the environments of the training set.

Because (6.14) is linear in the vector $\mathbf{y}$ of target observations, using the total energy errors $\mathcal{V}_C^\Delta$ of an EP to carry out the corresponding regression is equivalent to performing regression on the first-principles total energies and the EP total energies separately and subtracting the results of the former from those of the latter. Just as the application of GPR we have described is capable of learning a set of atomic energies from first-principles total energies, it can likewise infer a set of atomic energies for an EP from its total energies. Thus, the atomic energy errors produced by RATE may be regarded as indicating the level of agreement between the atomic energies inferred by the regression from the first-principles total energies and those which it learns from the EP total energies. However, just as in the case of the first-principles energies, there is a problem of non-uniqueness which arises. As we have emphasized, because EPs are phrased in terms of local energies (whether they be associated with atoms or bonds), there are infinitely many partitions of the energy which may be ascribed to individual atoms which will be consistent with the total energies of the EP for all possible atomic configurations. Thus, in the general application of RATE, the total energy errors of an EP can be accurately interpolated to test configurations directly using GPR, but the significance of the corresponding atomic energy errors underlying the process is unclear. If the interpretation of this data is to be useful in drawing conclusions about the accuracy of an arbitrary EP for configurations which contain particular types of atomic environments, it is imperative that an understanding of the atomic energies inferred by the regression for a given EP be gained. From a practical standpoint, this requires that observations be made for a large number of EPs and a wide variety of atomic configurations, and

[21] We show in Figure E.1 that with the current choice of SOAP parameters and noise parameter σ_n, the total energies are reproduced with great fidelity. Figure E.2 shows the same for the total energy errors of each EP. This rules out the possibility that the parameters we have used in this study, particularly σ_n, give rise to any numerical instability.

it is to this end that the implementation of this family of methods in the KIM framework becomes advantageous. In Section C.9, we have put forth one specific set of closed-form definitions for (permutationally invariant) atomic energies, chosen to be as simple as possible, which may be associated with the eight EPs considered in this chapter. From these atomic energies, which we denote $\mathcal{V}_\alpha^{\mathrm{EP}}$, we define an alternate measure of atomic energy error for each EP which might be compared with the corresponding quantities $\check{\varepsilon}_\alpha^{\mathrm{EP}}$ from RATE according to

$$\Lambda_\alpha^{\mathrm{EP}} \triangleq \mathcal{V}_\alpha^{\mathrm{EP}} - \varepsilon_\alpha^{\mathrm{DFT}}, \tag{6.20}$$

Plotting the atomic energy errors $\check{\varepsilon}_\alpha^{\mathrm{EP}}$ learned by RATE for each EP over isomap I in the left-hand columns of Figures 6.41 and 6.42 beside the quantities Λ_α in the right-hand columns shows only minor discrepancies between the two for nearly all of the EPs. The implication of this is that the EP atomic energies determined via the regression are approximately equal to the atomic energies $\mathcal{V}_\alpha^{\mathrm{EP}}$ we have defined analytically in Appendix C for these environments, which is remarkable for two reasons. First, as mentioned above, the atomic energies which can be defined for an EP which are consistent with its predictions for total energies are highly non-unique. Second, although the descriptor used to conduct the regression has been chosen so as to constitute a faithful representation of atomic environments, it is manifestly different (at least, in practical terms) from the bond lengths and bond angles which serve as the native input space of the EPs of our study, including with respect to the spatial cutoff enforced. Furthermore, because Figure 6.39 reveals the atomic energies $\varepsilon_\alpha^{\mathrm{DFT}}$ to be approximately constant across isomap I, the error patterns observed for each EP can be understood by examining the different components $\mathcal{V}_{2,\alpha}^{\mathrm{EP}}$ and $\mathcal{V}_{3,\alpha}^{\mathrm{EP}}$ of the atomic energies $\mathcal{V}_\alpha^{\mathrm{EP}}$ (or $\mathcal{V}_{2,\alpha}^{\mathrm{EP}}$ and $\mathcal{V}_{U,\alpha}^{\mathrm{EP}}$ in the case of LSA), which can be found in Figures E.3 and E.4.

Among the most striking atomic energy errors are those of the SW and SWS1 potentials, which indicate a strong mismatch between their atomic energies and the atomic energies inferred from DFT. This is especially true of SWS1, whose atomic energies differ from $\varepsilon_\alpha^{\mathrm{DFT}}$ by at least 1 eV for all environments in isomap I. While the two-body energies $\mathcal{V}_{2,\alpha}^{\mathrm{SW}}$ of SW are significantly lower than those of SWS1 for the hcp, fcc, and bcc environments, which have the greatest short-range coordination, the three-body energies $\mathcal{V}_{3,\alpha}^{\mathrm{EP}}$ are drastically higher for SW than SWS1, leading it to yield atomic energies for these environments which remain much higher than those inferred from the regression. Although these EPs have different equilibrium bond angles for their three-body interactions (109.47° and 116°, respectively, from Figure C.5), this effect is primarily due to the fact that the three-body energy associated with both very low and very high bond angles is considerably larger for SW than SWS1. Nevertheless, the atomic energies of SW do agree with the energies $\varepsilon_\alpha^{\mathrm{DFT}}$

for the rightmost environments in isomap I belonging to β-Sn, sc, and sh, which we note are closer to the diamond environments in the MDS coordinates shown in Figures 6.32 – 6.34. The atomic energy errors of the LSA potential appear to display an error pattern similar to SW, but significantly minimized in overall scale. Although the embedding energies $\mathcal{V}_{U,\alpha}^{\mathrm{LSA}}$ of LSA can be either positive or negative depending on the circumstances, Figure E.4 indicates that they are positive for nearly all environments in isomap I, implying that the three-body contribution to the embedding density ϱ^{CF} dominates its two-body component. However, their part in raising the atomic energies $\mathcal{V}_{\alpha}^{\mathrm{LSA}}$ is minimal compared to the remainder of the EPs, much like the three-body energies $\mathcal{V}_{3,\alpha}^{\mathrm{SWS1}}$ for these environments. Even with its two-body energies being higher than SW, this allows LSA to achieve closer agreement with the energies learned by the regression.

Of the Tersoff-type potentials, the T2 and TMOD potentials possess the lowest atomic energy errors over isomap I. Despite $\mathcal{V}_{3,\alpha}^{\mathrm{T2}}$ being nearly as high as the three-body energies of SW, the two-body energies $\mathcal{V}_{2,\alpha}^{\mathrm{T2}}$ are low enough to sufficiently compensate so that the overall atomic energies of T2 match the atomic energies determined by the regression. In fact, the two-body energies of T2 are almost as low as those of EA; although Figure C.3 illustrates that the two-body energy of EA for a single neighbor as a function of distance is substantially deeper than that of T2, the larger cutoff range of T2 is of evident importance for the environments in isomap I, implying that many of them feature neighbors which sit at distances in the range of 2.8–3.2 Å. A similar effect can be observed between T3 and the longer-ranged SW potential in Figure E.3. Although the values of $\mathcal{V}_{3,\alpha}^{\mathrm{T3}}$ are weaker than both $\mathcal{V}_{3,\alpha}^{\mathrm{T2}}$ and $\mathcal{V}_{3,\alpha}^{\mathrm{EA}}$, this results in T3 overpredicting many of the atomic energies. Finally, Figures E.3 and E.4 reveal TMOD as a balance of extremely high three-body energies and extremely low two-body energies compared to all of other potentials. As we previously remarked for the total energies, the fine correspondence between the atomic energies $\mathcal{V}_{\alpha}^{\mathrm{TMOD}}$ and $\varepsilon_{\alpha}^{\mathrm{DFT}}$ for the environments in isomap I, all of which belong to metallic polytypes, is surprising given the severe angular sensitivity of the three-body energies of TMOD.

The only EP for which our partition of atomic energies $\mathcal{V}_{\alpha}^{\mathrm{EP}}$ differs significantly from that learned by the regression procedure is EDIP. From Figure 6.41, the atomic energies of EDIP derived from the regression appear more extremized—there are more environments which are assigned energies far above or below the corresponding quantities $\varepsilon_{\alpha}^{\mathrm{DFT}}$ than in our partition. Interestingly, this also leaves the energy error partition of the regression for EDIP as the only one to deviate from the average energy error patterns in Figures 6.18 and 6.19. While it is difficult to precisely discern the cause of this disparity, we posit that it is related to the complexity of EDIP which stems from the coordination dependence of its

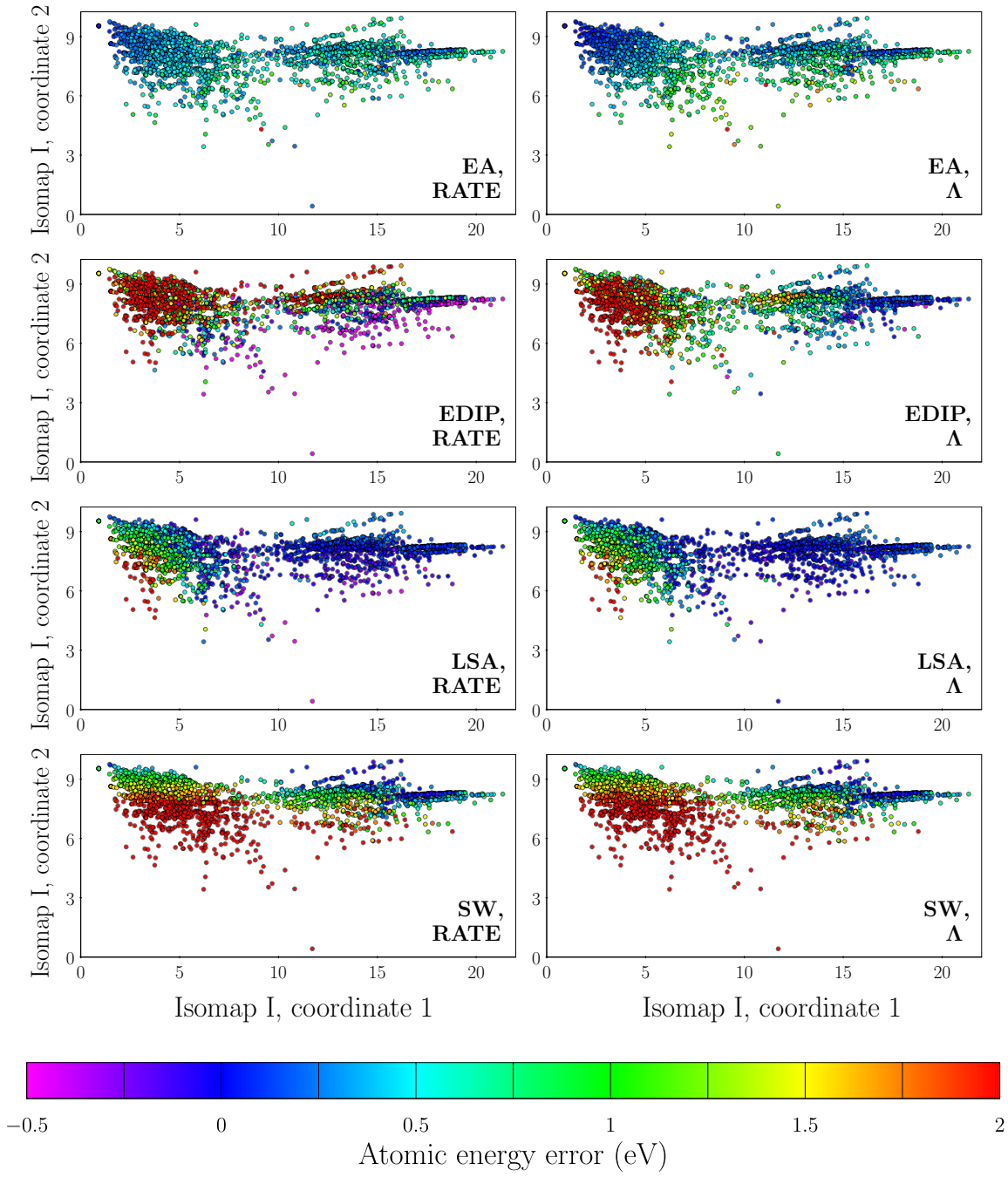


Figure 6.41: Atomic energy errors of the EA, EDIP, LSA, and SW EPs for the environments in group I. The left-hand column contains the atomic energy errors $\tilde{\varepsilon}_\alpha^{\mathrm{EP}}$ learned for each EP by RATE, while the right-hand column contains the corresponding errors $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20).

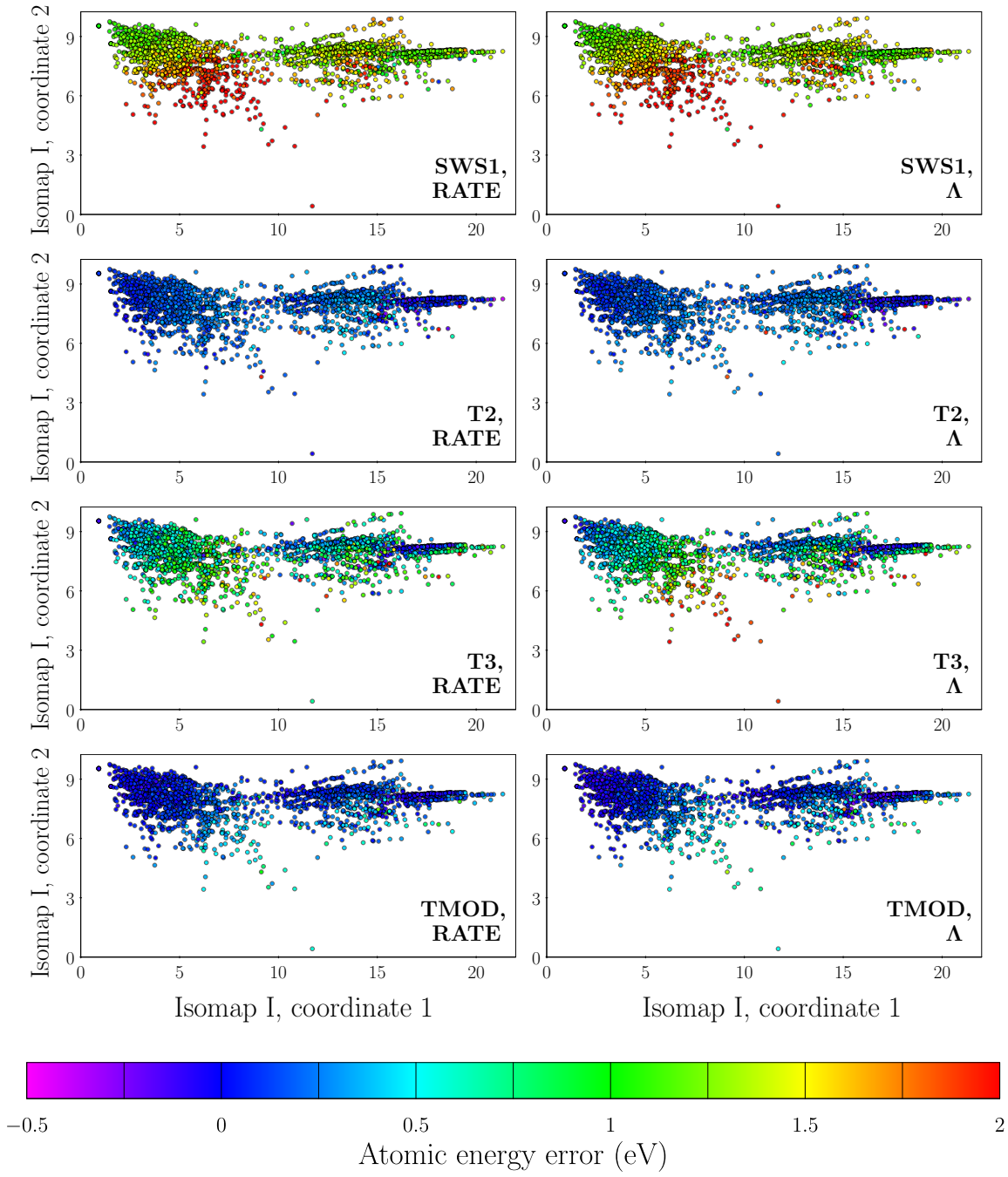


Figure 6.42: Atomic energy errors of the SWS1, T2, T3, and TMOD EPs for the environments in group I. The left-hand column contains the atomic energy errors $\check{\varepsilon}_\alpha^{\mathrm{EP}}$ learned for each EP by RATE, while the right-hand column contains the corresponding errors $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20).

functional forms. This mismatch notwithstanding, both partitions result in atomic energies which are far above the values of $\varepsilon_\alpha^{\mathrm{DFT}}$ in the leftmost portion of isomap I, where the environments possess the highest EDIP coordination values, approximately $Z_\alpha \geq 8$ (see Figure E.17). As the three-body energies $\mathcal{V}_{3,\alpha}^{\mathrm{EDIP}}$ are negligible over almost all of isomap I, it is evident that the two-body energies of EDIP become much too high with respect to $\varepsilon_\alpha^{\mathrm{DFT}}$ for these coordinations.

Fair agreement between the RATE atomic energy errors $\check{\varepsilon}_\alpha^{\mathrm{EP}}$ and the quantities $\Lambda_\alpha^{\mathrm{EP}}$ for the diamond, hexagonal diamond, and bc8 environments is shown in Figures 6.43 and 6.44, although EDIP again serves as a clear exception. The energy partition resolved by the regression for EDIP once more consists of a mixture of atomic energies which far exceed the corresponding $\varepsilon_\alpha^{\mathrm{DFT}}$ values or fall well below them. To the contrary, the energies $\mathcal{V}_\alpha^{\mathrm{EDIP}}$ defined in Section C.9 are shown to predominantly coincide with $\varepsilon_\alpha^{\mathrm{DFT}}$ on isomap II other than for the environments highlighted in Figure 6.40 which include neighbors at distances less than 1.85 Å. Comparing these environments for each of the potentials indicates that all of them but EDIP overpredict their atomic energies. This is in keeping with Figure C.3, which shows that when one atom of a dimer is considered, the two-body energies $\mathcal{V}_{2,\alpha}^{\mathrm{EP}}$ of each of these EPs are greater than one half of the total DFT energy of the dimer for small bond lengths; on the other hand, $\mathcal{V}_{2,\alpha}^{\mathrm{EDIP}}$ begins to underpredict the energy of the dimers which have separations below roughly 1.75 Å when the coordination Z_α is equal to four, as it is on almost all of isomap II (Figure E.17). Neglecting the environments with short bond distances in them, the energies $\mathcal{V}_\alpha^{\mathrm{EP}}$ of each of the EPs generally match $\varepsilon_\alpha^{\mathrm{DFT}}$ other than in the case of the SWS1 potential and for some of the environments in the far left of isomap II.

For the perturbed graphite environments in group III, shown in Figures 6.45 and 6.46, all of the Tersoff-type potentials show similar $\check{\varepsilon}_\alpha^{\mathrm{EP}}$ and $\Lambda_\alpha^{\mathrm{EP}}$ to one another and to the LSA potential, although the former quantities indicate somewhat greater discrepancies with $\varepsilon_\alpha^{\mathrm{DFT}}$ than the latter. The SW and SWS1 potentials show the greatest disagreement with the DFT atomic energies in terms of both $\check{\varepsilon}_\alpha^{\mathrm{EP}}$ and $\Lambda_\alpha^{\mathrm{EP}}$. In contrast to the environments in isomaps I and II, the errors of EDIP for the perturbed graphite environments show excellent consistency with those of our own partition. For the environments on the far right of isomap III, many of which are shown to contain neighbors within 1.85 Å in Figure 6.40, EDIP is once again indicated to underpredict the corresponding $\varepsilon_\alpha^{\mathrm{DFT}}$ as in isomap II, while the Tersoff and Stillinger–Weber potentials overpredict them. However, there are also many environments interspersed with them for which $\Lambda_\alpha^{\mathrm{EP}}$ indicates that the atomic energies of the EP are below $\varepsilon_\alpha^{\mathrm{DFT}}$, giving an error distribution which is noticeably less smooth than

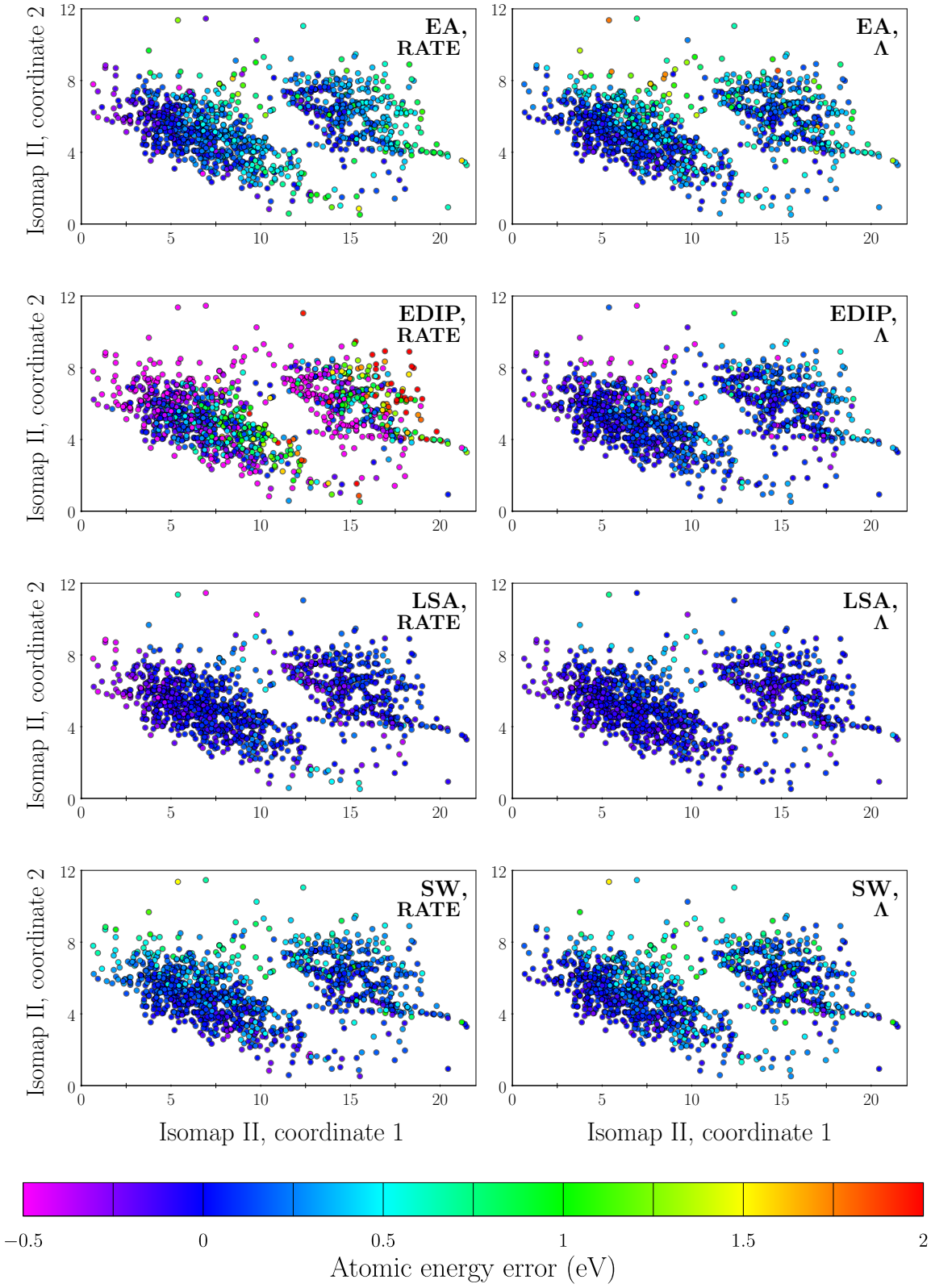


Figure 6.43: Atomic energy errors of the EA, EDIP, LSA, and SW EPs for the environments in group II. The left-hand column contains the atomic energy errors learned for each EP by RATE, while the right-hand column contains the corresponding errors $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20).

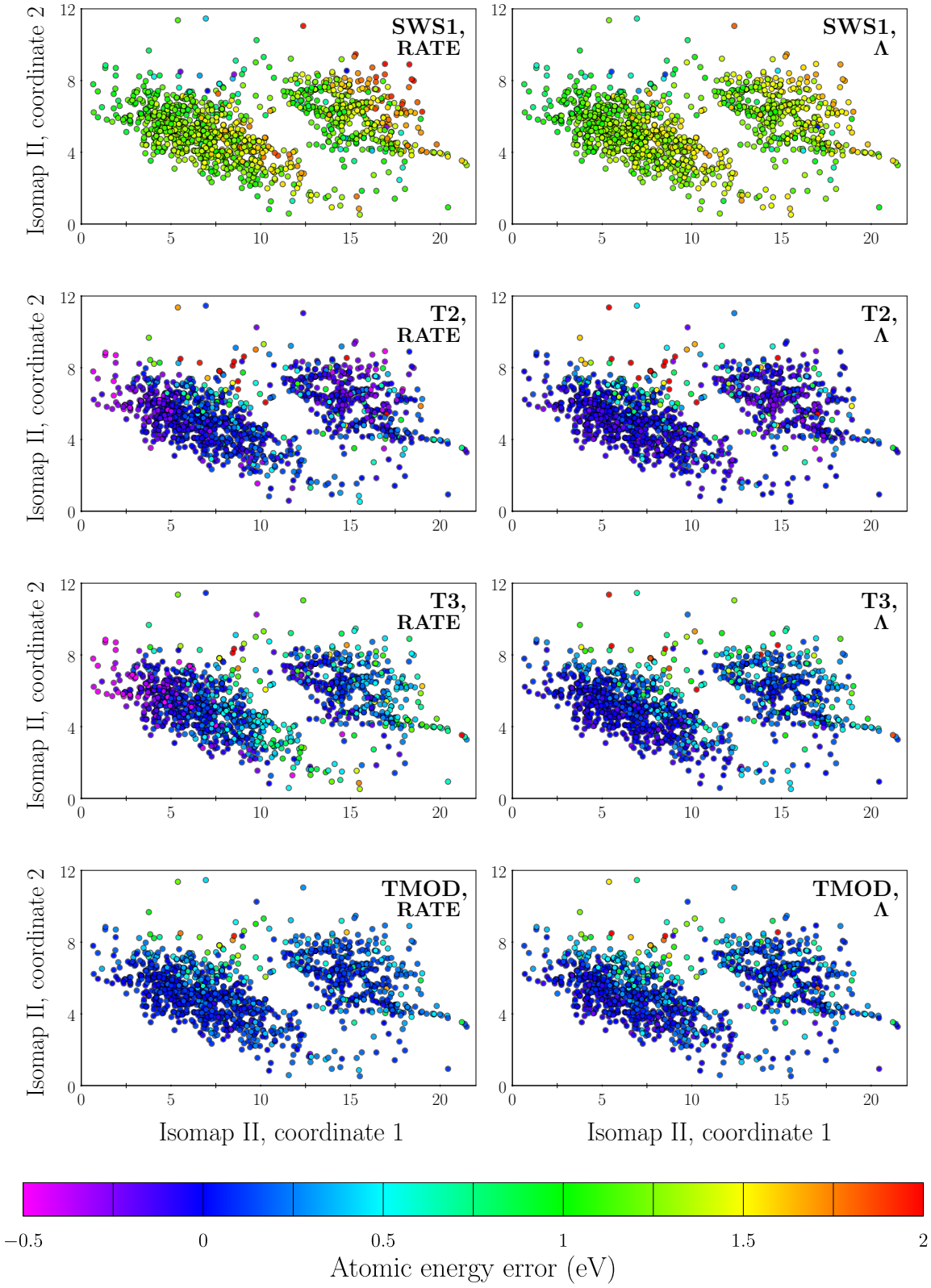


Figure 6.44: Atomic energy errors of the SWS1, T2, T3, and TMOD EPs for the environments in group II. The left-hand column contains the atomic energy errors learned for each EP by RATE, while the right-hand column contains the corresponding errors $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20).

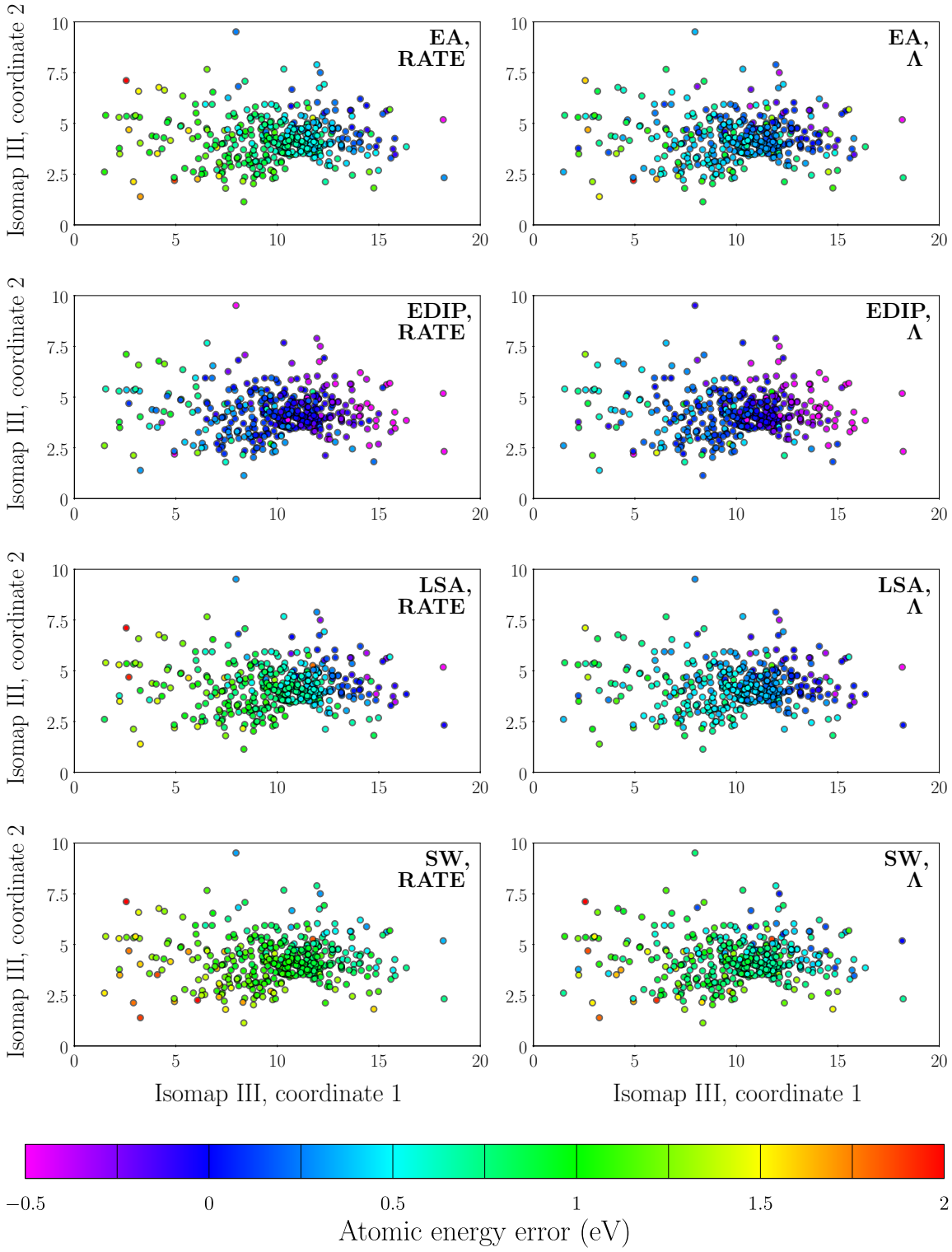


Figure 6.45: Atomic energy errors of the EA, EDIP, LSA, and SW EPs for the environments in group III. The left-hand column contains the atomic energy errors learned for each EP by RATE, while the right-hand column contains the corresponding errors $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20).

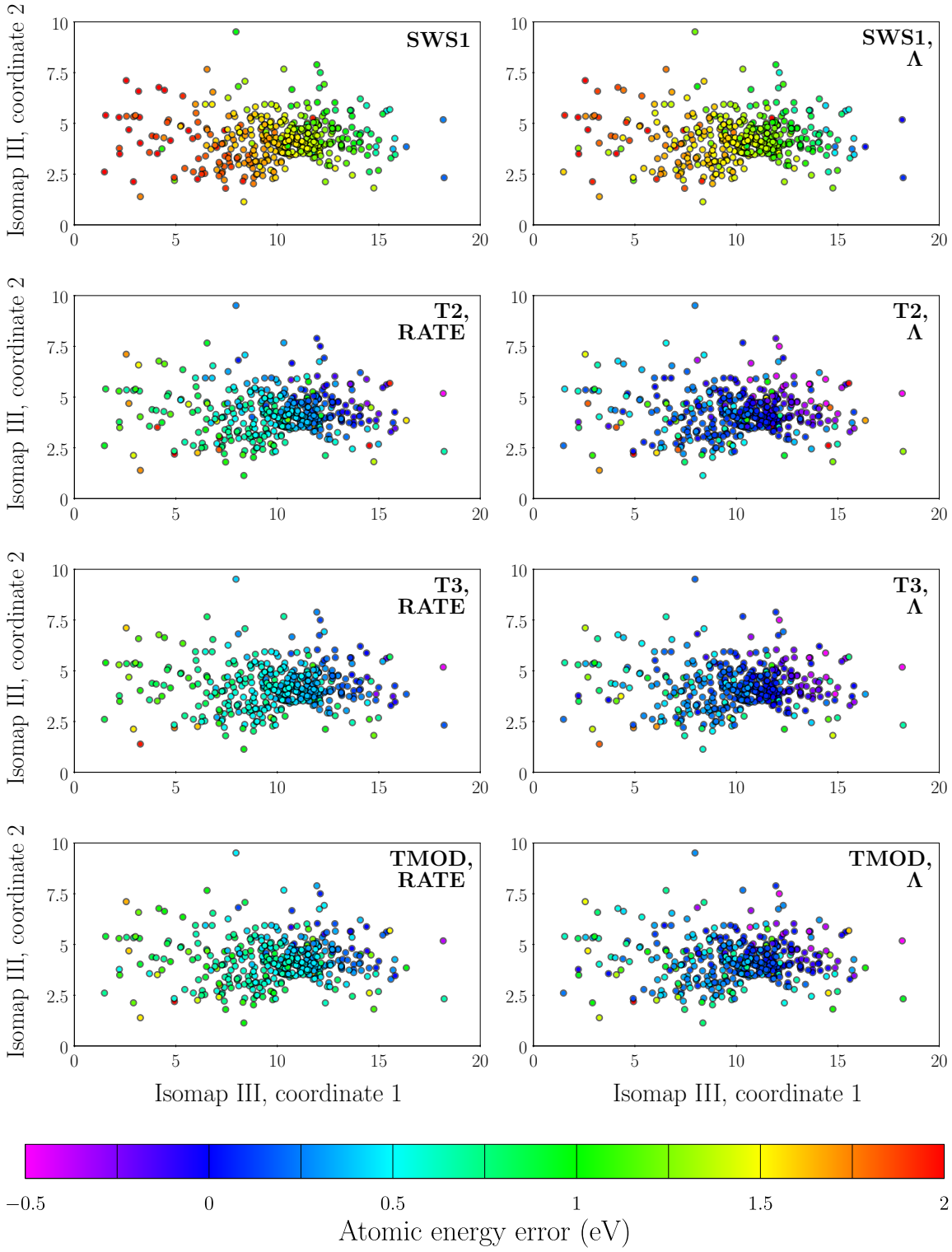


Figure 6.46: Atomic energy errors of the SWS1, T2, T3, and TMOD EPs for the environments in group III. The left-hand column contains the atomic energy errors learned for each EP by RATE, while the right-hand column contains the corresponding errors $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20).

that inferred by the regression. Aside from EDIP and SW, the atomic energy errors demonstrate greater overall deviation from the average energy errors than in isomaps I and II, implying a greater sensitivity of the atomic energies to local geometry.

Having reviewed the MDS coordinates of the atomic environments found in the bulks and their energy errors, we next examine the cluster environments of group IV. Figure 6.47 highlights the location of the dimer environments in isomap IV in terms of their bond lengths. As in Figure 6.20, the environments which have bond lengths closest to the nearest-neighbor distance in the ideal diamond lattice fall closest to the rest of the cluster environments. A flaw in the MDS embedding of the atomic environments revealed here is the fact that there are two distinct coordinates for each bond length which accordingly comprise two branches in isomap IV, a clearly erroneous feature considering both environments of a given dimer are trivially identical for any descriptor by way of symmetry. Although we believe this to be an isolated failure because the placement of the dimer environments in the MDS embedding is largely independent of the remaining environments, a thorough formal investigation to determine the complete implications of this phenomenon has not been performed. Figures 6.48 – 6.51 depict the positions of the clusters containing three or more atoms. Much like in Figures 6.21 and 6.22, a trend is demonstrated wherein the environments of the elongated clusters of a given size fall to the right of the corresponding compact clusters, and environments generally proceed to the left of the diagram as cluster size increases. However, for each cluster size, the elongated cluster environments are still distributed across nearly the entire portion of isomap IV which lies to the right of the environments of the compact clusters of that size. The coordination $\Gamma(\alpha)$ of the various environments in Figure 6.52 varies smoothly in accordance with these trends, as do the first-principles atomic energies $\varepsilon_\alpha^{\mathrm{DFT}}$ from the regression, shown in Figure 6.53. These energies are greatest for the environments which possess the highest and lowest coordination at the far left and far right, respectively, while the center contains a segment of environments with near-tetravalent coordinations which are assigned the lowest energies (note, however, that there are also many environments with coordination four which are assigned high energies by the regression).

Figures 6.54 – 6.61 show the atomic energy errors $\Lambda_\alpha^{\mathrm{EP}}$ and $\tilde{\varepsilon}_\alpha^{\mathrm{EP}}$ of each EP over isomap IV. Both sets of errors largely coincide other than for the environments in the upper left corner which, as can be seen from the EDIP coordination Z_α in Figure E.18, have the greatest short-range coordination among the cluster environments. For these environments, the energy errors of our partition are higher than those inferred by the regression due to the dominant three-body energies $\mathcal{V}_{3,\alpha}^{\mathrm{EP}}$ of nearly all of the potentials (cf. Figures E.13 – E.16).

A notable exception is EDIP, for which the values $\mathcal{V}_{3,\alpha}^{\mathrm{EDIP}}$ are comparatively low for almost all of the cluster environments. However, the two-body energies $\mathcal{V}_{2,\alpha}^{\mathrm{EDIP}}$ of EDIP are exceedingly high in this portion of isomap IV relative to the other potentials (Figures E.9 – E.12), yielding atomic energies substantially higher than those determined by the regression. The T3 potential also shows significant deviation between the errors inferred from regression and our own, with the distribution of high error being concentrated toward the lower portion of isomap IV—a tendency which can be observed to various extents with most of the EPs. The only potentials for which the errors $\Lambda_{\alpha}^{\mathrm{EP}}$ and $\check{\varepsilon}_{\alpha}^{\mathrm{EP}}$ show an almost exact correspondence are SW and SWS1. It may also be observed that errors $\Lambda_{\alpha}^{\mathrm{EP}}$ and $\check{\varepsilon}_{\alpha}^{\mathrm{EP}}$ of the dimer environments are identical for each EP, and this follows as a trivial consequence of the objective symmetry of dimers, i.e. both atoms of a dimer have identical chemical environments. Because the individual atomic energy errors are required to reproduce the total energy error of each dimer, the energy error assigned to each atom of a dimer must be precisely half of the total energy error.[22] In spite of the aforementioned differences between the errors inferred by the regression and those of our own partition, the T2 potential is decidedly more consistent with the values of $\varepsilon_{\alpha}^{\mathrm{DFT}}$ on isomap IV than the others. Moreover, in contrast to the bulk environments, its performance is significantly better than TMOD for all of the environments aside from those with the lowest coordination. In fact, insofar as isomap IV is concerned, $\Lambda_{\alpha}^{\mathrm{LSA}}$ and $\check{\varepsilon}_{\alpha}^{\mathrm{LSA}}$ are both more favorable than those of TMOD, whose atomic energy error pattern is similar to that of EA. The EDIP potential appears to provide the next best match after LSA, while the T3, SW, and SWS1 potentials prove to be the most inaccurate with respect to $\varepsilon_{\alpha}^{\mathrm{DFT}}$.

Histograms of the atomic energy errors $\check{\varepsilon}_{\alpha}^{\mathrm{EP}}$ of each EP determined by RATE for all of the environments in the training set (groups I-IV, as well as the nanostructures) are shown in Figure E.19, while histograms of their absolute values are shown in Figure E.20. We provide an equivalent perspective of the same data in Figure E.21, which contains superposed histograms of the atomic energies learned by the regression from both the first-principles total energies and from the total energies for each EP. Finally, a global comparison between the atomic energies learned from the EP total energies and the atomic energies $\mathcal{V}_{\alpha}^{\mathrm{EP}}$ defined for each EP in Section C.9 is made in Figure E.22, which shows superposed histograms of each.

[22] In fact, the same logic holds for any objective structure and, in this sense, the partition of total energy (or total energy error) amongst the individual atoms in a training set of atomic configurations is unique. Unfortunately, these are the only cases for which such uniqueness can be deduced.

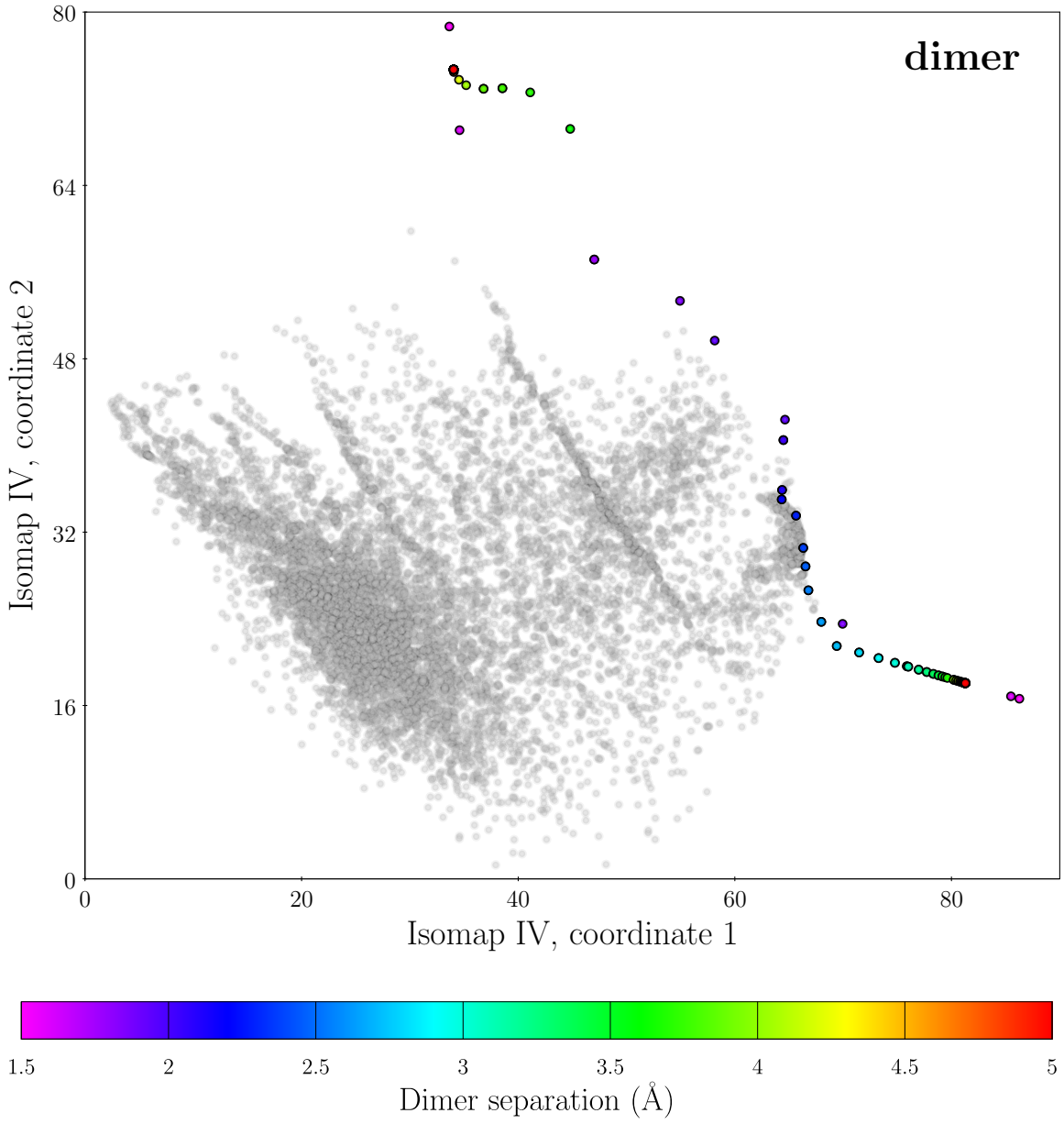


Figure 6.47: Isomap of group IV (clusters and dimers) highlighting the location of the atomic environments belonging to the dimers, which are shaded according to the associated bond lengths. The fact that there are two separate coordinates in isomap IV for each dimer environment constitutes a shortcoming of the MDS embedding.

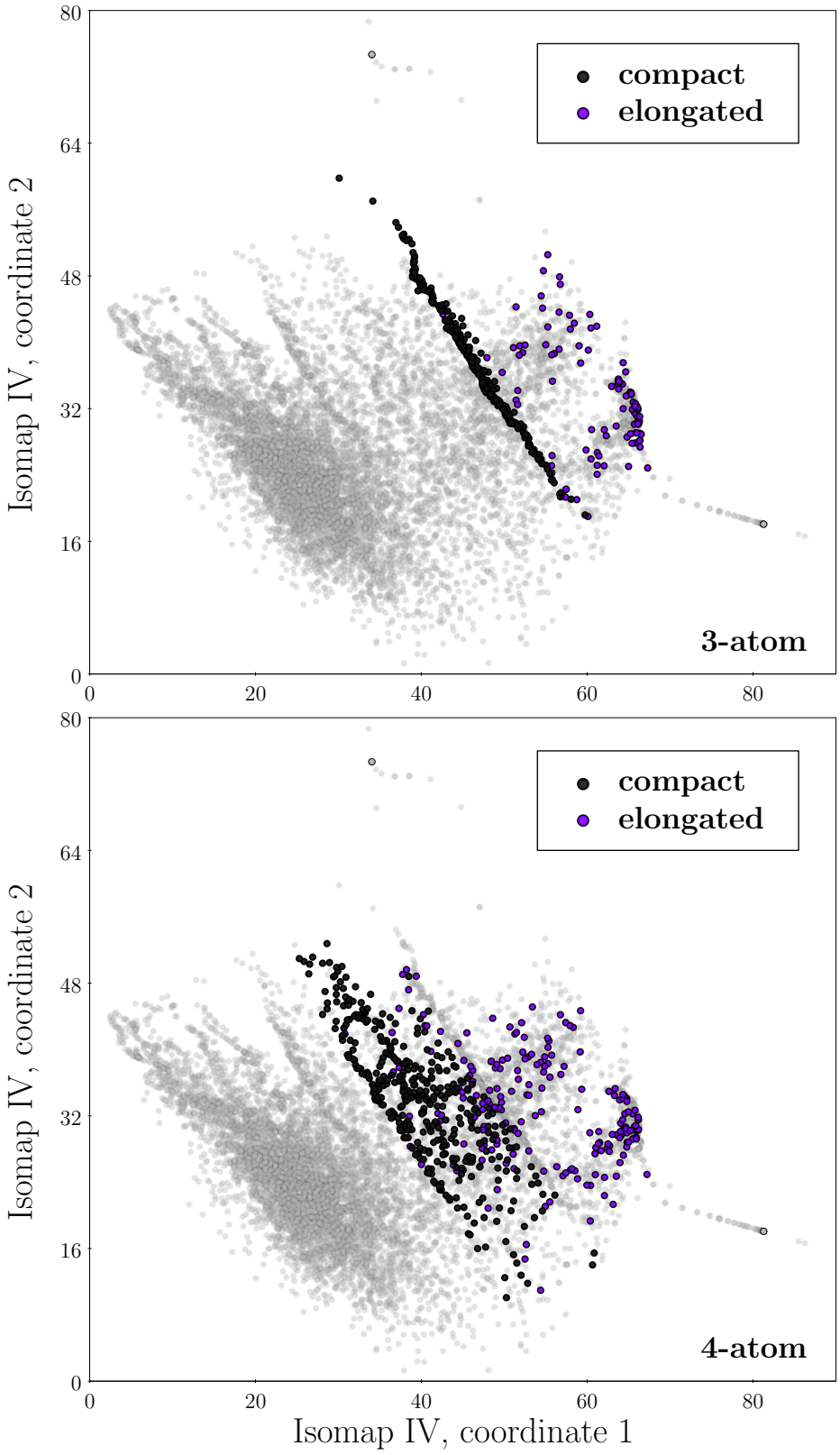


Figure 6.48: Isomap of group IV (clusters and dimers) illustrating the location of the atomic environments belonging to the 3-atom and 4-atom clusters.

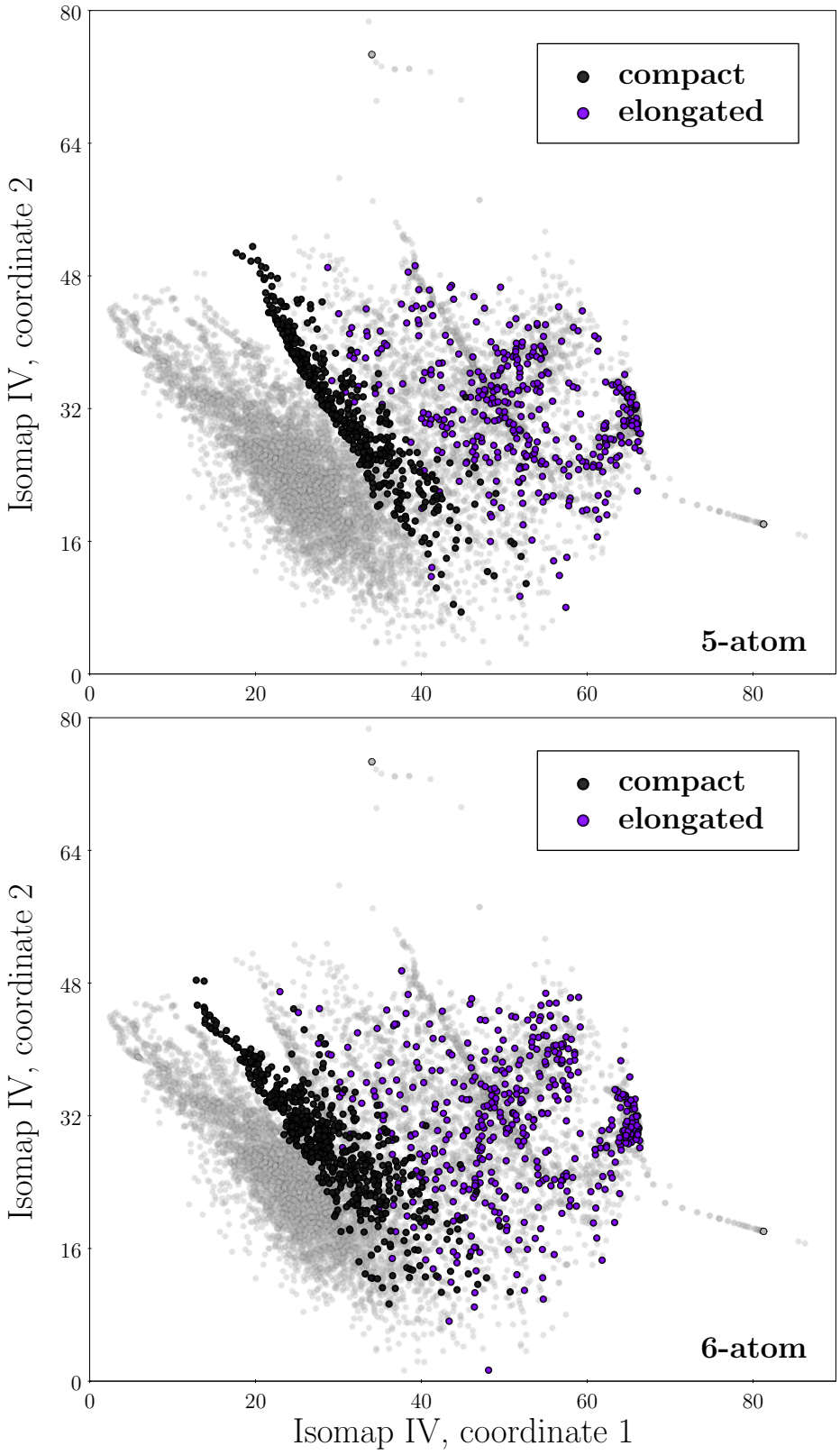


Figure 6.49: Isomap of group IV (clusters and dimers) illustrating the location of the atomic environments belonging to the 5-atom and 6-atom clusters.

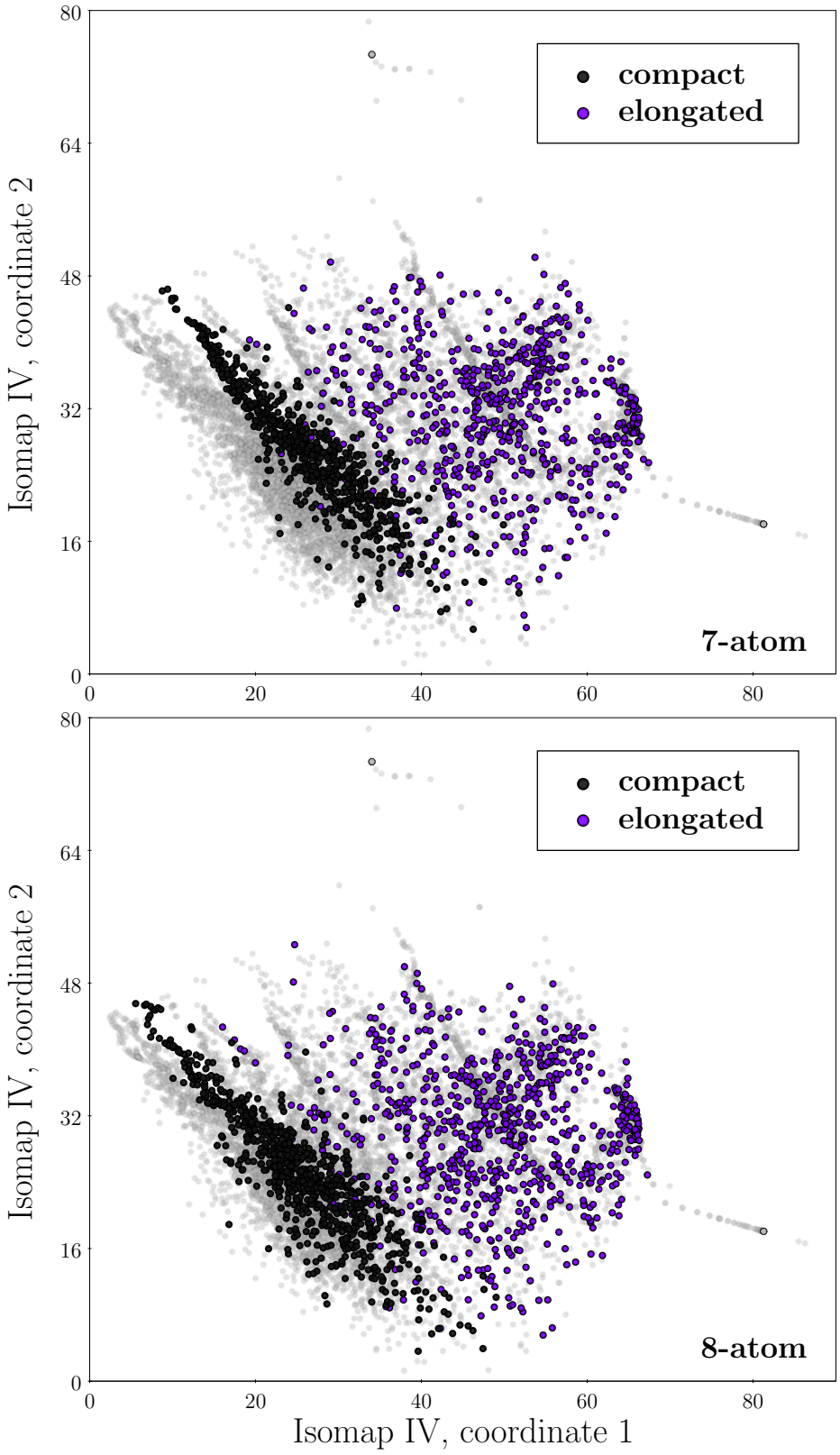


Figure 6.50: Isomap of group IV (clusters and dimers) illustrating the location of the atomic environments belonging to the 7-atom and 8-atom clusters.

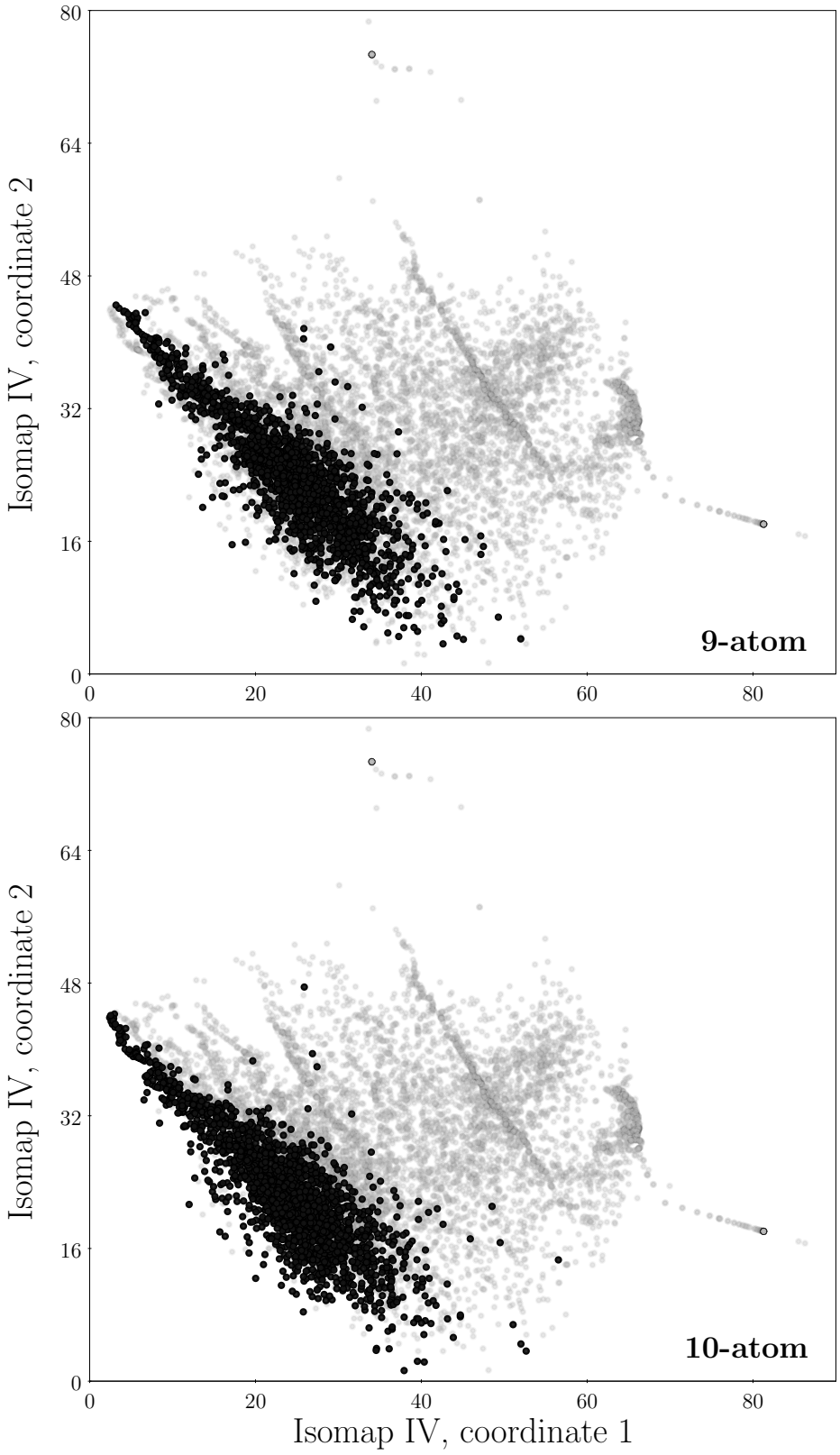


Figure 6.51: Isomap of group IV (clusters and dimers) illustrating the location of the atomic environments belonging to the 9-atom and 10-atom clusters.

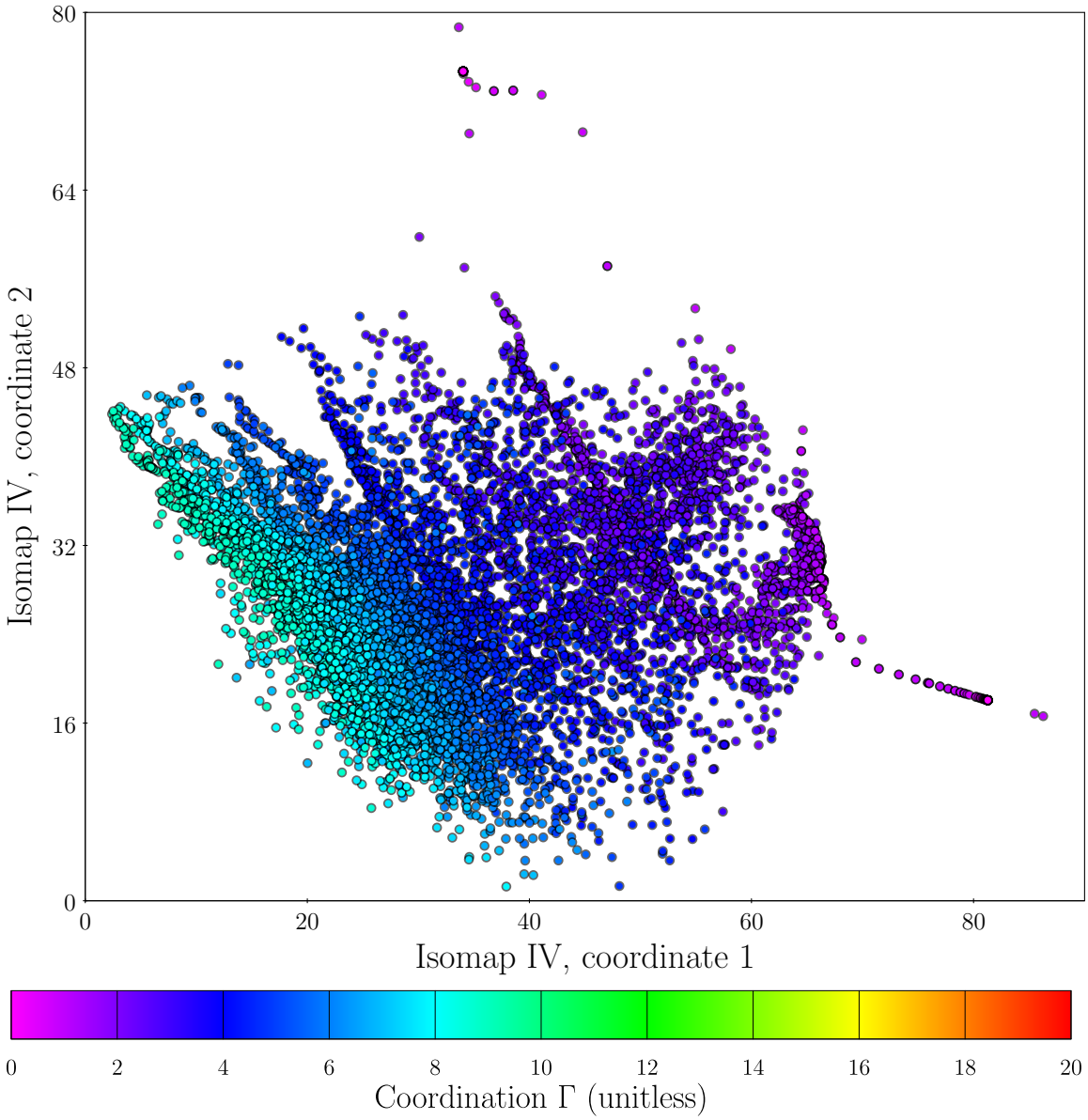


Figure 6.52: Coordination $\Gamma(\alpha)$ defined in (6.12) of the cluster and dimer environments, shown over isomap IV.

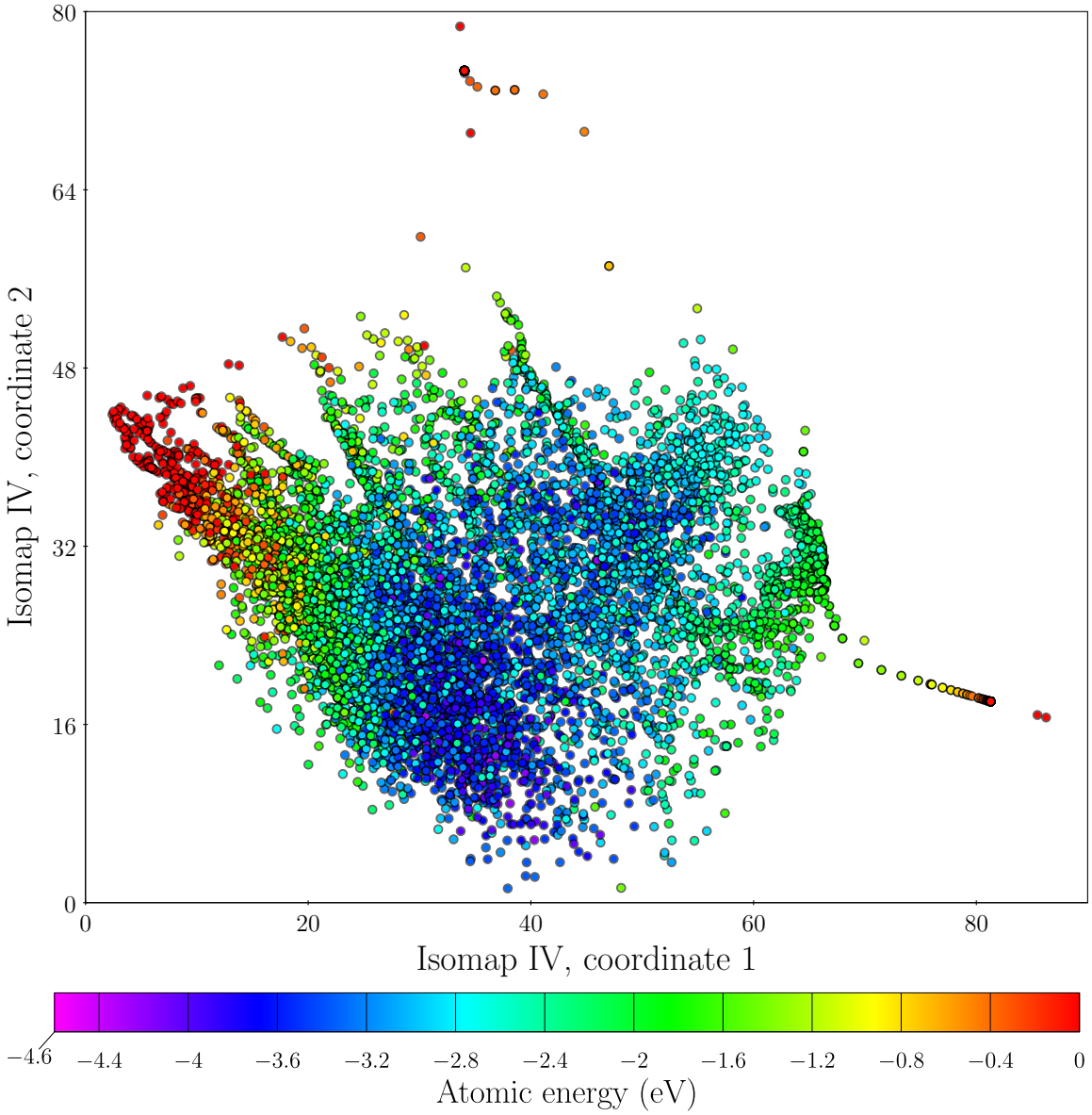


Figure 6.53: Atomic energies $\varepsilon_\alpha^{\mathrm{DFT}}$ inferred by the regression for the cluster configurations from the first-principles total energies of the training set configurations.

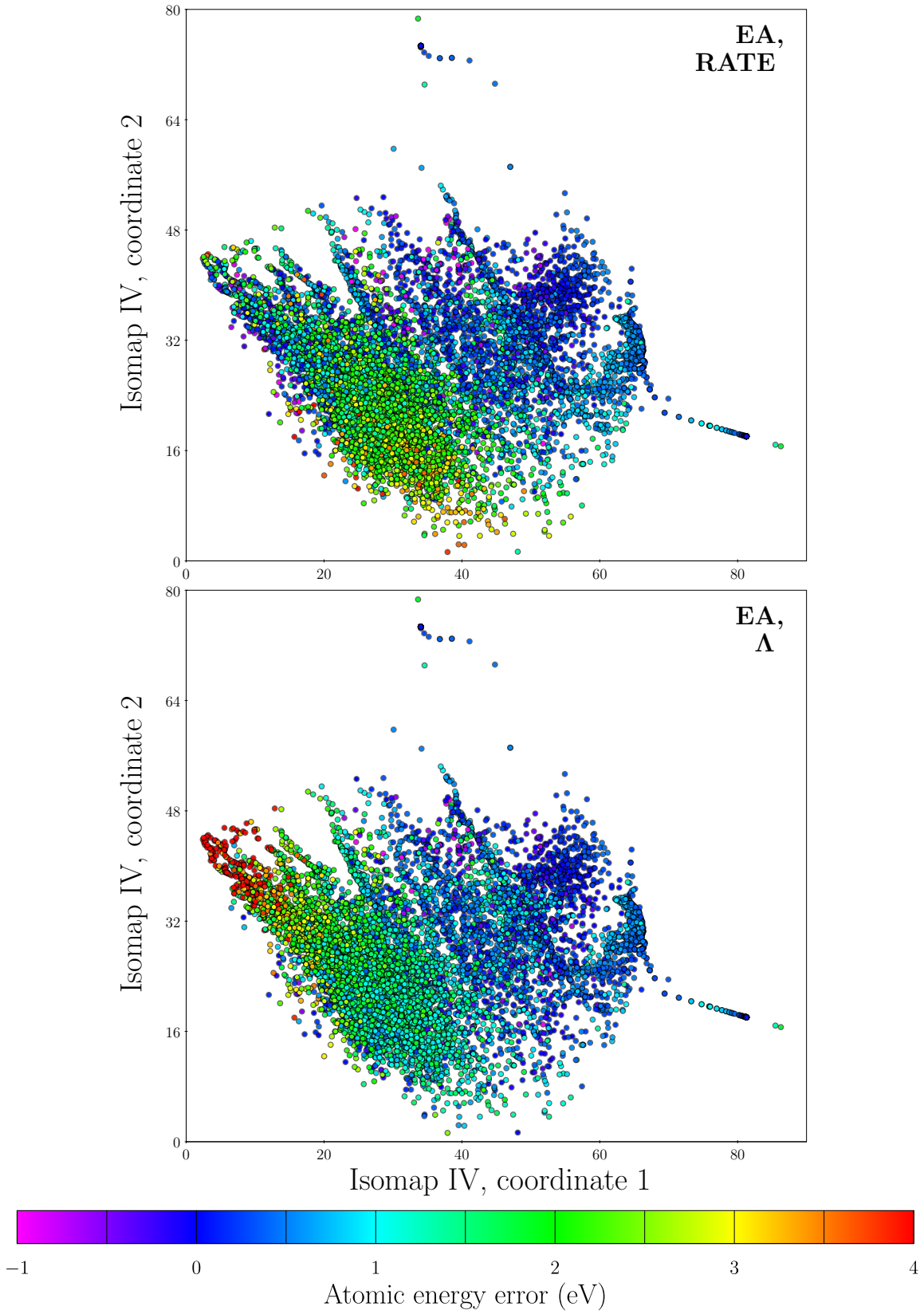


Figure 6.54: Atomic energy errors of the EA potential for the environments in group IV determined by RATE (top) and computed according to $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20) (bottom).

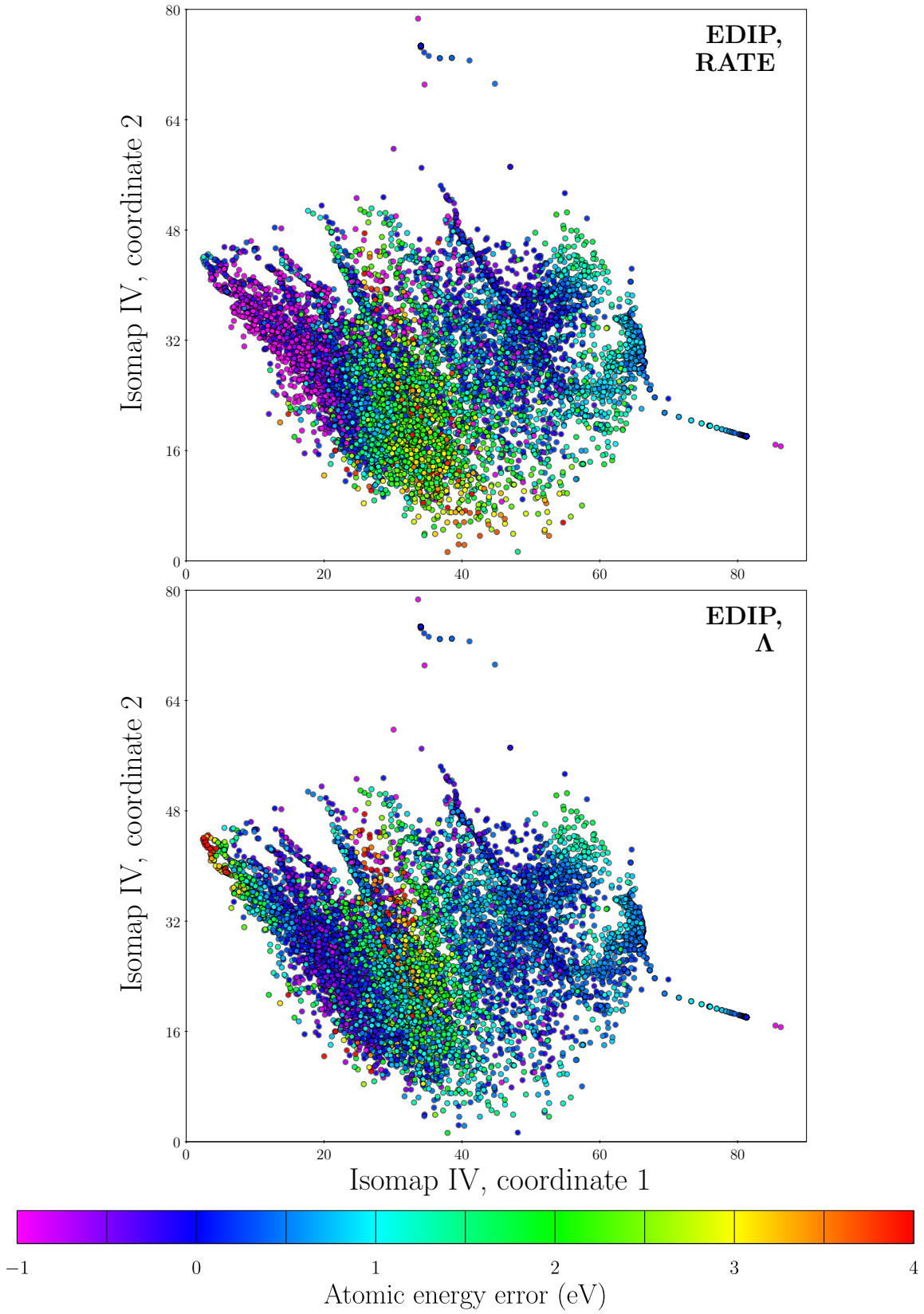


Figure 6.55: Atomic energy errors of the EDIP potential for the environments in group IV determined by RATE (top) and computed according to $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20) (bottom).

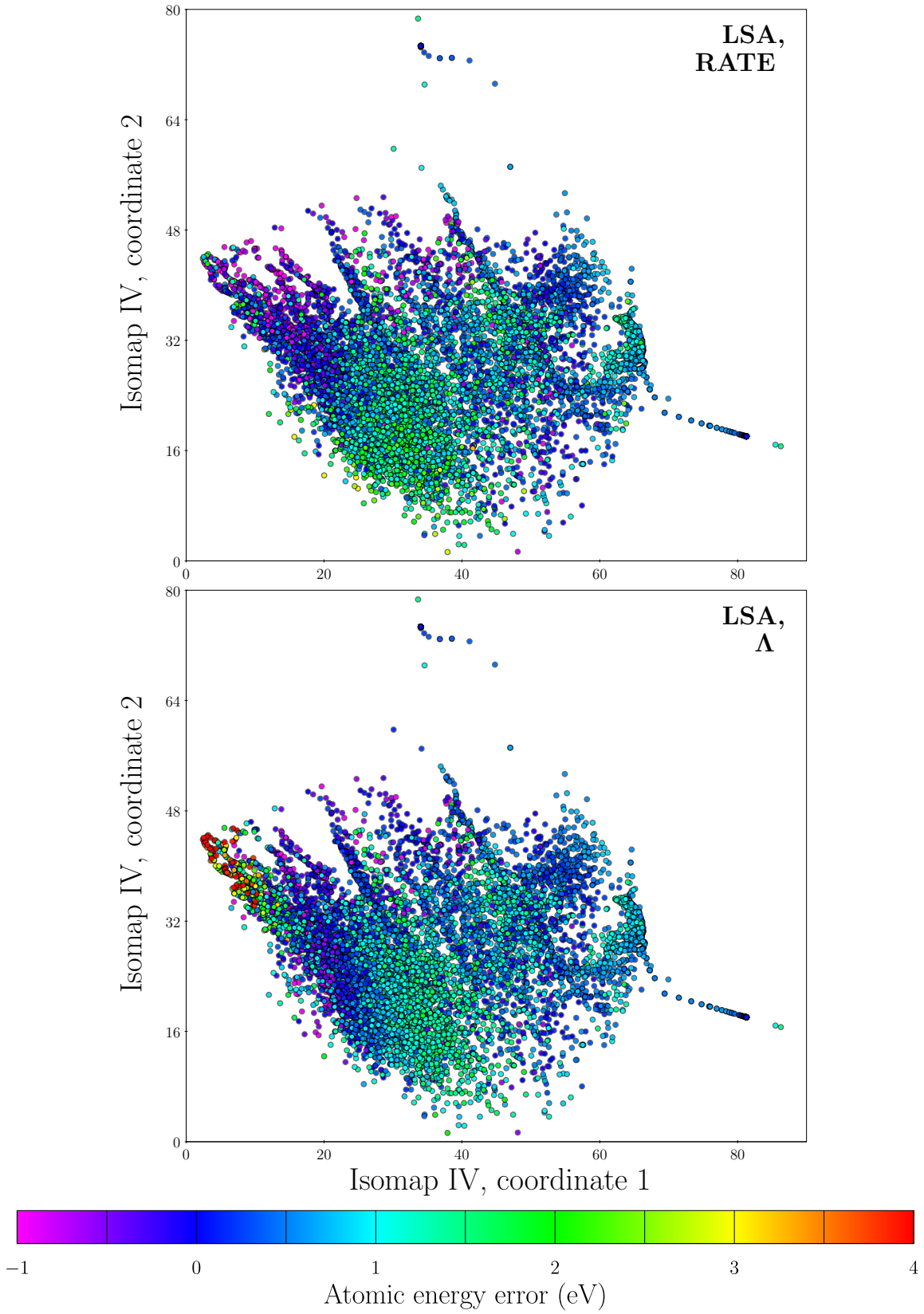


Figure 6.56: Atomic energy errors of the LSA potential for the environments in group IV determined by RATE (top) and computed according to $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20) (bottom).

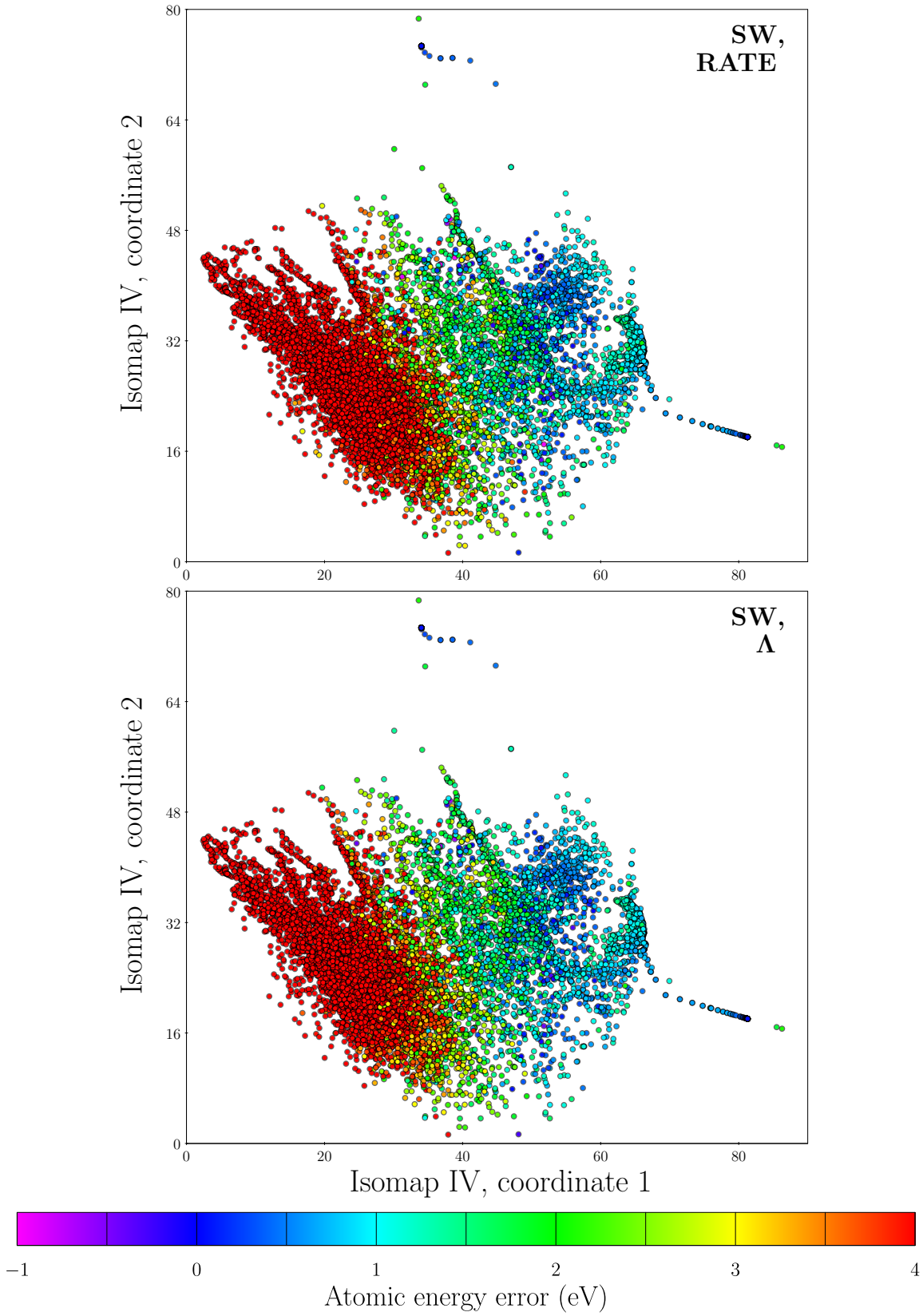


Figure 6.57: Atomic energy errors of the SW potential for the environments in group IV determined by RATE (top) and computed according to $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20) (bottom).

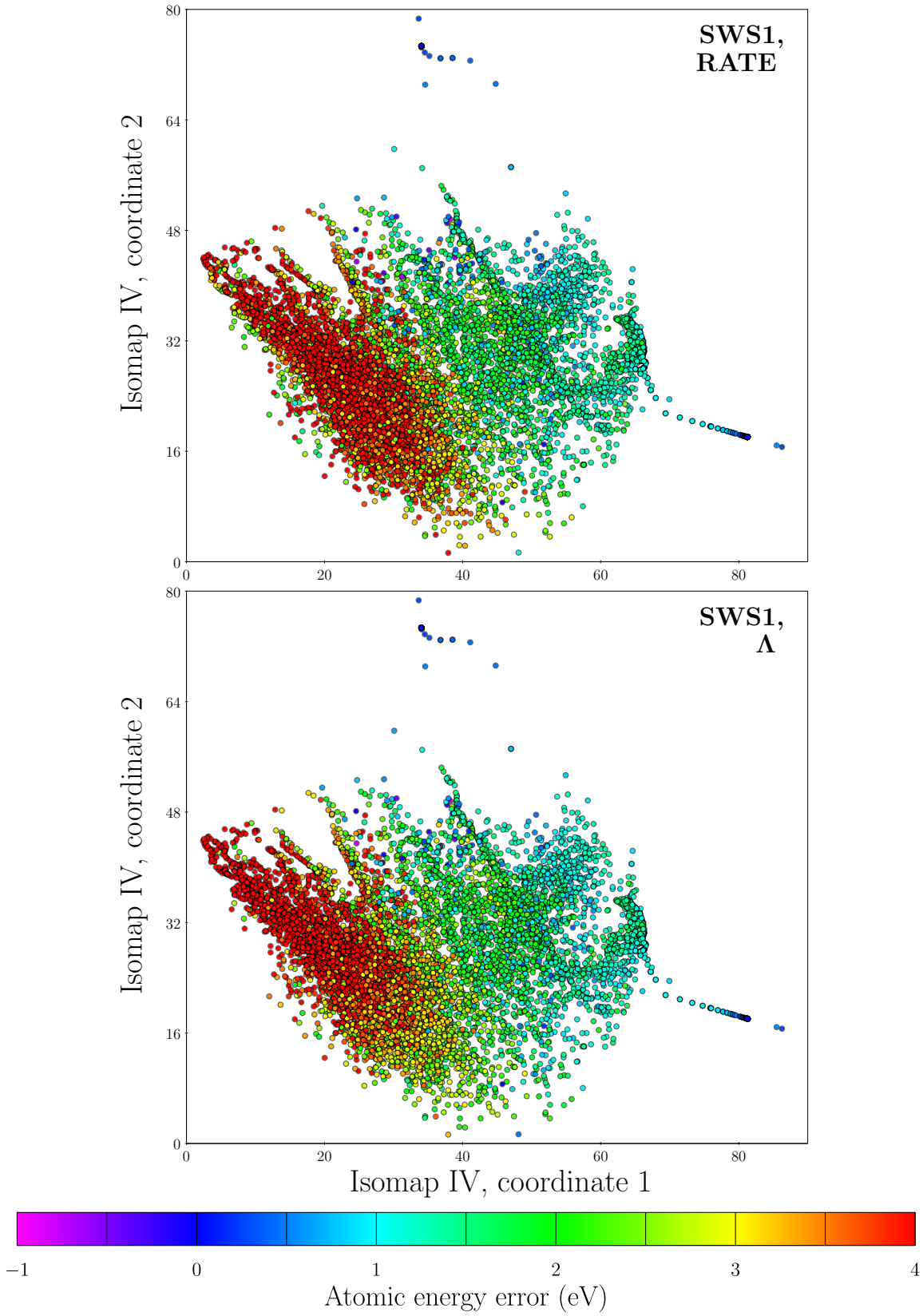


Figure 6.58: Atomic energy errors of the SWS1 potential for the environments in group IV determined by RATE (top) and computed according to $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20) (bottom).

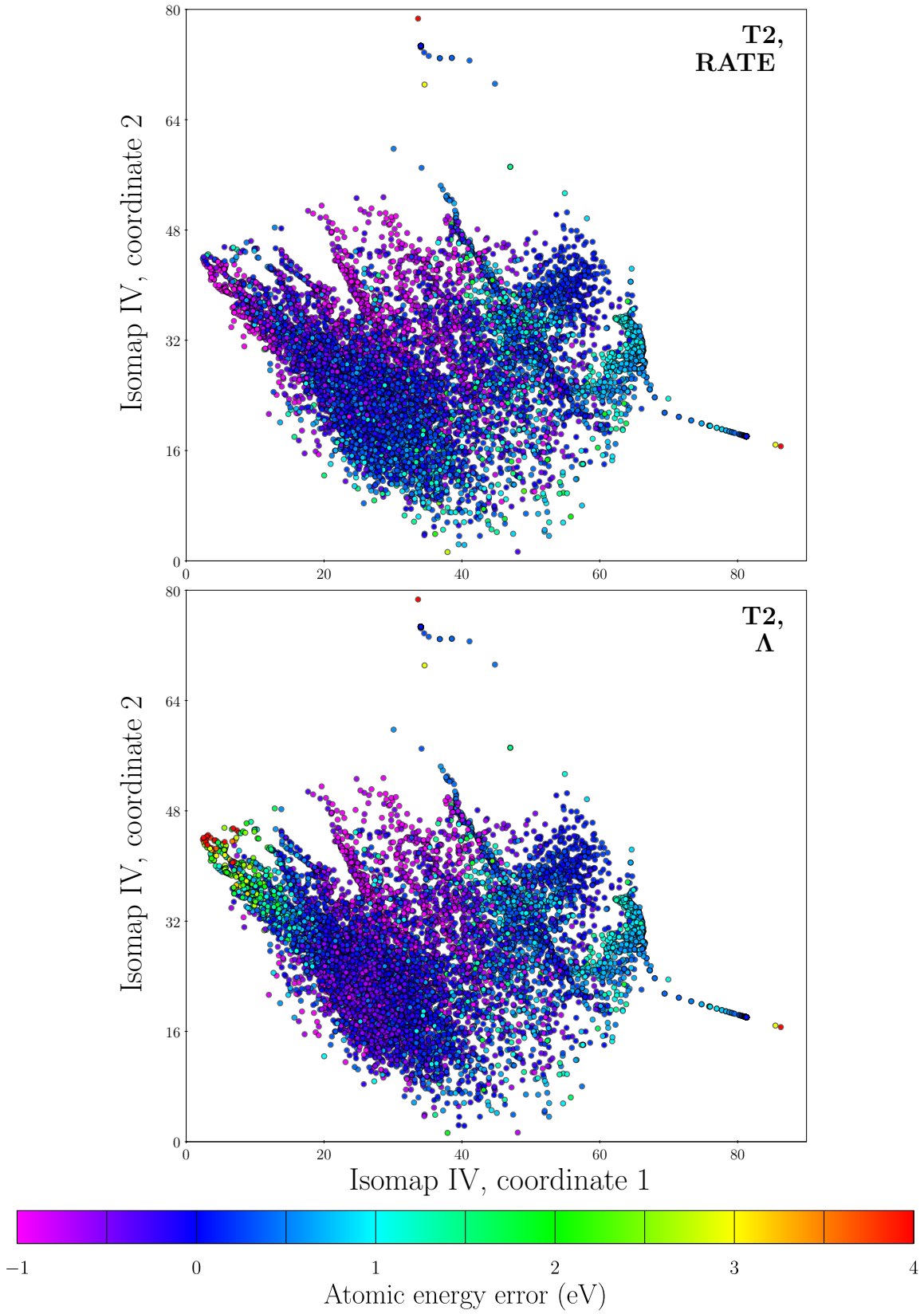


Figure 6.59: Atomic energy errors of the T2 potential for the environments in group IV determined by RATE (top) and computed according to $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20) (bottom).

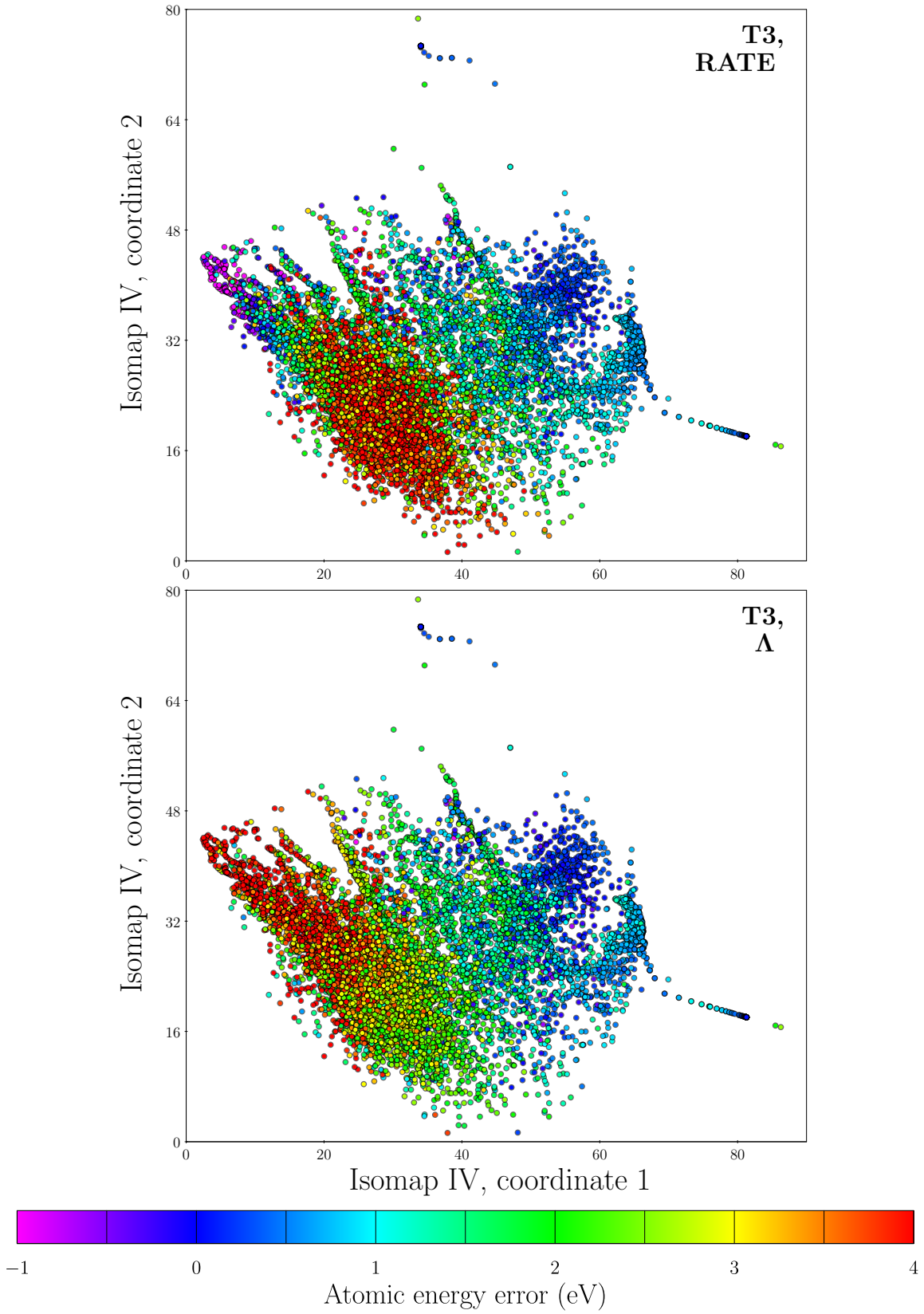


Figure 6.60: Atomic energy errors of the T3 potential for the environments in group IV determined by RATE (top) and computed according to $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20) (bottom).

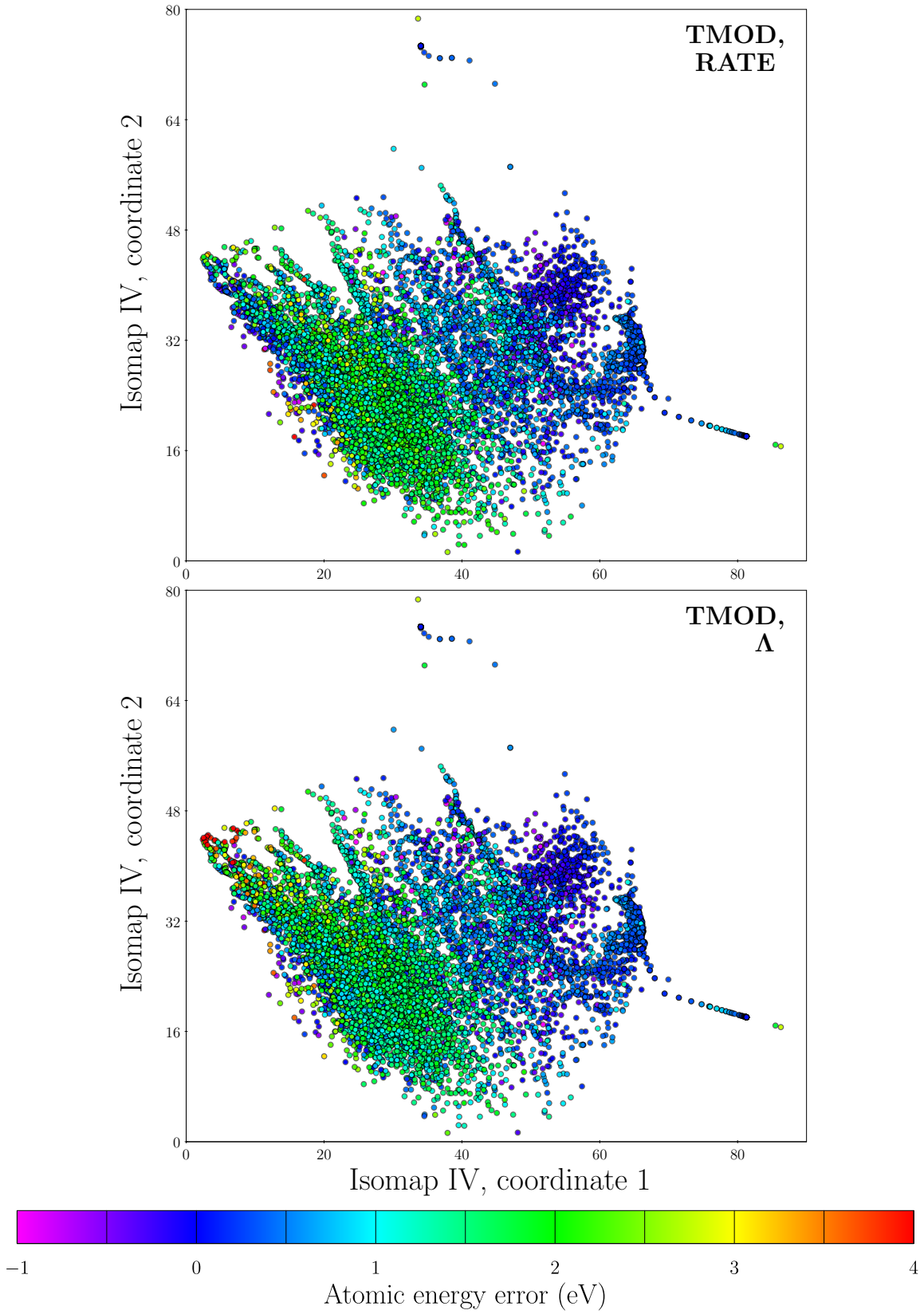


Figure 6.61: Atomic energy errors of the TMOD potential for the environments in group IV determined by RATE (top) and computed according to $\Lambda_\alpha^{\mathrm{EP}}$ defined in (6.20) (bottom).

6.4.3 Cross-validation

The previous sections of this chapter focused on understanding the regression on the total energy errors of an EP across atomic configurations proposed in Section 6.2. We began by visualizing the atomic configurations of a training set consisting of bulks, clusters, surfaces, and nanostructures in a low-dimensional space where the relative coordinates of each configuration reflected their corresponding similarity as it is defined in the regression method. Within this space, the average energy error per atom of a specific set of EPs was examined and it was shown that variations of these quantities were smooth, implying that the regression technique we pursue could prove efficacious in application. Prompted by the fact that the similarities between entire atomic configurations in our method are a function of the similarities between their respective atomic environments, further investigation was undertaken in Section 6.4.2 in which we performed a similar visualization of individual atomic environments and their energy errors. However, while these endeavors constitute an important step in understanding the underlying details and practical feasibility of the regression, they give no indication of its interpolative or extrapolative ability. Because it is these characteristics which ultimately determine the utility of any regression procedure, we conclude this chapter with their explicit analysis.

The standard method of evaluating the interpolative and extrapolative properties of a statistical model is known as *cross-validation*. Recall the notion of a *validation set* introduced in Sections 5.2.2 and 5.3 as a component of the training procedure used in the context of neural network and nonparametric EPs. There, the first step of the training process is to divide the primary training set into two complementary pieces: the secondary training set and the validation set. In each training iteration, a set of potential parameters are fit using the secondary training set and, thereafter, these parameters are used to make predictions for the material properties contained in the validation set, i.e. the material properties in the validation set are used as a test set. Comparison of the predictions for these properties with the current set of parameters with their actual values is then carried out according to some specific algorithm which then adjusts the potential parameters for the next training iteration. What was left unmentioned in this discussion is the arbitrariness of the division imposed on the primary training set. For this reason, it is common practice to consider several such divisions in parallel in order to mitigate any bias which might occur due to the selection process, and it is precisely this more general procedure which defines cross-validation. Typically, the subdivisions of the primary training set are defined at random subject to the constraint that each segment thus defined is of approximately equal size. At each stage of the cross-validation, referred to as a *fold*, a single subdivision is used as a

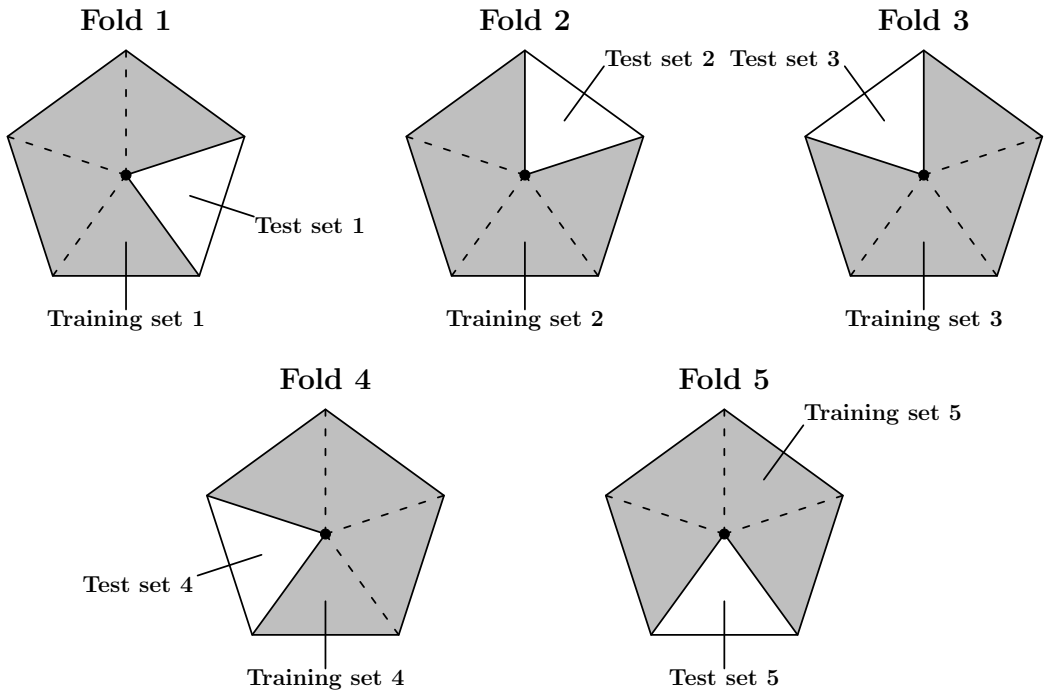


Figure 6.62: Schematic illustration of a five-fold cross-validation.

test set with the rest serving as a training set. The interpolative and extrapolative capability of the underlying model is then reflected in its cumulative accuracy across all five folds (which is sometimes averaged). A schematic illustration of a five-fold cross-validation is shown in Figure 6.62.

In applying cross-validation to our regression model, we have defined five complementary segments of the training set introduced in Section 6.3 whose corresponding subdivisions each contain 422 atomic configurations selected via uniform random sampling. Because this method of sampling is unbiased, it is expected that if the underlying model adequately captures the relevant features of the training set, it will reproduce the target observations of the test set.[23] Two cross-validation procedures were executed using these folds: first, the first-principles total energies were taken as the objective values. The outcome of this cross-validation is shown in Figure 6.63, where we have combined the results of all five folds and plotted the average first-principles energy per atom predicted for each test set configuration against its true value. The red line spanning the diagonal represents predictions which are free from error, while the dashed black lines indicate deviations of ± 0.15 eV from the correct values of the average energies and, for each prediction, we have in-

[23] At least, the cross-validation accuracy can be expected to be reasonably high if the process is repeated many times and/or the number of folds is taken to be large. The latter technique may or may not be applicable to a given problem based on the number of training samples available.

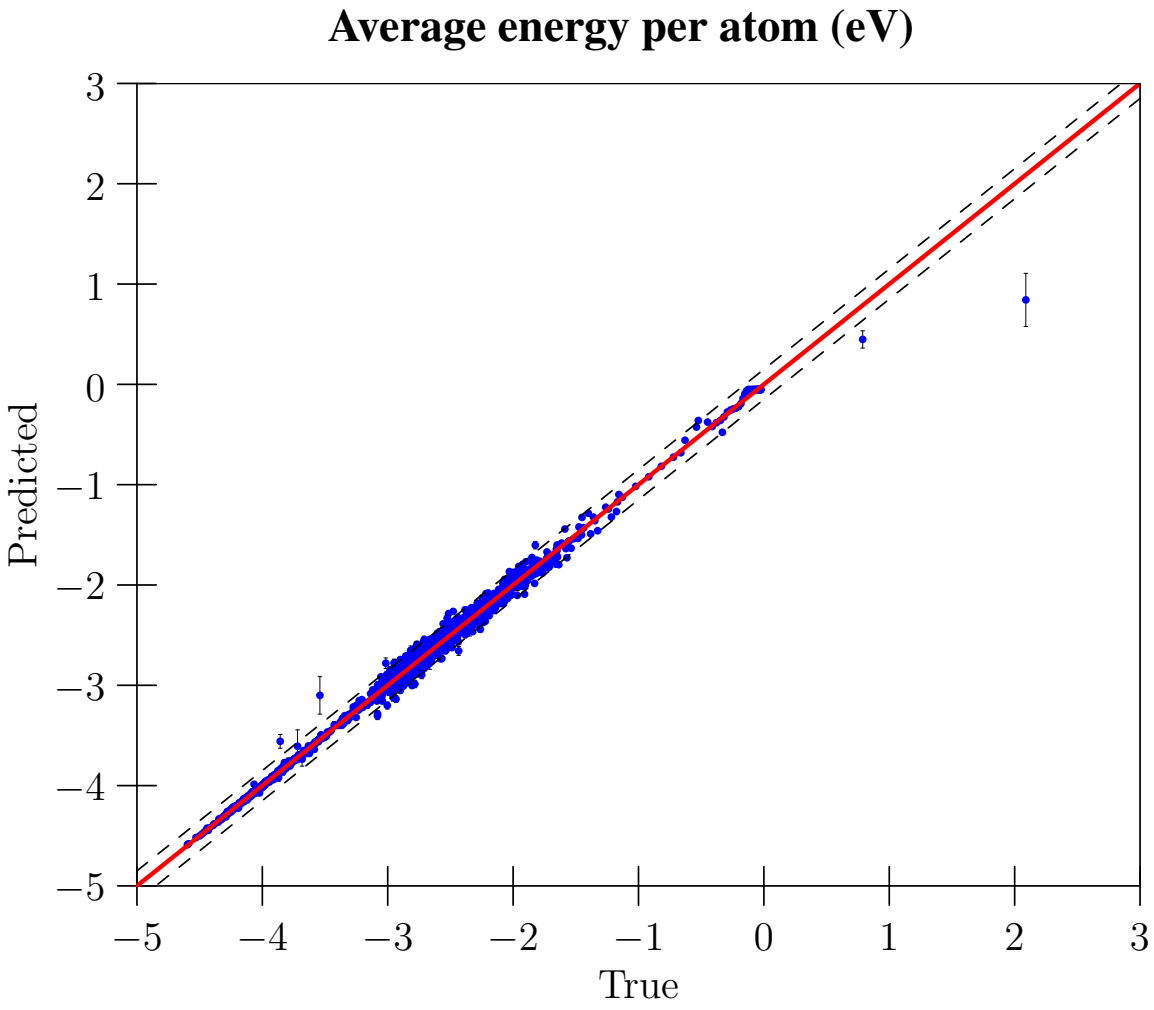


Figure 6.63: Combined results of the five-fold cross-validation carried out on the first-principles total energies. The vertical bars over each data point represent 95% confidence intervals of the predictions. Perfectly accurate predictions would fall on the diagonal indicated by the red line, while the dashed lines indicate $\pm$ 0.15 eV from the diagonal.

cluded 95% confidence intervals (which were calculated using (6.2)). At a given step of the cross-validation, test set configurations which happen to fall somewhat far away from the training set configurations in terms of the GPR covariance function (and, thus, amount to an extrapolation) may not necessarily possess correct predictions, but should be expected to be sufficiently accurate in most cases that their 95% confidence interval contains the correct value of average energy. Albeit there are several outliers, the results show that nearly all of the predictions are within 0.15 eV of the corresponding true energy and, while this investigation is not exhaustive, it supports the conclusion that the first-principles total energies can be interpolated to reasonable accuracy using our regression model.[24] Figure 6.64

[24]The error bounds of 0.15 eV obtained in this cross-validation appear to conform to the same level of

shows the combined results of the second cross-validation carried out using this set of folds, where the total energy errors of the eight EPs of Table 6.1 play the role of the objectives. Although these quantities span a larger range of values than the first-principles total energies, similar accuracy may be observed.

We now consider a cross-validation procedure which uses the same objective values (first-principles total energies and total energy errors of EPs) as the previous cross-validation, but a completely different set of folds. Because none of the atomic configurations in our training set were found to contain environments belonging to more than one of the groups denoted I-IV in Figures 6.32 – 6.34, we defined the folds of this second, independent cross-validation to correspond to these groups, while omitting the nanostructures. That is, the first subdivision contained all of the β-Sn, bcc, fcc, hcp, sc, and sh configurations, the second subdivision contained all of the bc8, diamond, and hexagonal configurations, etc. This is depicted graphically in Figure 6.65 in a manner similar to Figure 6.62, where we have referred to each fold as a "Structure Fold" to differentiate it from the folds of the previous cross-validation. If the MDS embedding of the atomic environments presented in this chapter is assumed to be meaningful, the minimal overlap of groups I-IV then implies that each fold of the cross-validation should amount to using a test set of atomic configurations for which all predictions made require significant extrapolation away from the atomic environments contained in the corresponding training set configurations of that fold. As noted at the beginning of this chapter, the regression methods used within mathematical potentials, including GPR, are nontransferable. Therefore, the results of this latter set of cross-validations should be expected to contain significant inaccuracies. Figures 6.66 and 6.67 show the results of the cross-validation for the first-principles total energies and the total energy errors of the EPs, respectively. Although structure folds 1, 2, and 4 are predominantly consistent with the above hypothesis, structure fold 3 conveys completely accurate extrapolations to the perturbed graphite configurations. Closer examination reveals that there are accurate predictions made in structure folds 1 and 2, as well, implying that there is greater overlap of groups I-III (in the sense of the similarity used in the regression) than is indicated by our MDS embedding of the atomic environments in Figures 6.32 – 6.34. To the contrary, the results of the cross-validation for the total energy errors of the EPs of our study indicate that little extrapolation can be done to high accuracy.

accuracy reported by Lorenz *et al.* [215] (0.16 eV), who used a neural network potential to study H_2 dissociation on the (100) surface of palladium. However, it is worth noting that their potential contained over 3,000 parameters, whereas the GPR method we have exercised here contained only eight.

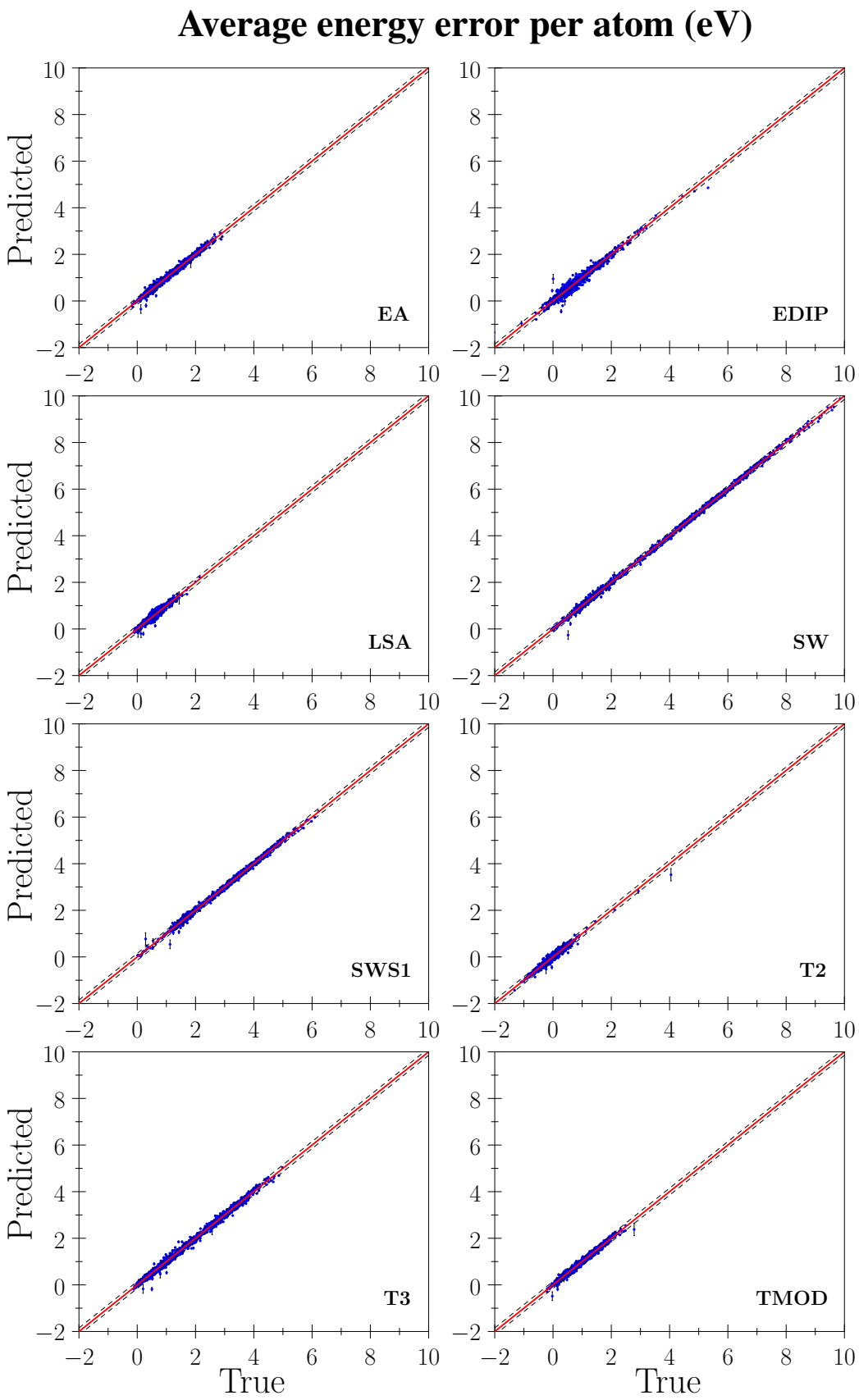


Figure 6.64: Combined results of the five-fold cross-validation carried out on the total energy errors of each EP.

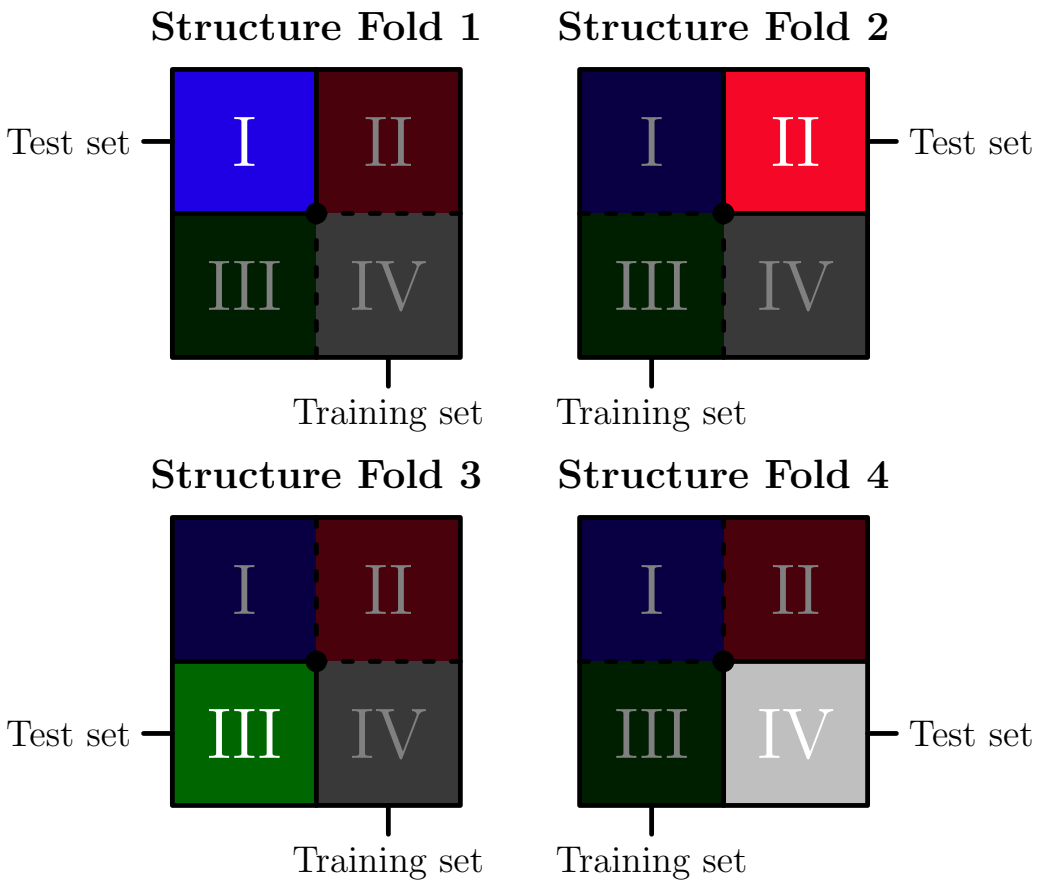


Figure 6.65: Overview of the folds used in the second cross-validation.

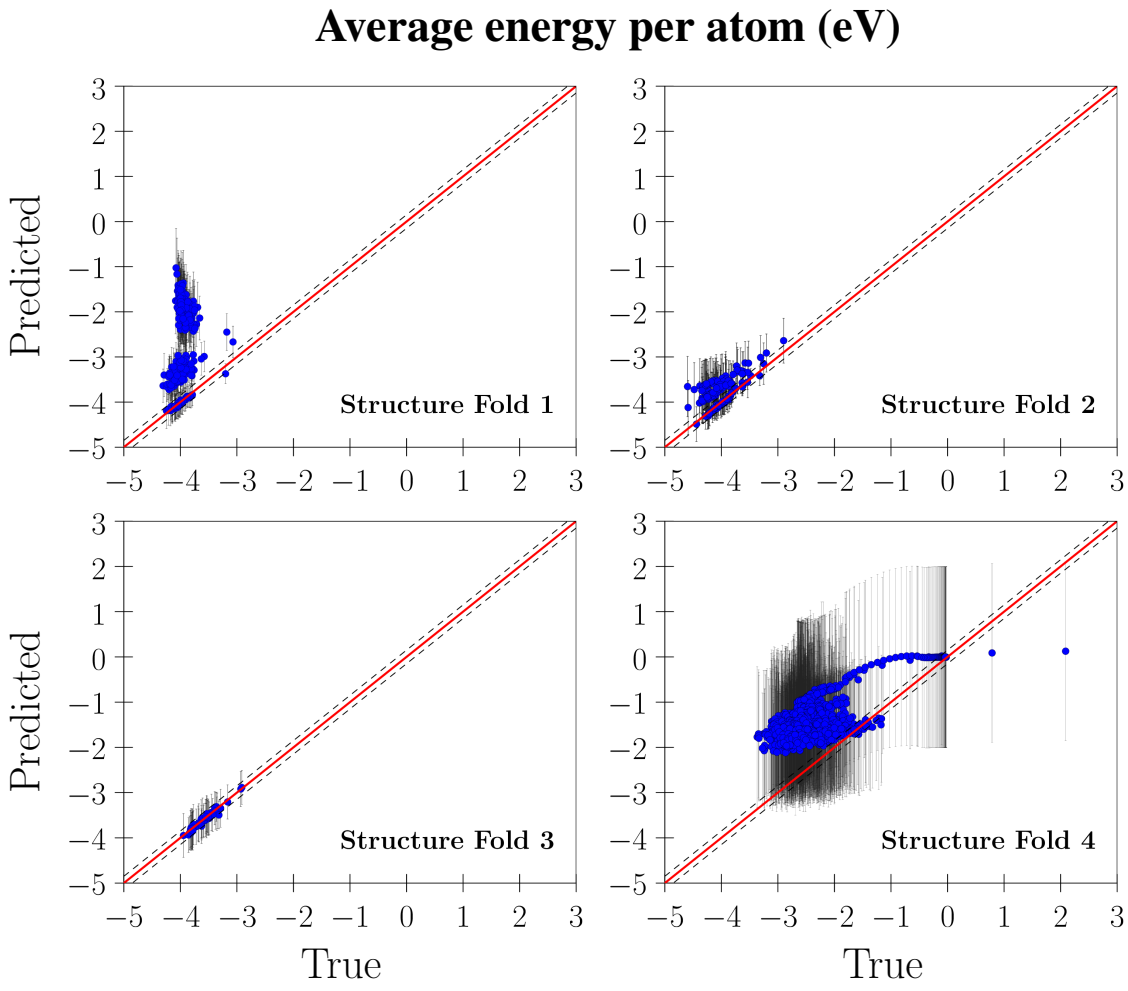


Figure 6.66: Results of the four-fold cross-validation carried out on the first-principles total energies.

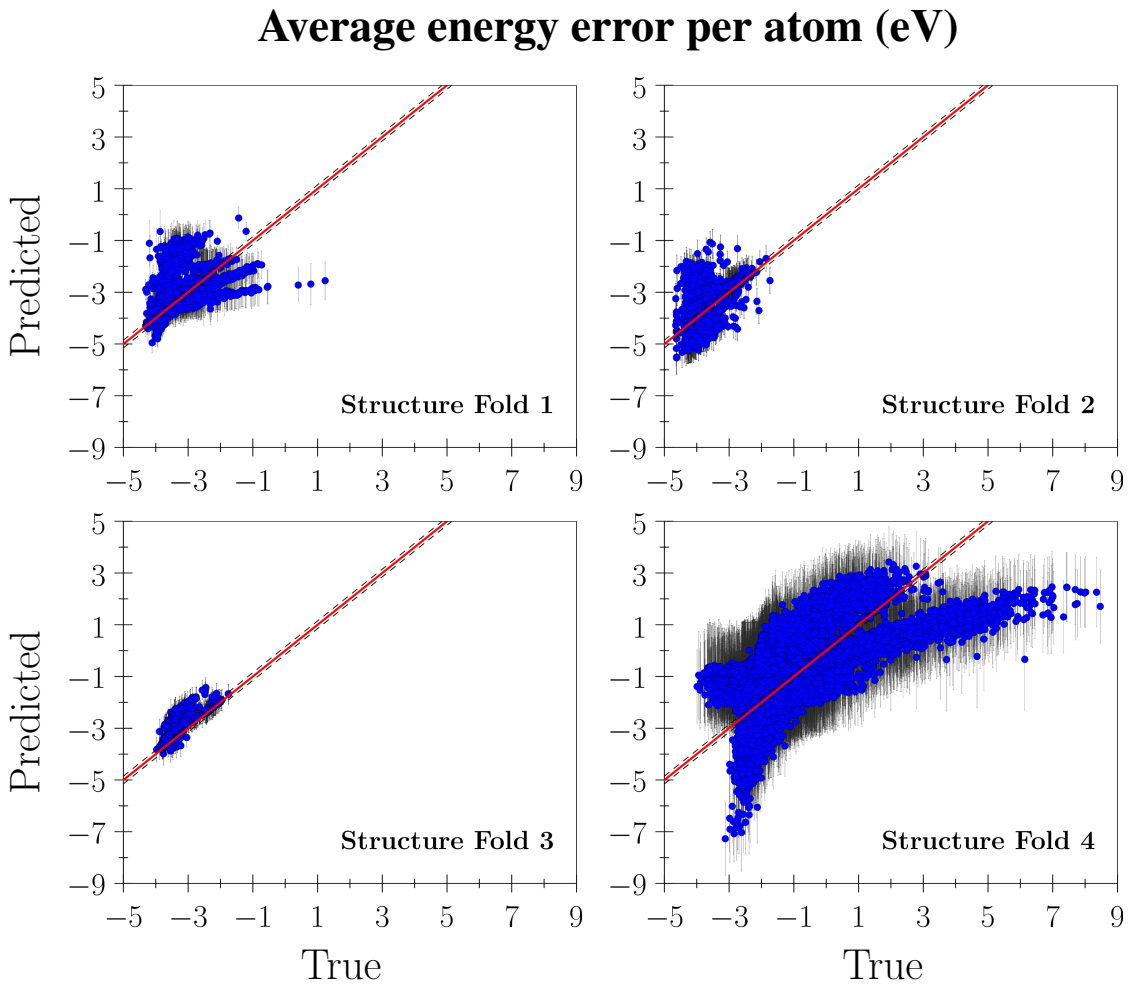


Figure 6.67: Results of the four-fold cross-validation carried out on the total energy errors of each EP.

Chapter 7

Conclusion and Future Work

The understanding of materials has become an essential component of contemporary physics and engineering. Because of the wide-ranging applications of the many materials which are utilized in modern technologies, various models of their behavior at length scales spanning both the macroscopic and nanoscopic domains have been developed, and remain an active topic of scientific research. However, the lack of availability of experimental observations at the nanoscale, coupled with rapid progress in fields such as microprocessor design and biotechnology, have accelerated the need for accurate and efficient atomistic models, in particular. To this end, the focus of the present work has been the creation of a method which can be used to predict the transferability of such models to material behavior which they were not fitted to reproduce.

In Chapter 2, a background of the mechanics involved in atomistic modeling was given which began with the introduction of the Schrödinger equation, which governs the quantum mechanical manner in which all atoms behave. Due to the enormous complexity inherent to solving this equation for all but the most trivial cases, we were led to the Born–Oppenheimer approximation, which admitted considerable simplification by separating its spatial and temporal dependence. Leveraging this approximation, a further simplification is made possible by replacing the many-electron wave function with a surrogate quantity: the electron density. Unfortunately, the family of methods which utilize the latter approach to reach approximate solutions to the Schrödinger equation, known collectively as density functional theory, are still computationally prohibitive with regard to the simulation of dynamical processes which occur over appreciable time and length scales in the majority of applications. Accordingly, empirical potentials (EPs) were introduced as a practical means of encompassing a greater breadth of material behavior. Rather than explicitly including the

electronic degrees of freedom, empirical potentials posit that the internal potential energy of a system of atoms can be written solely in terms of the corresponding nuclear positions, and an expansion is performed which decomposes the energy into contributions of increasing complexity. The simplest potentials write the energy as a function of only the pairwise distances between the atoms, while more sophisticated potentials include dependence on bond angles and higher-order terms. This graduation in complexity was formalized by Carlsson, who provided a taxonomy of empirical potentials which was reviewed at the conclusion of Chapter 2. Finally, it was emphasized that although EPs are always written in terms of local energies which may be associated with individual atoms, these quantities are never uniquely defined.

Chapter 3 introduced the Knowledgebase of Interatomic Models (KIM), a project aimed at ameliorating many of the difficulties which currently exist in the design and use of empirical potentials by providing public access to a repository of standardized implementations of potentials (Models), simulations which use them to compute material properties (Tests), and first-principles calculations (Reference Data). The first problem solved by KIM is the limited availability and portability of empirical potentials. In many cases, an EP used in producing the results of a publication is not made publically available in any format, either as an independent code or as part of a larger software package, leaving prospective users to construct their own implementation; this also oftentimes leads multiple research groups to invest significant effort in implementing the same potential. Next, because the concept of a Model in the KIM framework is abstract, empirical potentials can be written in a cross-language format compatible with numerous simulation environments. This acts to facilitate the creation of Tests in KIM by enabling the use of a range of molecular dynamics software, which is particularly important because of the role KIM Tests play in obviating the irreproducibility of results obtained using EPs. Because KIM is a centralized repository within which each Model, Test, etc. is catalogued with a specific version number, every result stored in KIM can be traced back to its origins through an explicit revision history and recreated. This data provenance moreover implies that all changes which are made to an EP, such as optimizations and bug fixes, are simultaneously made available to all of its users. The final topic explored in Chapter 3 concerned the unique applications enabled by KIM. Because KIM gives rise to a multitude of data in the form of Test Results, the results of a single Test can be compared across numerous Models, exposing trends which might exist between different classes of potentials. Furthermore, by complementing a Model with an ensemble of values of its parameters, its precision can be gauged. Finally, comparing the results of a single Model across an array of Tests for which corresponding Reference Data

is available permits an investigation of its transferability to various material properties.

Chapter 4 was devoted to the subject of representations of atomic environments, from which all empirical potentials are formulated. There, we presented two general categories of representations, real-space descriptors and spectral descriptors, which take different approaches to satisfying the translational, permutational, and rotational invariance which is required to be useful in the context of EPs. Real-space representations are written in terms of the bond lengths and bond angles of an atomic environment, and thus trivially satisfy translational and rotational invariance. However, the most primitive real-space descriptors, such as the Weyl matrix, are not invariant with respect to permutation of the indices used to refer to each of the neighbors of the target atom. Thus, any measure of the similarity between two environments which are represented by these descriptors must explicitly consider all possible indicial permutations. The next category of real-space descriptors encountered in Chapter 4 were the Behler–Parrinello symmetry functions and the Angular Fourier Series. These descriptors were distinguished from the Weyl matrix by the fact that they are additive, i.e. they are calculated by performing summations over each neighbor present in the environment. Because this (finite) summation is permutationally invariant, measures of environment similarity using these descriptors may be obtained by simply taking the normed difference of the descriptor vectors corresponding to each environment. We next considered the spectral representations in greater detail. While these representations are based on the notion of additivity in the sense that they define an atomic neighborhood density function, they establish rotational invariance by appealing to specific characteristics of their respective bases. Although we saw that there was no obvious way of constructing rotational invariants from the geometric moments of the neighborhood density, it was shown that they could be equivalently derived by transforming the basis used to express the density to one with the symmetry of SO(3). In this transformed basis, the behavior of the coefficients of the density under rotation was revealed to be simple, and rotation invariants such as the power spectrum and bispectrum could be derived with relative ease. We concluded Chapter 4 with the introduction of the SOAP kernel, which began by defining an overlap function between two given atomic neighborhood densities ρ and ρ'. It was demonstrated that, for densities which consist of Gaussians placed on each atom, the overlap between ρ and ρ' could be integrated over all rotations of the latter analytically and that, moreover, the result of this integration was equivalent to taking the dot product of the power spectrum of each density.

In Chapter 5, we proposed a categorization of empirical potentials into those which are parametric, semiparametric, and nonparametric. The first category, parametric potentials,

are constructed based on physical intuition and have a finite set of parameters which are typically fit to canonical material properties. The process of fitting the parameters of such potentials was revealed to be highly indeterminate in most cases due to the small number of canonical properties used, requiring careful oversight. In order to remedy this situation, we then saw that point observations, i.e. individual points on the energy landscape, could be used to fit potentials. The modest computational expense associated with producing a large number of such quantities using first-principles calculations accordingly enables the use of an expanded variety of optimization techniques to assist in the fitting process. This relatively large volume of fitting data can also be used to take alternative approaches to the problem of constructing the functional forms of an EP. The first such extended approach we reviewed was the class of tabulated EPs, which are subject to an overall functional form but contain tabulated subordinate functions. The precise number of parameters which determine a tabulated EP may vary depending on its specifics, and we thus defined these to be semiparametric. The second example of a semiparametric EP we observed was the neural network potential. Similar to the tabulated EP, neural network EPs are free to adjust the size of their parameter set during fitting, which is carried out primarily using point observations. However, unlike tabulated potentials, neural network EPs do not possess an overall functional form which restricts their variance, and it is because of this that we have classified them as mathematical potentials. Although mathematical potentials are capable of reproducing nearly any training set of data so long as the complementary descriptor used is well-founded, they are incapable of accurately extrapolating outside of their training set, i.e. they are nontransferable by definition. The final type of potential we considered was a nonparametric EP, using the specific example of a Gaussian process regression (GPR) potential. Like neural network potentials, nonparametric potentials are mathematical in nature. However, they are distinguished from the previous categories of potentials in that their parameter set grows directly in proportion to the size of their training set.

The final chapter of our study introduced the Regression Algorithm for Transferability Estimation (RATE). Although the mathematical potentials surveyed in Chapter 5 are non-transferable by definition, we proposed that they could nevertheless be used to ascertain the transferability of non-mathematical models such as parametric and tabulated EPs. This is accomplished by exploiting the ability of a mathematical potential to infer from the total energy errors of an EP for a training set of atomic configurations an atomic energy error function, as has been previously done in some mathematical potentials. This partition can then subsequently be utilized to predict its total energy error for configurations not in the training set. This procedure was demonstrated for eight specific potentials for silicon and

a training set consisting of perturbed bulk configurations, random clusters, unrelaxed surfaces, and nanostructures. We first selected the regression technique and descriptor used in our study. Because of the various computational advantages offered by GPR, we opted to use it in lieu of a neural network. Further, because the atomic environments only enter the GPR predictive equation in the form of similarities, the SOAP kernel was chosen as the effective descriptor. In order to understand the inner workings of the GPR predictive equation employed, we commenced by visualizing the atomic configurations in a manner which reflected their similarity in the regression itself. Defining a set of dissimilarities between configurations based on the SOAP kernel, the multidimensional scaling (MDS) algorithm was used to generate a three-dimensional embedding of the data, where each point represented an atomic configuration. Examination of the relative MDS coordinates of the various atomic configurations in the training set showed that their geometric arrangement was reasonable: the perturbed bulk configurations fell into nearby groups, the cluster configurations comprised another group, and the nanostructure configurations laid in between them. Using the isomap algorithm to further project subsets of the data onto two-dimensional spaces, plotting the coordination and average energy error per atom of each EP over the resulting coordinates revealed smooth variations. We then proceeded to repeat the same visualization procedure for the individual atomic environments. Proceeding under the assumption that the atomic energies learned by the regression from the first-principles total energies were meaningful, our objective was to understand the individual atomic energy errors (or, equivalently, the atomic energies of the EP) learned for each of the eight silicon EPs considered. To this end, we then compared them to a corresponding set of atomic energy errors which we calculated using the atomic energies defined for each of the eight EPs in Appendix C. Despite the fact that the atomic energy errors of an EP are highly non-unique, inspection revealed that in many cases, the values learned by the regression matched our own. Finally, the interpolative and extrapolative abilities of the regression method were investigated, which showed that interpolation on the total energy errors was accurately achieved, while extrapolation was inconsistent as expected.

The foremost prospect for future work in the context of RATE is the incorporation of atomic force and stress observations into the training set. Specifically, the extent to which this would affect the atomic energy errors learned by the regression and the possible qualitative differences that would accordingly manifest are unclear. A further topic of interest is the variability of the atomic energy errors in response to specific changes in the training set, including not only changing the configurations present but the types of observations (energy errors, atomic force errors, etc) associated with specific training instances. Although

RATE currently relies on the assumption that the set of atomic configurations relevant for selecting an EP for use in an application are known, it is foreseeable that it could be used as a tool for identifying statistical correlations between the accuracy of an EP for different configurations and its accuracy in computing specific material properties. Finally, as mentioned in Section 6.2.3, the use of RATE as a corrective supplement to a potential such as GAP is another application which warrants additional research.

Bibliography

[1] J. Wood. The top ten advances in materials science. *Materials Today*, 11(1-2):40 – 45, 2008.

[2] N. Fomina, C. L. McFearin, M. Sermsakdi, J. M. Morachis, and A. Almutairi. Low Power, Biologically Benign NIR Light Triggers Polymer Disassembly. *Macromolecules*, 44(21):8590–8597, 2011, http://dx.doi.org/10.1021/ma201850q.

[3] P. Gratia, A. Magomedov, T. Malinauskas, M. Daskeviciene, A. Abate, S. Ahmad, M. Grätzel, V. Getautis, and M. K. Nazeeruddin. Methoxydiphenylamine-Substituted Carbazole Twin Derivative: An Efficient Hole-Transporting Material for Perovskite Solar Cells. *Angewandte Chemie International Edition*, 2015.

[4] E. Tadmor, R. Miller, and R. Elliott. *Continuum Mechanics and Thermodynamics: From Fundamental Concepts to Governing Equations*. Cambridge University Press, 2012.

[5] E. B. Tadmor and R. E. Miller. *Modeling Materials: Continuum, Atomistic and Multiscale Techniques*. Cambridge University Press, Cambridge, 2011.

[6] D. E. Shaw, R. O. Dror, J. K. Salmon, J. P. Grossman, K. M. Mackenzie, J. A. Bank, C. Young, M. M. Deneroff, B. Batson, K. J. Bowers, E. Chow, M. P. Eastwood, D. J. Ierardi, J. L. Klepeis, J. S. Kuskin, R. H. Larson, K. Lindorff-Larsen, P. Maragakis, M. A. Moraes, S. Piana, Y. Shan, and B. Towles. Millisecond-scale Molecular Dynamics Simulations on Anton. In *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis*, SC '09, pages 39:1–39:11, New York, NY, USA, 2009. ACM.

[7] M. Mendelev, C. Deng, C. Schuh, and D. Srolovitz. Comparison of molecular dynamics simulation methods for the study of grain boundary migration. *Modelling and Simulation in Materials Science and Engineering*, 21(4):045017, 2013.

[8] J. Vaari. Molecular dynamics simulations of vacancy diffusion in chromium(iii) oxide, hematite, magnetite and chromite. *Solid State Ionics*, 270:10 – 17, 2015.

[9] Y. Shibuta, K. Oguchi, and M. Ohno. Million-atom molecular dynamics simulation on spontaneous evolution of anisotropy in solid nucleus during solidification of iron. *Scripta Materialia*, 86:20 – 23, 2014.

[10] E. Tadmor, R. Elliott, J. Sethna, R. Miller, and C. A. Becker. Knowledgebase of Interatomic Models (KIM). 2011.

[11] E. Tadmor, R. Elliott, J. Sethna, R. Miller, and C. Becker. The Potential of Atomistic Simulations and the Knowledgebase of Interatomic Models. *JOM*, 63:17–17, 2011.

[12] E. B. Tadmor, R. S. Elliott, S. R. Phillpot, and S. B. Sinnott. NSF cyberinfrastructures: A new paradigm for advancing materials simulation. *Current Opinion in Solid State and Materials Science*, 17(6):298 – 304, 2013.

[13] https://wiki.fysik.dtu.dk/asap.

[14] Bahn, S.R. and Jacobsen, Karsten W. An object-oriented scripting interface to a legacy electronic structure code. *Computing in Science Engineering*, 4(3):56–66, May 2002.

[15] I. T. Todorov, W. Smith, K. Trachenko, and M. T. Dove. DL_POLY_3: new dimensions in molecular dynamics simulations via massive parallelism. *J. Mater. Chem.*, 16:1911–1918, 2006.

[16] J. D. Gale. GULP: A computer program for the symmetry-adapted simulation of solids. *J. Chem. Soc., Faraday Trans.*, 93:629–637, 1997.

[17] J. Stone, J. Gullingsrud, P. Grayson, and K. Schulten. A System for Interactive Molecular Dynamics Simulation. In J. F. Hughes and C. H. Séquin, editors, *2001 ACM Symposium on Interactive 3D Graphics*, pages 191–194, New York, 2001. ACM SIGGRAPH.

[18] S. Plimpton. Fast Parallel Algorithms for Short-Range Molecular Dynamics. *Journal of Computational Physics*, 117(1):1 – 19, 1995.

[19] A. P. Bartók et al. libAtoms+QUIP: A software library for carrying out molecular dynamics simulations.

[20] C. A. Becker, F. Tavazza, Z. T. Trautt, and R. A. B. d. Macedo. Considerations for choosing and using force fields and interatomic potentials in materials science and engineering . *Current Opinion in Solid State and Materials Science*, 17(6):277 – 283, 2013.

[21] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. a. Persson. The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013.

[22] G. Bertin. *Dynamics of Galaxies*. Cambridge University Press, 2014.

[23] J. Evans and A. Thorndike. *Quantum Mechanics at the Crossroads: New Perspectives from History, Philosophy and Physics*. The Frontiers Collection. Springer Berlin Heidelberg, 2006.

[24] J. v. Neumann. *Mathematische Grundlagen der Quantenmechanik*. Springer, 1932.

[25] M. Schlosshauer. Decoherence, the measurement problem, and interpretations of quantum mechanics. *Rev. Mod. Phys.*, 76:1267–1305, Feb 2005.

[26] P. Holland. *The Quantum Theory of Motion: An Account of the de Broglie-Bohm Causal Interpretation of Quantum Mechanics*. Cambridge University Press, 1995.

[27] D. Bohm. A Suggested Interpretation of the Quantum Theory in Terms of "Hidden" Variables. I. *Phys. Rev.*, 85:166–179, Jan 1952.

[28] D. Bohm. A Suggested Interpretation of the Quantum Theory in Terms of "Hidden" Variables. II. *Phys. Rev.*, 85:180–193, Jan 1952.

[29] J. Bell and A. Aspect. *Speakable and Unspeakable in Quantum Mechanics: Collected Papers on Quantum Philosophy*. Cambridge University Press, 2004.

[30] J. W. Bush. Pilot-Wave Hydrodynamics. *Annual Review of Fluid Mechanics*, 47(1):269–292, 2015.

[31] R. Hughes. *The Structure and Interpretation of Quantum Mechanics*. Harvard University Press, 1992.

[32] R. Healey. *The Philosophy of Quantum Mechanics: An Interactive Interpretation*. Interactive Interpretation. Cambridge University Press, 1991.

[33] R. Penrose. *The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics*. Popular Science Series. OUP Oxford, 1999.

[34] R. Shankar. *Principles of Quantum Mechanics*. Springer US, 2012.

[35] M. Razavy. *Heisenberg's Quantum Mechanics*. Heisenberg's Quantum Mechanics. World Scientific, 2011.

[36] A. Messiah. *Quantum Mechanics*. Dover Books on Physics. Dover Publications, 2014.

[37] M. Cafiero, S. Bubin, and L. Adamowicz. Non-Born-Oppenheimer calculations of atoms and molecules. *Phys. Chem. Chem. Phys.*, 5:1491–1501, 2003.

[38] S. Bubin, M. Pavanello, W.-C. Tung, K. L. Sharkey, and L. Adamowicz. Born-Oppenheimer and Non-Born-Oppenheimer, Atomic and Molecular Calculations with Explicitly Correlated Gaussians. *Chemical Reviews*, 113(1):36–79, 2013.

[39] G. Arfken and H. Weber. *Mathematical Methods for Physicists*. Elsevier, 2005.

[40] P. Hohenberg and W. Kohn. Inhomogeneous Electron Gas. *Phys. Rev.*, 136:B864–B871, Nov 1964.

[41] A. Carlsson. Beyond Pair Potentials in Elemental Transition Metals and Semiconductors. volume 43 of *Solid State Physics*, pages 1 – 91. Academic Press, 1990.

[42] M. Finnis. Interatomic forces in materials. *Progress in Materials Science*, 49(1):1 – 18, 2004.

[43] F. Stillinger and T. Weber. Computer simulation of local order in condensed phases of silicon. *Phys. Rev. B*, 31:5262, 1985.

[44] X. Zhang, H. Xie, M. Hu, H. Bao, S. Yue, G. Qin, and G. Su. Thermal conductivity of silicene calculated using an optimized Stillinger-Weber potential. *Phys. Rev. B*, 89:054310, Feb 2014.

[45] E. B. Tadmor, R. S. Elliott, J. P. Sethna, C. A. Becker, and R. E. Miller. KIM Requirements Document, 2011.

[46] M. S. Daw and M. I. Baskes. Embedded-Atom Method: Derivation and Application to Impurities, Surfaces, and Other Defects in Metals. *Phys. Rev. B*, 29:6443–6453, 1984.

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

[48] M. S. Daw. Model of metallic cohesion: The embedded-atom method. *Phys. Rev. B*, 39:7441–7452, Apr 1989.

[49] R. S. Elliott. Efficient "universal" truncated Lennard-Jones model for all KIM API supported species. `https://openkim.org/cite/MO_826355984548_001`.

[50] J. E. Jones. On the Determination of Molecular Fields. I. From the Variation of the Viscosity of a Gas with Temperature. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 106(738):441–462, 1924.

[51] J. E. Jones. On the Determination of Molecular Fields. II. From the Equation of State of a Gas. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 106(738):463–477, 1924.

[52] J. E. Lennard-Jones. On the Forces between Atoms and Ions. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 109(752):584–597, 1925.

[53] R. S. Elliott. Third universal Cu potential of Foiles, Baskes, and Daw; obtained from LAMMPS.

[54] S. M. Foiles, M. I. Baskes, and M. S. Daw. Embedded-atom-method functions for the fcc metals Cu, Ag, Au, Ni, Pd, Pt, and their alloys. *Phys. Rev. B*, 33:7983–7991, Jun 1986.

[55] A. Akerson and R. Elliott. Private communication, 2015.

[56] S. L. Frederiksen, K. W. Jacobsen, K. S. Brown, and J. P. Sethna. Bayesian Ensemble Approach to Error Estimation of Interatomic Potentials. *Phys. Rev. Lett.*, 93:165501, Oct 2004.

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

[58] A.-L. Cauchy. Mémoire sur les systèmes isotropes de points matériels. In *Oeuvres complètes*, volume 2, pages 351–386. Cambridge University Press, 2009. Cambridge Books Online.

[59] C. Truesdell and W. Noll. The Non-Linear Field Theories of Mechanics. In *The Non-Linear Field Theories of Mechanics / Die Nicht-Linearen Feldtheorien der Mechanik*, volume 2 / 3 / 3 of *Encyclopedia of Physics / Handbuch der Physik*, pages 1–541. Springer Berlin Heidelberg, 1965.

[60] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. v. Lilienfeld. Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning. *Phys. Rev. Lett.*, 108:058301, Jan 2012.

[61] H. Weyl. *The Classical Groups: Their Invariants and Representations*. Princeton mathematical series. Princeton University Press, 1939.

[62] A. P. Bartók, R. Kondor, and G. Csányi. On representing chemical environments. *Phys. Rev. B*, 87:184115, May 2013.

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

[64] C.-H. Lo and H.-S. Don. 3D Moment Forms: Their Construction and Application to Object Identification and Positioning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 11(10):1053–1064, October 1989.

[65] M.-K. Hu. Visual pattern recognition by moment invariants. *Information Theory, IRE Transactions on*, 8(2):179–187, February 1962.

[66] T. Reiss. The revised fundamental theorem of moment invariants. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(8):830–834, Aug 1991.

[67] J. Flusser and T. Suk. Pattern recognition by affine moment invariants. *Pattern Recognition*, 26(1):167 – 174, 1993.

[68] J. Flusser and T. Suk. Pattern recognition by means of affine moment invariants. Research Report 1726, Institute of Information Theory and Automation, 1991.

[69] A. Mamistvalov. n-dimensional moment invariants and conceptual mathematical theory of recognition n-dimensional solids. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(8):819–831, Aug 1998.

[70] E. Elliott. *Algebra of Quantics*. American Mathematical Society, 1895.

[71] F. A. Sadjadi and E. L. Hall. Three-Dimensional Moment Invariants. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-2(2):127–136, March 1980.

[72] T. H. Reiss. Object recognition using algebraic and differential invariants. *Signal Processing*, 32(3):367 – 395, 1993.

[73] M. Bôcher and E. Duval. *Introduction to Higher Algebra*. Macmillan, 1907.

[74] G. Salmon. *Lessons Introductory to the Modern Higher Algebra*. Hodges, Smith, 1866.

[75] L. Dickson. *Algebraic Invariants*. Cornell University Library historical math monographs. J. Wiley & Sons, Incorporated, 1914.

[76] R. Courant and D. Hilbert. *Methods of Mathematical Physics*, volume 1 of *Methods of Mathematical Physics*. Wiley, 2008.

[77] W. Thompson. *Angular Momentum*. Wiley, 2008.

[78] W. Byerly. *An Elementary Treatise on Fourier's Series: and Spherical, Cylindrical, and Ellipsoidal Harmonics, with Applications to Problems in Mathematical*. Dover Books on Mathematics. Dover Publications, 2014.

[79] N. Ferrers. *An Elementary Treatise on Spherical Harmonics and Subjects Connected with Them*. Macmillan and Company, 1877.

[80] E. O. Steinborn and K. Ruedenberg. *Rotation and translation of regular and irregular solid spherical harmonics*, volume 7, pages 1–81. Elsevier Science, 1973.

[81] M. J. Caola. Solid harmonics and their addition theorems. *Journal of Physics A: Mathematical and General*, 11(2):L23, 1978.

[82] A. Edmonds. *Angular Momentum in Quantum Mechanics*. Investigations in Physics. Princeton University Press, 1996.

[83] T. Özdoğan and M. Orbay. Cartesian Expressions for Surface and Regular Solid Spherical Harmonics Using Binomial Coefficients and Its Use in the Evaluation of Multicenter Integrals. *Czechoslovak Journal of Physics*, 52(12):1297–1302, 2002.

[84] K.-I. Kanatani. Distribution of directional data and fabric tensors. *International Journal of Engineering Science*, 22(2):149 – 164, 1984.

[85] R. Kakarala, P. Kaliamoorthi, and W. Li. Viewpoint invariants from three-dimensional data: The role of reflection in human activity understanding. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pages 57–62, June 2011.

[86] R. Kakarala and D. Mao. A theory of phase-sensitive rotation invariance with spherical harmonic and moment-based representations. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 105–112, June 2010.

[87] M. Rose. *Elementary Theory of Angular Momentum*. Dover books on physics and chemistry. Dover, 1995.

[88] D. Varshalovich, A. Moskalev, and V. Khersonskii. *Quantum Theory of Angular Momentum: Irreducible Tensors, Spherical Harmonics, Vector Coupling Coefficients, 3nj Symbols*. WorldScientificPub., 1988.

[89] P. Delaney and D. Walsh. A bibliography of higher-order spectra and cumulants. *Signal Processing Magazine, IEEE*, 11(3):61–70, July 1994.

[90] R. Kakarala. The Bispectrum as a Source of Phase-Sensitive Invariants for Fourier Descriptors: A Group-Theoretic Approach. *Journal of Mathematical Imaging and Vision*, 44(3):341–353, 2012.

[91] P. J. Steinhardt, D. R. Nelson, and M. Ronchetti. Bond-orientational order in liquids and glasses. *Phys. Rev. B*, 28:784–805, Jul 1983.

[92] C. D. Taylor. Connections between the energy functional and interaction potentials for materials simulations. *Phys. Rev. B*, 80:024104, Jul 2009.

[93] K. Kaufmann and W. Baumeister. Single-centre expansion of Gaussian basis functions and the angular decomposition of their overlap integrals. *Journal of Physics B: Atomic, Molecular and Optical Physics*, 22(1):1, 1989.

[94] G. Arfken, H. Weber, and F. Harris. *Mathematical Methods for Physicists: A Comprehensive Guide*. Elsevier, 2012.

[95] R. Carbó, L. Leyda, and M. Arnau. How similar is a molecule to another? An electron density measure of similarity between two molecular structures. *International Journal of Quantum Chemistry*, 17(6):1185–1189, 1980.

[96] W. J. Szlachta. *First principles interatomic potential for tungsten based on Gaussian process regression*. Ph.D. thesis, University of Cambridge, March 2014.

[97] W. J. Szlachta, A. P. Bartók, and G. Csányi. Accuracy and transferability of Gaussian approximation potential models for tungsten. *Phys. Rev. B*, 90:104108, Sep 2014.

[98] E. Alpaydin. *Introduction to Machine Learning*. The MIT Press, 2014.

[99] D. J. C. MacKay. *Information Theory, Inference & Learning Algorithms*. Cambridge University Press, New York, NY, USA, 2002.

[100] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[101] S. Theodoridis and K. Koutroumbas. *Pattern Recognition, Fourth Edition*. Academic Press, 4th edition, 2008.

[102] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition, 2009.

[103] M. J. Stott and E. Zaremba. Quasiatoms: An approach to atoms in nonuniform electronic systems. *Phys. Rev. B*, 22:1564–1583, Aug 1980.

[104] J. K. Nørskov and N. D. Lang. Effective-medium theory of chemical binding: Application to chemisorption. *Phys. Rev. B*, 21:2131–2136, Mar 1980.

[105] K. W. Jacobsen, J. K. Nørskov, and M. J. Puska. Interatomic interactions in the effective-medium theory. *Phys. Rev. B*, 35:7423–7442, May 1987.

[106] K. Jacobsen, P. Stoltze, and J. Nørskov. A semi-empirical effective medium theory for metals and alloys. *Surface Science*, 366(2):394 – 402, 1996.

[107] T. J. Raeker and A. E. Depristo. Theory of chemical bonding based on the atom-homogeneous electron gas system. *International Reviews in Physical Chemistry*, 10(1):1–54, 1991.

[108] M. S. Daw and M. I. Baskes. Semiempirical, Quantum Mechanical Calculation of Hydrogen Embrittlement in Metals. *Phys. Rev. Lett.*, 50:1285–1288, Apr 1983.

[109] M. W. Finnis and J. E. Sinclair. A simple empirical N-body potential for transition metals. *Philosophical Magazine A*, 50(1):45–55, 1984.

[110] G. C. Abell. Empirical chemical pseudopotential theory of molecular and metallic bonding. *Phys. Rev. B*, 31:6184–6196, May 1985.

[111] F. Ercolessi, E. Tosatti, and M. Parrinello. Au (100) Surface Reconstruction. *Phys. Rev. Lett.*, 57:719–722, Aug 1986.

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

[113] M. I. Baskes and S. G. Srinivasan. The embedded atom method ansatz: validation and violation. *Modelling and Simulation in Materials Science and Engineering*, 22(2):025025, 2014.

[114] D. Pettifor. *Bonding and Structure of Molecules and Solids*. Oxford science publications. Clarendon Press, 1995.

[115] D. W. Brenner, O. A. Shenderova, and D. A. Areshkin. *Quantum-Based Analytic Interatomic Forces and Materials Simulation*, pages 207–239. John Wiley & Sons, Inc., 2007.

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

[117] G. Ackland, A. Sutton, and V. Vitek, editors. *Special issue: 25 years of Finnis-Sinclair potentials and related issues*, volume 89 (34-36). 2009.

[118] S. M. Foiles and M. I. Baskes. Contributions of the embedded-atom method to materials science and engineering. *MRS Bulletin*, 37:485–491, 5 2012.

[119] A. Voter. The embedded atom method. In J. Westbrook and R. Fleischer, editors, *Intermetallic Compounds: Principles and Practice*, pages 77–90. John Wiley and Sons, Ltd, London, 1994.

[120] M. I. Baskes. Application of the Embedded-Atom Method to Covalent Materials: A Semiempirical Potential for Silicon. *Phys. Rev. Lett.*, 59:2666–2669, Dec 1987.

[121] M. I. Baskes, J. S. Nelson, and A. F. Wright. Semiempirical modified embedded-atom potentials for silicon and germanium. *Phys. Rev. B*, 40:6085–6100, Sep 1989.

[122] M. I. Baskes. Modified embedded-atom potentials for cubic materials and impurities. *Phys. Rev. B*, 46:2727–2742, Aug 1992.

[123] D. W. Brenner. Relationship between the embedded-atom method and Tersoff potentials. *Phys. Rev. Lett.*, 63:1022–1022, Aug 1989.

[124] J. Tersoff. New empirical approach for the structure and energy of covalent systems. *Phys. Rev. B*, 37:6991, 1988.

[125] P. M. Morse. Diatomic Molecules According to the Wave Mechanics. II. Vibrational Levels. *Phys. Rev.*, 34:57–64, Jul 1929.

[126] J. H. Rose, J. Ferrante, and J. R. Smith. Universal Binding Energy Curves for Metals and Bimetallic Interfaces. *Phys. Rev. Lett.*, 47:675–678, Aug 1981.

[127] J. Ferrante, J. R. Smith, and J. H. Rose. Diatomic Molecules and Metallic Adhesion, Cohesion, and Chemisorption: A Single Binding-Energy Relation. *Phys. Rev. Lett.*, 50:1385–1386, May 1983.

[128] J. H. Rose, J. R. Smith, F. Guinea, and J. Ferrante. Universal features of the equation of state of metals. *Phys. Rev. B*, 29:2963–2969, Mar 1984.

[129] A. Banerjea and J. R. Smith. Origins of the universal binding-energy relation. *Phys. Rev. B*, 37:6632–6645, Apr 1988.

[130] Y. Mishin, M. J. Mehl, D. A. Papaconstantopoulos, A. F. Voter, and J. D. Kress. Structural stability and lattice defects in copper: *Ab initio*, tight-binding, and embedded-atom calculations. *Phys. Rev. B*, 63:224106, May 2001.

[131] M. T. Yin and M. L. Cohen. Microscopic Theory of the Phase Transformation and Lattice Dynamics of Si. *Phys. Rev. Lett.*, 45:1004–1007, Sep 1980.

[132] M. T. Yin and M. L. Cohen. Theory of static structural properties, crystal stability, and phase transformations: Application to Si and Ge. *Phys. Rev. B*, 26:5668–5687, Nov 1982.

[133] M. T. Yin and M. L. Cohen. Structural theory of graphite and graphitic silicon. *Phys. Rev. B*, 29:6996–6998, Jun 1984.

[134] K. J. Chang and M. L. Cohen. Structural and electronic properties of the high-pressure hexagonal phases of Si. *Phys. Rev. B*, 30:5376–5378, Nov 1984.

[135] A. P. Horsfield, A. M. Bratkovsky, M. Fearn, D. G. Pettifor, and M. Aoki. Bond-order potentials: Theory and implementation. *Phys. Rev. B*, 53:12694–12712, May 1996.

[136] D. G. Pettifor and I. I. Oleinik. Analytic bond-order potentials beyond Tersoff-Brenner. I. Theory. *Phys. Rev. B*, 59:8487–8499, Apr 1999.

[137] I. I. Oleinik and D. G. Pettifor. Analytic bond-order potentials beyond Tersoff-Brenner. II. Application to the hydrocarbons. *Phys. Rev. B*, 59:8500–8507, Apr 1999.

[138] D. G. Pettifor and I. I. Oleinik. Bounded Analytic Bond-Order Potentials for σ and π Bonds. *Phys. Rev. Lett.*, 84:4124–4127, May 2000.

[139] D. Pettifor and I. Oleynik. Interatomic bond-order potentials and structural prediction. *Progress in Materials Science*, 49(3-4):285 – 312, 2004. A Festschrift in Honor of T. B. Massalski.

[140] D. W. Brenner. Empirical potential for hydrocarbons for use in simulating the chemical vapor deposition of diamond films. *Phys. Rev. B*, 42:9458–9471, Nov 1990.

[141] D. Brenner. The Art and Science of an Analytic Potential. *Physica Status Solidi B*, 217(1):23–40, 2000.

[142] M. Finnis. Bond-order potentials through the ages. *Progress in Materials Science*, 52(23):133 – 153, 2007. Modelling electrons and atoms for materials science.

[143] J. D. Kress and A. F. Voter. Low-order moment expansions to tight binding for interatomic potentials: Successes and failures. *Phys. Rev. B*, 52:8766–8775, Sep 1995.

[144] J. Tersoff. Empirical Interatomic Potential for Silicon with Improved Elastic Properties. *Phys. Rev. B*, 38(14):9902–9905, 1988.

[145] A. Singh. A three-body Stillinger-Weber (SW) Model (Parameterization) for Silicon Optimized for Silicene. `https://openkim.org/cite/MO_800412945727_001`.

[146] M. Z. Bazant and E. Kaxiras. Modeling of Covalent Bonding in Solids by Inversion of Cohesive Energy Curves. *Phys. Rev. Lett.*, 77:4370–4373, Nov 1996.

[147] P. Erhart and K. Albe. Analytical potential for atomistic simulations of silicon, carbon, and silicon carbide. *Phys. Rev. B*, 71:035211, Jan 2005.

[148] F. Gao, R. L. Johnston, and J. N. Murrell. Empirical many-body potential energy functions for iron. *The Journal of Physical Chemistry*, 97(46):12073–12082, 1993, http://dx.doi.org/10.1021/j100148a038.

[149] J. B. Sturgeon and B. B. Laird. Adjusting the melting point of a model system via Gibbs-Duhem integration: Application to a model of aluminum. *Phys. Rev. B*, 62:14720–14727, Dec 2000.

[150] M. G. Martin and J. I. Siepmann. Transferable Potentials for Phase Equilibria. 1. United-Atom Description of n-Alkanes. *The Journal of Physical Chemistry B*, 102(14):2569–2577, 1998, http://dx.doi.org/10.1021/jp972543+.

[151] G. Bonny, R. Pasianot, and L. Malerba. Fitting interatomic potentials consistent with thermodynamics: Fe, Cu, Ni and their alloys. *Philosophical Magazine*, 89(34-36):3451–3464, 2009.

[152] Y. G. Xu and G. R. Liu. Fitting interatomic potentials using molecular dynamics simulations and inter-generation projection genetic algorithm. *Journal of Micromechanics and Microengineering*, 13(2):254, 2003.

[153] C. Voglis, P. Hadjidoukas, D. Papageorgiou, and I. Lagaris. A parallel hybrid optimization algorithm for fitting interatomic potentials. *Applied Soft Computing*, 13(12):4481 – 4492, 2013.

[154] J. A. Martinez, D. E. Yilmaz, T. Liang, S. B. Sinnott, and S. R. Phillpot. Fitting empirical potentials: Challenges and methodologies. *Current Opinion in Solid State and Materials Science*, 17(6):263 – 270, 2013.

[155] F. Ercolessi and J. B. Adams. Interatomic potentials from first-principles calculations - the force-matching method. *Europhysics Letters*, 26(8):583–588, 1994.

[156] M. Masia, E. Guàrdia, and P. Nicolini. The force matching approach to multiscale simulations: Merits, shortcomings, and future perspectives. *International Journal of Quantum Chemistry*, 114(16):1036–1040, 2014.

[157] X.-Y. Liu, J. B. Adams, F. Ercolessi, and J. A. Moriarty. EAM potential for magnesium from quantum mechanical forces. *Modelling and Simulation in Materials Science and Engineering*, 4(3):293, 1996.

[158] Y. Li, D. J. Siegel, J. B. Adams, and X.-Y. Liu. Embedded-atom-method tantalum potential developed by the force-matching method. *Phys. Rev. B*, 67:125101, Mar 2003.

[159] Y. Umeno, T. Kitamura, K. Date, M. Hayashi, and T. Iwasaki. Optimization of interatomic potential for Si/SiO2 system based on force matching. *Computational Materials Science*, 25(3):447 – 456, 2002.

[160] Y. Saito, N. Sasaki, H. Moriya, A. Kagatsume, and S. Noro. Parameter Optimization of Tersoff Interatomic Potentials Using a Genetic Algorithm. *JSME international journal. Series A, Solid mechanics and material engineering*, 44(2):207–213, apr 2001.

[161] Y. Lee and G. S. Hwang. Force-matching-based parameterization of the Stillinger-Weber potential for thermal conduction in silicon. *Phys. Rev. B*, 85:125204, Mar 2012.

[162] D. González and S. Davis. Fitting of interatomic potentials without forces: A parallel particle swarm optimization algorithm. *Computer Physics Communications*, 185(12):3090–3093, 2014.

[163] S.-G. Kim, M. F. Horstemeyer, M. I. Baskes, M. Rais-Rohani, S. Kim, B. Jelinek, J. Houze, A. Moitra, and L. Liyanage. Semi-Empirical Potential Methods for Atomistic Simulations of Metals and Their Construction Procedures. *Journal of Engineering Materials and Technology*, 131(4):041210–041210, Sep 2009.

[164] P. Brommer and F. Gähler. Effective potentials for quasicrystals from ab-initio data. *Philosophical Magazine*, 86(6-8):753–758, 2006, http://dx.doi.org/10.1080/14786430500333349.

[165] P. Brommer and F. Gähler. Potfit: effective potentials from ab initio data. *Modelling and Simulation in Materials Science and Engineering*, 15(3):295, 2007.

[166] P. Brommer, A. Kiselev, D. Schopf, P. Beck, J. Roth, and H.-R. Trebin. Classical interaction potentials for diverse materials from ab initio data: a review of potfit. *Modelling and Simulation in Materials Science and Engineering*, 23(7):074002, 2015.

[167] D. Schopf, P. Brommer, B. Frigan, and H.-R. Trebin. Embedded atom method potentials for Al-Pd-Mn phases. *Phys. Rev. B*, 85:054201, Feb 2012.

[168] A. Slepoy, M. D. Peters, and A. P. Thompson. Searching for globally optimal functional forms for interatomic potentials using genetic programming with parallel tempering. *Journal of Computational Chemistry*, 28(15):2465–2471, 2007.

[169] W. M. Brown, A. P. Thompson, and P. A. Schultz. Efficient hybrid evolutionary optimization of interatomic potential models. *The Journal of Chemical Physics*, 132(2), 2010.

[170] M. A. Bellucci and D. F. Coker. Empirical valence bond models for reactive potential energy surfaces: A parallel multilevel genetic program approach. *The Journal of Chemical Physics*, 135(4), 2011.

[171] P. Prenter. *Splines and Variational Methods*. Dover Books on Mathematics Series. Dover Publications, 2008.

[172] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 2nd edition, 1992.

[173] M. J. D. Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal*, 7(2):155–162, 1964, http://comjnl.oxfordjournals.org/content/7/2/155.full.pdf+html.

[174] M. J. D. Powell. A Method for Minimizing a Sum of Squares of Non-Linear Functions Without Calculating Derivatives. *The Computer Journal*, 7(4):303–307, 1965, http://comjnl.oxfordjournals.org/content/7/4/303.full.pdf+html.

[175] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, 1983, http://science.sciencemag.org/content/220/4598/671.full.pdf.

[176] S. Kirkpatrick. Optimization by simulated annealing: Quantitative studies. *Journal of Statistical Physics*, 34(5):975–986.

[177] R. Storn and K. Price. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11(4):341–359, December 1997.

[178] K. Price, R. M. Storn, and J. A. Lampinen. *Differential Evolution: A Practical Approach to Global Optimization*. Natural Computing Series. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[179] T. Lenosky, B. Sadigh, E. Alonso, V. Bulatov, T. d. l. Rubia, J. Kim, A. Voter, and J. Kress. Highly optimized empirical potential model of silicon. *Modelling and Simulation in Materials Science and Engineering*, 8(6):825, 2000.

[180] M. Wen, S. Whalen, R. Elliott, and E. Tadmor. Interpolation effects in tabulated interatomic potentials. *Modelling and Simulation in Materials Science and Engineering*, 23(7):074008, 2015.

[181] D. Wolff and W. G. Rudd. Tabulated potentials in molecular dynamics simulations. *Computer Physics Communications*, 120(1):20 – 32, 1999.

[182] C. M. Bishop. Neural networks and their applications. *Review of Scientific Instruments*, 65(6):1803–1832, January 1994.

[183] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Inc., New York, NY, USA, 1995.

[184] F. Rosenblatt. The Perceptron–a perceiving and recognizing automaton. Technical Report 85-460-1, Cornell Aeronautical Laboratory, 1957.

[185] Elsevier Pergamon, New York. *Neural Networks*, 1988-2015.

[186] Institute of Electrical and Electronics Engineers, New York. *IEEE transactions on neural networks*, 1990-2015.

[187] J. Behler. Representing potential energy surfaces by high-dimensional neural network potentials. *Journal of Physics: Condensed Matter*, 26(18):183001, 2014.

[188] M. Malshe, R. Narulkar, L. M. Raff, M. Hagan, S. Bukkapatnam, and R. Komanduri. Parametrization of analytic interatomic potential functions using neural networks. *The Journal of Chemical Physics*, 129(4), 2008.

[189] S. Bukkapatnam, M. Malshe, P. M. Agrawal, L. M. Raff, and R. Komanduri. Parametrization of interatomic potential functions using a genetic algorithm accelerated with a neural network. *Phys. Rev. B*, 74:224102, Dec 2006.

[190] L. Raff, R. Komanduri, M. Hagan, and S. Bukkapatnam. *Neural Networks in Chemical Reaction Dynamics*. Oxford University Press, USA, 2012.

[191] G. Montavon, K. Hansen, S. Fazli, M. Rupp, F. Biegler, A. Ziehe, A. Tkatchenko, A. V. Lilienfeld, and K.-R. Müller. Learning Invariant Representations of Molecules for Atomization Energy Prediction. In *Advances in Neural Information Processing Systems 25*, pages 449–457. 2012.

[192] G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, and O. A. v. Lilienfeld. Machine learning of molecular electronic properties in chemical compound space. *New Journal of Physics*, 15(9):095003, 2013.

[193] M. Rupp, M. R. Bauer, R. Wilcken, A. Lange, M. Reutlinger, F. M. Boeckler, and G. Schneider. Machine learning estimates of natural product conformational energies. *PLoS Comput Biol*, 10(1):e1003400, 01 2014.

[194] R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. v. Lilienfeld. Big Data Meets Quantum Chemistry Approximations: The Δ-Machine Learning Approach. *Journal of Chemical Theory and Computation*, 11(5):2087–2096, 2015.

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

[196] Handley, Christopher Michael and Behler, Jörg. Next generation interatomic potentials for condensed systems. *The European Physical Journal B*, 87(7), 2014.

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

[198] A. P. Thompson, P. A. Schultz, P. Crozier, S. G. Moore, L. P. Swiler, J. A. Stephens, S. M. Foiles, and G. J. Tucker. Automated Algorithms for Quantum-Level Accuracy in Atomistic Simulations: LDRD Final Report. Technical report, Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States), 2014.

[199] H. Cartwright. *Development and Uses of Artificial Intelligence in Chemistry*, pages 349–390. John Wiley & Sons, Inc., 2007.

[200] D. A. R. S. Latino, R. P. S. Fartaria, F. F. M. Freitas, J. a. Aires-De-Sousa, and F. M. S. Silva Fernandes. Approach to potential energy surfaces by neural networks. A review of recent work. *International Journal of Quantum Chemistry*, 110(2):432–445, 2010.

[201] C. M. Handley and P. L. A. Popelier. Potential Energy Surfaces Fitted by Artificial Neural Networks. *The Journal of Physical Chemistry A*, 114(10):3371–3383, 2010, http://dx.doi.org/10.1021/jp9105585. PMID: 20131763.

[202] J. Behler. Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations. *Phys. Chem. Chem. Phys.*, 13:17930–17955, 2011.

[203] A. Pukrittayakamee, M. Malshe, M. Hagan, L. M. 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*, 130(13), 2009.

[204] S. Curteanu and H. Cartwright. Neural networks applied in chemistry. I. Determination of the optimal topology of multilayer perceptron neural networks. *Journal of Chemometrics*, 25(10):527–549, 2011.

[205] H. Cartwright and S. Curteanu. Neural Networks Applied in Chemistry. II. Neuro-Evolutionary Techniques in Process Modeling and Optimization. *Industrial & Engineering Chemistry Research*, 52(36):12673–12688, 2013.

[206] A. Skinner and J. Broughton. Neural networks in computational materials science: training algorithms. *Modelling and Simulation in Materials Science and Engineering*, 3(3):371, 1995.

[207] L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Applications of mathematics : stochastic modelling and applied probability. Springer, 1996.

[208] U. Grenander. *Abstract Inference*. Probability and Statistics Series. John Wiley & Sons, 1981.

[209] C.-R. H. Stuart Geman. Nonparametric maximum likelihood estimation by the method of sieves. *The Annals of Statistics*, 10(2):401–414, 1982.

[210] H. White and J. Wooldridge. Some results on sieve estimation with dependent observations. In W. Barnett, J. Powell, and G. Tauchen, editors, *Nonparametric and Semiparametric Methods in Econometrics and Statistics: Proceedings of the Fifth International Symposium in Economic Theory and Econometrics*, International Symposia in Economic Theory and Econometrics, pages 459–493. Cambridge University Press, New York, 1991.

[211] S. Geman, E. Bienenstock, and R. Doursat. Neural Networks and the Bias/Variance Dilemma. *Neural Comput.*, 4(1):1–58, January 1992.

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

[213] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. v. Lilienfeld, A. Tkatchenko, and K.-R. Müller. Assessment and Validation of Machine Learning Methods for Predicting Molecular Atomization En-

ergies. *Journal of Chemical Theory and Computation*, 9(8):3404–3419, 2013, http://dx.doi.org/10.1021/ct400195d.

[214] J. Li, B. Jiang, and H. Guo. Permutation invariant polynomial neural network approach to fitting potential energy surfaces. ii. four-atom systems. *The Journal of Chemical Physics*, 139(20), 2013.

[215] S. Lorenz, M. Scheffler, and A. Gross. Descriptions of surface chemical reactions using a neural network representation of the potential-energy surface. *Phys. Rev. B*, 73:115431, Mar 2006.

[216] T. Morawietz and J. Behler. A full-dimensional neural network potential-energy surface for water clusters up to the hexamer. *Zeitschrift für Physikalische Chemie*, 227:1559–1581, 2013.

[217] H. Eshet. *Unraveling microscopic origins of complex behavior of carbon and sodium with neural-network potentials*. PhD thesis, Eidgenössische Technische Hochschule ETH Zürich, Nr. 19877, 2011, 2011.

[218] J. B. Witkoskie and D. J. Doren. Neural Network Models of Potential Energy Surfaces: Prototypical Examples. *Journal of Chemical Theory and Computation*, 1(1):14–23, 2005, http://dx.doi.org/10.1021/ct049976i.

[219] N. Artrith, B. Hiller, and J. Behler. Neural network potentials for metals and oxides - first applications to copper clusters at zinc oxide. *Physica Status Solidi B*, 250(6):1191–1203, 2013.

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

[221] Haley, P.J. and Soloway, Donald. Extrapolation limitations of multilayer feedforward neural networks. In *Neural Networks, 1992. IJCNN., International Joint Conference on*, volume 4, pages 25–30 vol.4, Jun 1992.

[222] D. H. Wolpert. A Mathematical Theory of Generalization: Part I. *Complex Systems*, 4(2):151–200, 1990.

[223] D. H. Wolpert. A Mathematical Theory of Generalization: Part II. *Complex Systems*, 4(2):201–249, 1990.

[224] D. Wolpert. *The Mathematics of Generalization: The Proceedings of the SFI/CNLS Workshop on Formal Approaches to Supervised Learning*. Proceedings volume in the Santa Fe Institute studies in the sciences of complexity. Addison-Wesley, 1995.

[225] D. Wolpert. The Lack of A Priori Distinctions Between Learning Algorithms. *Neural Computation*, 8(7):1341–1390, Oct 1996.

[226] D. Wolpert. The Existence of A Priori Distinctions Between Learning Algorithms. *Neural Computation*, 8(7):1391–1420, Oct 1996.

[227] D. H. Wolpert. *Soft Computing and Industry: Recent Applications*, chapter The Supervised Learning No-Free-Lunch Theorems, pages 25–42. Springer London, London, 2002.

[228] J. Ischtwan and M. A. Collins. Molecular potential energy surfaces by interpolation. *The Journal of Chemical Physics*, 100(11):8080–8088, 1994.

[229] A. J. Skinner and J. Q. Broughton. Generating optimal structural databases for developing atomistic potentials. *Computational materials science*, 4(1):1–9, May 1995.

[230] L. B. Pártay, A. P. Bartók, and G. Csányi. Efficient sampling of atomic configurational spaces. *The Journal of Physical Chemistry B*, 114(32):10502–10512, 2010. PMID: 20701382.

[231] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[232] K. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. *Neural Networks, IEEE Transactions on*, 12(2):181–201, Mar 2001.

[233] A. J. S. Thomas Hofmann, Bernhard Schölkopf. Kernel Methods in Machine Learning. *The Annals of Statistics*, 36(3):1171–1220, 2008.

[234] C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. Adaptive computation and machine learning. MIT Press, 2006.

[235] O. A. v. Lilienfeld, R. Ramakrishnan, M. Rupp, and A. Knoll. Fourier series of atomic radial distribution functions: A molecular fingerprint for machine learning models of quantum chemical properties. *International Journal of Quantum Chemistry*, 115(16):1084–1093, 2015.

[236] K. T. Schütt, H. Glawe, F. Brockherde, A. Sanna, K. R. Müller, and E. K. U. Gross. How to represent crystal structures for machine learning: Towards fast prediction of electronic properties. *Phys. Rev. B*, 89:205118, May 2014.

[237] F. Faber, A. Lindmaa, O. A. v. Lilienfeld, and R. Armiento. Crystal structure representations for machine learning models of formation energies. *International Journal of Quantum Chemistry*, 115(16):1094–1101, 2015.

[238] V. Botu and R. Ramprasad. Adaptive machine learning framework to accelerate ab initio molecular dynamics. *International Journal of Quantum Chemistry*, 115(16):1074–1083, 2015.

[239] K. Vu, J. C. Snyder, L. Li, M. Rupp, B. F. Chen, T. Khelif, K.-R. Mller, and K. Burke. Understanding kernel ridge regression: Common behaviors from simple functions to density functionals. *International Journal of Quantum Chemistry*, 115(16):1115–1128, 2015.

[240] T. Hollebeek, T.-S. Ho, and H. Rabitz. Constructing multidimensional molecular potential energy surfaces from ab initio data. *Annual Review of Physical Chemistry*, 50(1):537–570, 1999. PMID: 15012421.

[241] A. Bartók-Pártay. *Gaussian Approximation Potential: an interatomic potential derived from first principles Quantum Mechanics*. Ph.D. thesis, University of Cambridge, 2009.

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

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

[244] Z. Li, J. R. Kermode, and A. De Vita. Molecular Dynamics with On-the-Fly Machine Learning of Quantum-Mechanical Forces. *Phys. Rev. Lett.*, 114:096405, Mar 2015.

[245] M. Rupp. Machine learning for quantum mechanics in a nutshell. *International Journal of Quantum Chemistry*, 115(16):1058–1073, 2015.

[246] M. Rupp. Special issue on machine learning and quantum mechanics. *International Journal of Quantum Chemistry*, 115(16):1003–1004, 2015.

[247] G. Kresse and J. Hafner. *Ab initio* molecular dynamics for liquid metals. *Phys. Rev. B*, 47:558–561, Jan 1993.

[248] G. Kresse and J. Hafner. *Ab initio* molecular-dynamics simulation of the liquid-metal-amorphous-semiconductor transition in germanium. *Phys. Rev. B*, 49:14251–14269, May 1994.

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

[250] G. Kresse and J. Furthmüller. Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set. *Phys. Rev. B*, 54:11169–11186, Oct 1996.

[251] J. P. Perdew, K. Burke, and M. Ernzerhof. Generalized Gradient Approximation Made Simple. *Phys. Rev. Lett.*, 77:3865, 1996.

[252] J. P. Perdew, K. Burke, and M. Ernzerhof. Erratum: Generalized Gradient Approximation Made Simple. *Phys. Rev. Lett.*, 78:1396, 1997.

[253] P. E. Blöchl. Projector augmented-wave method. *Phys. Rev. B*, 50:17953, 1994.

[254] G. Kresse and D. Joubert. From ultrasoft pseudopotentials to the projector augmented-wave method. *Phys. Rev. B*, 59:1758, 1999.

[255] N. Kosugi. Modification of the Liu-Davidson method for obtaining one or simultaneously several eigensolutions of a large real-symmetric matrix. *Journal of Computational Physics*, 55(3):426 – 436, 1984.

[256] T. Brink. Tersoff-style three-body potential for SiC by Erhart/Albe, parameter set Si II. `https://openkim.org/cite/MO_408791041969_000`.

[257] T. Brink. Model driver for Tersoff-style potentials ported from LAMMPS. `https://openkim.org/cite/Tersoff_LAMMPS__MD_077075034781_001`.

[258] M. Bazant, E. Kaxiras, and J. Justo. The Environment-Dependent Interatomic Potential applied to silicon disordered structures and phase transitions. *Mat. Res. Soc. Proc.*, 491:339, 1997.

[259] D. S. Karls. Original EDIP potential for elemental silicon. `https://openkim.org/cite/MO_958932894036_001`.

[260] D. S. Karls. A C-based implementation of the EDIP three-body bond-order potential of Bazant and Kaxiras. https://openkim.org/cite/MD_506186535567_001.

[261] H. Balamane, T. Halicioglu, and W. A. Tiller. Comparative study of silicon empirical interatomic potentials. *Phys. Rev. B*, 46:2250–2279, 1992.

[262] A. Singh. A three-body Stillinger-Weber (SW) Model (Parameterization) for Silicon modified by Balamane et al. https://openkim.org/cite/MO_113686039439_001.

[263] A. Singh. A three-body Stillinger-Weber (SW) potential for Silicon. https://openkim.org/cite/MD_335816936951_001.

[264] T. Brink. Tersoff's silicon potential (PRB 37, 1988). https://openkim.org/cite/MO_245095684871_000.

[265] T. Brink. Tersoff's silicon potential (PRB 38, 1988). https://openkim.org/cite/MO_186459956893_000.

[266] T. Kumagai, S. Izumi, S. Hara, and S. Sakai. Development of bond-order potentials that can reproduce the elastic constants and melting point of silicon for classical molecular dynamics simulation. *Computational Materials Science*, 39(2):457 – 464, 2007.

[267] R. M. Neal. *Bayesian Learning for Neural Networks*. PhD thesis, Toronto, Ont., Canada, Canada, 1995. AAINN02676.

[268] C. K. I. Williams. Computation with Infinite Neural Networks. *Neural Comput.*, 10(5):1203–1216, July 1998.

[269] D. J. C. Mackay. Probable networks and plausible predictions – a review of practical Bayesian methods for supervised neural networks. *Network: Computation in Neural Systems*, 6(3):469–505, 1995.

[270] D. J. C. MacKay. Bayesian Non-Linear Modelling with Neural Networks, 1995.

[271] D. M. Titterington. Bayesian Methods for Neural Networks and Related Models. *Statist. Sci.*, 19(1):128–139, 02 2004.

[272] G. Csányi, T. Albaret, M. C. Payne, and A. De Vita. "Learn on the Fly": A Hybrid Classical and Quantum-Mechanical Molecular Dynamics Simulation. *Phys. Rev. Lett.*, 93:175503, Oct 2004.

[273] J. F. Ziegler, J. Biersack, and U. Littmark. The stopping and range of ions in matter, Vol. 1, 1985.

[274] J. B. Tenenbaum, V. d. Silva, and J. C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323, 2000, http://science.sciencemag.org/content/290/5500/2319.full.pdf.

[275] J. d. Leeuw. Applications of convex analysis to multidimensional scaling. In J. R. Barra, F. Brodeau, G. Romier, and B. v. Cutsem, editors, *Recent developments in statistics*, pages 133–145, Amsterdam, The Netherlands, 1977. North Holland Publishing Company.

[276] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[277] I. Borg and P. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer, 2005.

[278] E. Pękalska and R. P. W. Duin. *The Dissimilarity Representation for Pattern Recognition: Foundations And Applications*. World Scientific Publishing Company, River Edge, NJ, USA, December 2005.

[279] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Second Edition*. The MIT Press, 2nd edition, September 2001.

[280] B. A. Gillespie, X. W. Zhou, D. A. Murdick, H. N. G. Wadley, R. Drautz, and D. G. Pettifor. Bond-order potential for silicon. *Phys. Rev. B*, 75:155207, Apr 2007.

[281] H. Xie, F. Yin, T. Yu, J.-T. Wang, and C. Liang. Mechanism for direct graphite-to-diamond phase transition. *Scientific Reports*, 4:5930 EP –, Aug 2014.

[282] L. A. Garvie, P. Németh, and P. R. Buseck. Transformation of graphite to diamond via a topotactic mechanism. *American Mineralogist*, 99(2-3):531–538, 2014, http://ammin.geoscienceworld.org/content/99/2-3/531.full.pdf+html.

[283] R. Z. Khaliullin, H. Eshet, T. D. Kühne, J. Behler, and M. Parrinello. Nucleation mechanism for the direct graphite-to-diamond phase transition. *Nat Mater*, 10(9):693–697, Sep 2011.

[284] K. J. Caspersen, A. Lew, M. Ortiz, and E. A. Carter. Importance of Shear in the bcc-to-hcp Transformation in Iron. *Phys. Rev. Lett.*, 93:115501, Sep 2004.

[285] J. B. Liu and D. D. Johnson. bcc-to-hcp transformation pathways for iron versus hydrostatic pressure: Coupled shuffle and shear modes. *Phys. Rev. B*, 79:134113, Apr 2009.

[286] M. Ekman, B. Sadigh, K. Einarsdotter, and P. Blaha. *Ab initio* study of the martensitic bcc-hcp transformation in iron. *Phys. Rev. B*, 58:5296–5304, Sep 1998.

[287] B. C. Bolding and H. C. Andersen. Interatomic potential for silicon clusters, crystals, and surfaces. *Phys. Rev. B*, 41:10568–10585, May 1990.

[288] B. J. Thijsse. Relationship between the modified embedded-atom method and Stillinger-Weber potentials in calculating the structure of silicon. *Phys. Rev. B*, 65:195207, May 2002.

[289] C. L. Allred, X. Yuan, M. Z. Bazant, and L. W. Hobbs. Elastic constants of defected and amorphous silicon with the environment-dependent interatomic potential. *Phys. Rev. B*, 70:134113, Oct 2004.

Appendix A

Spherical harmonics and related functions

A.1 Spherical harmonics

In Chapter 4, we define the *spherical harmonics* Y_ℓ^m as

$$Y_\ell^m(\theta,\phi) = \sqrt{\frac{2\ell+1}{4\pi}\frac{(\ell-m)!}{(\ell+m)!}}\, P_\ell^m(\cos\theta)e^{im\phi}, \tag{A.1}$$

where $i = \sqrt{-1}$. In the above, the functions P_ℓ^m are the associated Legendre polynomials defined for $m \geq 0$ as

$$P_\ell^m(x) = (-1)^m(1-x^2)^{m/2}\frac{d^m}{dx^m}P_\ell(x), \tag{A.2}$$

where P_ℓ are the (unassociated) Legendre polynomials given by

$$P_\ell(x) = \frac{1}{2^\ell \ell!}\frac{d^\ell}{dx^\ell}\left[(x^2-1)^\ell\right]. \tag{A.3}$$

In order to encompass the associated Legendre polynomials for negative values of m, we make use of the following relation, where m is assumed to be positive:

$$P_\ell^{-m}(x) = (-1)^m\frac{(\ell-m)!}{(\ell+m)!}P_\ell^m(x). \tag{A.4}$$

The definition of (A.1) includes a normalizing factor so that the following orthogonality relation holds:

$$\langle Y_\ell^m, Y_\ell^m \rangle = \int_0^\pi \int_0^{2\pi} \overline{Y_\ell^m(\theta,\phi)} Y_\ell^m(\theta,\phi) \sin\theta d\phi d\theta = \delta_{\ell\ell'}\delta_{mm'}. \tag{A.5}$$

Finally, taking the conjugate of Y_ℓ^m has the following effect:

$$\overline{Y_\ell^m(\theta,\phi)} = Y_\ell^m(\theta,-\phi) = (-1)^m Y_\ell^{-m}(\theta,\phi). \tag{A.6}$$

A.2 Wigner D-matrices

As with the spherical harmonics, our definition follows the convention of Varshalovich et al. [88]. In terms of the Euler angles α, β, γ which correspond to a rotation $\mathcal{R}$, the Wigner D-matrices are defined as

$$D_{mm'}^\ell(\mathcal{R}) = D_{mm'}^\ell(\alpha,\beta,\gamma) = e^{-im\alpha} d_{mm'}^\ell(\beta) e^{-im'\gamma}, \tag{A.7}$$

where the functions $d_{mm'}^\ell$ are given by

$$d_{mm'}^\ell(\beta) = (-1)^{\ell-m'} \sqrt{(\ell+m)!(\ell-m)!(\ell+m')!(\ell-m')!} \tag{A.8}$$

$$\times \sum_k (-1)^k \frac{(\cos\frac{\beta}{2})^{m+m'+2k} (\sin\frac{\beta}{2})^{2\ell-m-m'-2k}}{k!(\ell-m-k)!(\ell-m'-k)!(m+m'+k)!} \tag{A.9}$$

where the summation over k is taken to run over all indices such that all of the factorial arguments in the denominator are non-negative. The Wigner D-matrices obey the following orthogonality relationship:

$$\int_{\mathcal{R}} \overline{D_{m'm}^\ell(\mathcal{R})} D_{\mu'\mu}^{\ell'}(\mathcal{R}) d\mathcal{R} = \frac{8\pi^2}{2\ell+1} \delta_{\ell\ell'}\delta_{m'\mu'}\delta_{m\mu}, \tag{A.10}$$

where the integration takes place over all rotations $\mathcal{R} \in \mathrm{SO}(3)$, while the direct product of two Wigner D-matrices can be expressed as

$$D_{m_1 m_1'}^{\ell_1}(\mathcal{R}) D_{m_2 m_2'}^{\ell_2}(\mathcal{R}) = \sum_{\ell=|\ell_1-\ell_2|}^{\ell_1+\ell_2} \sum_{m,m'} C_{\ell_1 m_1 \ell_2 m_2}^{\ell m} D_{mm'}^\ell(\mathcal{R}) C_{\ell_1 m_1' \ell_2 m_2'}^{\ell m'}, \tag{A.11}$$

where explicit forms for the Clebsch–Gordan coefficients $C^{\ell m}_{\ell_1 m_1 \ell_2 m_2}$ can be found in [88, Ch. 8]. In direct notation, we write (A.11) as

$$\mathbf{D}^{\ell_1}(\mathcal{R}) \otimes \mathbf{D}^{\ell_2}(\mathcal{R}) = (\mathbf{C}^{\ell_1,\ell_2})^\dagger \left[\bigoplus_{\ell=|\ell_1-\ell_2|}^{\ell_1+\ell_2} \mathbf{D}^{\ell}(\mathcal{R}) \right] \mathbf{C}^{\ell_1,\ell_2}, \tag{A.12}$$

as is similarly done in [62]. Finally, we note that $C^{\ell m}_{\ell_1 m_1 \ell_2 m_2} = 0$ unless $m_1 + m_2 = m$.

Appendix B

Calculation of SOAP coefficients

In this appendix, we address the question of how to calculate the coefficients $c_{n\ell m}$ used in the SOAP similarity kernel of Section 4.3.1. Although these quantities are defined for a density $\rho(\boldsymbol{r})$ to be equal to the L^2 inner product $\langle g_n(r)Y(\hat{\boldsymbol{r}}), \rho(\boldsymbol{r})\rangle$, there is no obvious way to evaluate the associated integrals. However, we may equate the expressions for $\rho(\boldsymbol{r})$ in (4.87) and (4.98):

$$\sum_{\beta=0}^{N_{\text{neigh}}} \sum_{\ell=0}^{\ell_{\max}} \sum_{m=-\ell}^{\ell} c_{\ell m}^{\beta}(r) Y_{\ell}^{m}(\hat{\boldsymbol{r}}) f_{\text{cut}}(r_\beta) = \sum_{n=1}^{\infty} \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} c_{n\ell m} g_n(r) Y_{\ell}^{m}(\hat{\boldsymbol{r}}). \tag{B.1}$$

Rearranging the (finite) summations, we have

$$\sum_{\ell=0}^{\ell_{\max}} \sum_{m=-\ell}^{\ell} \left(\sum_{\beta=0}^{N_{\text{neigh}}} c_{\ell m}^{\beta}(r) f_{\text{cut}}(r_\beta) \right) Y_{\ell}^{m}(\hat{\boldsymbol{r}}) = \sum_{\ell=0}^{\ell_{\max}} \sum_{m=-\ell}^{\ell} \left(\sum_{n=1}^{n_{\max}} c_{n\ell m} g_n(r) \right) Y_{\ell}^{m}(\hat{\boldsymbol{r}}), \tag{B.2}$$

and we attempt to determine $c_{n\ell m}$ so as to match the terms in parentheses for a given ℓ and m:

$$\sum_{\beta=0}^{N_{\text{neigh}}} c_{\ell m}^{\beta}(r) f_{\text{cut}}(r_\beta) \approx \sum_{n=1}^{n_{\max}} c_{n\ell m} g_n(r). \tag{B.3}$$

In order to satisfy this relation, we sample the expression on the left at a set of distances $\boldsymbol{r}^{\text{rad}} = (r_1^{\text{rad}}, r_2^{\text{rad}}, \ldots, r_{n_{\max}}^{\text{rad}})$ which extend out to a cutoff distance $r_{\text{cut}}^{\text{rad}}$, so that

$$r_n^{\text{rad}} = \frac{n-1}{n_{\max}} r_{\text{cut}}^{\text{rad}}. \tag{B.4}$$

As in [97], we choose the radial basis functions $g_n(r)$ as an orthonormalized set of Gaussians which reside at each of the distances in $\boldsymbol{r}^{\mathrm{rad}}$. We begin with a basis of unnormalized Gaussians $\phi_n(r)$ defined by

$$\phi_n(r) \triangleq \exp\left(\frac{-(r - r_n^{\mathrm{rad}})^2}{2\sigma_{\mathrm{rad}}^2}\right), \tag{B.5}$$

which often referred to as a "radial basis function network." With regard to this work, we have found that taking

$$\sigma_{\mathrm{rad}} = \frac{r_{n_{\max}}^{\mathrm{rad}}}{4\sqrt{2n_{\max}}} \tag{B.6}$$

produces a reasonable radial basis for the approximation in (B.3), and this rule has been used for all calculations presented here. In light of (4.107), the overlap matrix $\mathbf{S}^{\mathrm{rad}}$ is defined by

$$[\mathbf{S}^{\mathrm{rad}}]_{nn'} \triangleq \int_0^\infty \phi_n(r)\phi_{n'}(r)r^2 dr, \tag{B.7}$$

and the orthonormalized functions $g_n(r)$ can then be obtained from the functions ϕ_n by

$$g_n(r) = \sum_{n'=1}^{n_{\max}} [\mathbf{U}^{-1}]_{nn'}\phi_{n'}(r), \tag{B.8}$$

where $\mathbf{U}$ is the upper Cholesky factor of $\mathbf{S}^{\mathrm{rad}}$, i.e. $\mathbf{S}^{\mathrm{rad}} = \mathbf{U}^T\mathbf{U}$. Collecting the sampled values of the left-hand side of (B.3) for each $\ell = 0, \ldots, \ell_{\max}$ and $m = -\ell, \ldots, \ell$ into corresponding vectors

$$\mathbf{y}_{\ell m}^{\mathrm{rad}} \triangleq \left[\sum_{\beta=0}^{N_{\mathrm{neigh}}} c_{\ell m}^{\beta}(r_1^{\mathrm{rad}})f_{\mathrm{cut}}(r_\beta), \ldots, \sum_{\beta=0}^{N_{\mathrm{neigh}}} c_{\ell m}^{\beta}(r_{n_{\max}}^{\mathrm{rad}})f_{\mathrm{cut}}(r_\beta)\right]^T, \tag{B.9}$$

the coefficients $c_{n\ell m}$ are now determined with the aid of *kernel ridge regression*, a method discussed in Section 5.3.1. In this case, using kernel ridge regression allows for a smooth interpolation between the values contained in $\mathbf{y}_{\ell m}^{\mathrm{rad}}$. Defining the matrix K^{rad} by

$$[K^{\mathrm{rad}}]_{nn'} \triangleq \phi_n(r_{n'}^{\mathrm{rad}}), \tag{B.10}$$

the coefficients $c_{n\ell m}$ are given for a given ℓ and m by the expression

$$\mathbf{c}_{\ell m} = (K^{\mathrm{rad}})^{-1}\mathbf{y}_{\ell m}^{\mathrm{rad}}, \tag{B.11}$$

where we have defined for each ℓ and m a vector

$$\mathbf{c}_{\ell m} = \left[c_{1\ell m}, c_{2\ell m}, \ldots, c_{n_{\max}\ell m}\right]^T . \tag{B.12}$$

Appendix C

Empirical potentials for silicon

Below, we list the eight silicon EPs which are used in the case study of Chapter 6. Literary references are provided for each EP and, for those of them which are published as Models in the OpenKIM repository, we provide the corresponding citation. The notation used for the EP parameters in Sections C.1 – C.8 should be considered local in scope, i.e. it is inconsistent with the notation used in the body of this work. The interested reader may note that the relationship between the MEAM form adopted in the LSA potential and the form of the Stillinger–Weber potential is exposed in [288], while the relationship between the EAM framework and the functional form of Tersoff is given in [123].

C.1 Erhart–Albe (EA)

Literary reference: [147]
OpenKIM references: [256, 257]
Functional form: In order to facilitate comparison with the T2, T3, and TMOD potentials, we have reformulated the functional form of this EP from its original publication so that it more closely matches that given in the T2 section of this appendix.

$$\mathcal{V} = \frac{1}{2}\sum_{i}\sum_{j\neq i} V_{ij}$$

$$V_{ij} = f_c(r_{ij})\left[f_R(r_{ij}) - b_{ij} f_A(r_{ij})\right]$$

$$f_R(r) = A\exp\left[-\lambda_1(r - r_0)\right]$$

$$f_A(r) = B\exp\left[-\lambda_2(r - r_0)\right].$$

$$f_c(r) = \begin{cases} 1 & r \leq R_1 \\ \frac{1}{2}\left[1 + \cos\left(\pi \frac{r - R_1}{R_2 - R_1}\right)\right] & R_1 < r < R_2 \\ 0 & r \geq R_2 \end{cases}$$

$$b_{ij} = (1 + \zeta_{ij}^{\eta})^{-\delta}$$

$$\zeta_{ij} = \sum_{k \neq i,j} f_c(r_{ik}) g(\theta_{ijk}) \exp\left[m(r_{ij} - r_{ik})^n\right]$$

$$g(\theta) = c_1 + g_o(\theta) g_a(\theta)$$

$$g_o(\theta) = \frac{c_2 (h - \cos\theta)^2}{c_3 + (h - \cos\theta)^2}$$

$$g_a(\theta) = 1 + c_4 \exp[-c_5 (h - \cos\theta)^2]$$

Parameters:

$A = 5.684210526315789\,\text{eV}$ $B = 8.92421052631579\,\text{eV}$

$\lambda_1 = 2.615478663648396\,\text{Å}^{-1}$ $\lambda_2 = 1.6659\,\text{Å}^{-1}$

$r_0 = 2.222\,\text{Å}$ $\eta = 1$

$\eta \times \delta = 0.5$ $m = 0.0\,\text{Å}^{-1}$

$n = 1$ $c_1 = 0.09253$

$c_2 = 0.29752325606639246$ $c_3 = 0.4019179609$

$c_4 = 0$ $c_5 = 0$

$h = -0.335$ $R_1 = 2.75\,\text{Å}$

$R_2 = 3.05\,\text{Å}$

C.2 Environment-Dependent Interatomic Potential (EDIP)

Literary reference: [147]

OpenKIM references: [259, 260]

Functional form:

$$\mathcal{V} = \sum_i \sum_{j \neq i} V_2(r_{ij}, Z_i) + \sum_i \sum_{j \neq i} \sum_{\substack{k \neq i \\ k > j}} V_3(\mathbf{r}_{ij}, \mathbf{r}_{ik}, Z_i)$$

$$Z_i = \sum_{m \neq i} f_{c,Z}(r_{im})$$

$$f_{c,Z}(r) = \begin{cases} 1 & r \leq c \\ \exp\left(\dfrac{\alpha}{1-x^{-3}}\right) & c < r < a \\ 0 & r \geq a \end{cases}$$

$$x = \frac{(r-c)}{(a-c)}$$

$$V_2(r, Z) = A\left[\left(\frac{B}{r}\right)^{\rho} - p(Z)\right] f_{c,2}(r)$$

$$f_{c,2}(r) = \begin{cases} \exp\left(\dfrac{\sigma}{r-a} + \dfrac{\sigma}{a}\right) & r < a \\ 0 & r \geq a \end{cases}$$

$$p(Z) = \exp\left(-\beta Z^2\right)$$

$$V_3(\mathbf{r}_{ij}, \mathbf{r}_{ik}, Z_i) = C\lambda f_{c,3}(r_{ij}) f_{c,3}(r_{ik}) g_Z(\theta_{ijk}, Z_i)$$

$$\theta_{ijk} = \arccos\left(\frac{\mathbf{r}_{ij} \cdot \mathbf{r}_{ik}}{r_{ij} r_{ik}}\right)$$

$$f_{c,3}(r) = \begin{cases} \exp\left(\dfrac{\gamma}{r-a} + \dfrac{\gamma}{a}\right) & r < a \\ 0 & r \geq a \end{cases}$$

$$g_Z(\theta, Z) = \left[\left(1 - e^{-Q(Z)(\cos\theta + \tau(Z))^2}\right) + \eta Q(Z)(\cos\theta + \tau(Z))^2\right]$$

$$Q(Z) = Q_0 \exp(-\mu Z)$$

$$\tau(Z) = u_1 + u_2\left(u_3 e^{-u_4 Z} - e^{-2u_4 Z}\right)$$

Parameters: In order to identify two-body and three-body cutoff functions $f_{c,2}(r)$ and $f_{c,3}(r)$ which are equal to unity at $r = 0$, we have incorporated the appropriate normalization factors into the parameter $A = \exp(-\sigma/a) A_{\text{orig}}$ and an additional parameter $C = \exp(-2\gamma/a)$.

$A = 6.63411353973\,\text{eV}$	$B = 1.5075463\,\text{Å}$	$C = 0.486410247273$
$\rho = 1.2085196$	$\sigma = 0.5774108\,\text{Å}$	$\lambda = 1.4533108\,\text{eV}$
$\gamma = 1.1247945\,\text{Å}$	$\mu = 0.6966326$	$Q_0 = 312.1341346$
$\eta = 0.2523244$	$\beta = 0.0070975$	$\alpha = 3.1083847$
$u_1 = -0.165799$	$u_2 = 32.557$	$u_3 = 0.286198$
$u_4 = 0.66$	$a = 3.1213820\,\text{Å}$	$c = 2.5609104\,\text{Å}$

C.3 LSA

Literary reference: [179]

OpenKIM references: None

Functional form: This EP is based on the MEAM formalism:

$$\mathcal{V} = \sum_i \sum_{j>i} \phi(r_{ij}) + \sum_i U\left[\underbrace{\sum_{j\neq i} \varrho(r_{ij}) + \sum_{\substack{j\neq i \\ k>j}} f_c(r_{ij}) f_c(r_{ik}) g(\cos\theta_{jk})}_{\varrho^{\mathrm{CF}}}\right] - \sum_i U[0].$$

The spline functions $\phi(r)$, $\varrho(r)$, $f_c(r)$, $U(x)$, and $g(\cos\theta)$ are pictured in Figure C.1 below.

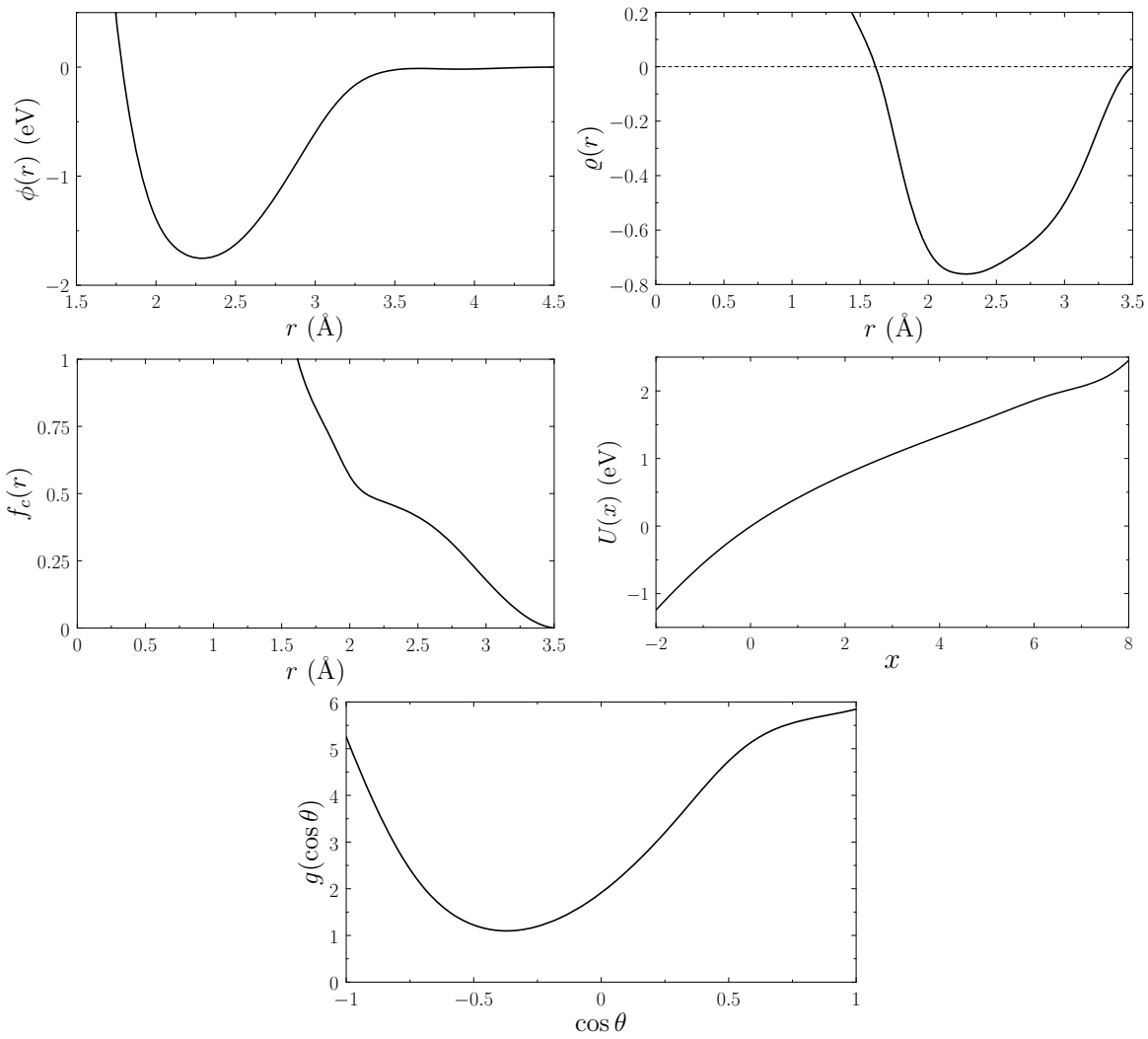


Figure C.1: Spline functions defining the LSA potential. Recall from the discussion at the end of Section 5.2.1 that these forms do not all have physical meaning.

C.4 Stillinger–Weber (SW)

Literary reference: [43, 261]
OpenKIM references: [262, 263]
Functional form:

$$\mathcal{V} = \sum_i \sum_{i<j} V_2(r_{ij}) + \sum_i \sum_{i<j} \sum_{j<k} V_3(\mathbf{r}_{ij}, \mathbf{r}_{ik}, \mathbf{r}_{jk})$$

$$V_2(r_{ij}) = \varepsilon A \left[B \left(\frac{\sigma}{r}\right)^p - \left(\frac{\sigma}{r}\right)^q \right] f_{c,2}\left(r_{ij}\right)$$

$$f_{c,2}(r) = \begin{cases} \exp\left(\dfrac{\sigma}{r-a} + \dfrac{\sigma}{a}\right) & r < a \\ 0 & r \geq a \end{cases}$$

$$V_3(\mathbf{r}_{ij}, \mathbf{r}_{ik}, \mathbf{r}_{jk}) = \varepsilon C \lambda \left[h(r_{ij}, r_{ik}, \theta_{jik}) + h(r_{ij}, r_{jk}, \theta_{ijk}) + h(r_{ik}, r_{jk}, \theta_{ikj}) \right]$$

$$\theta_{jik} = \arccos\left(\frac{\mathbf{r}_{ij} \cdot \mathbf{r}_{ik}}{r_{ij} r_{ik}}\right), \quad \theta_{ijk} = \arccos\left(-\frac{\mathbf{r}_{ij} \cdot \mathbf{r}_{jk}}{r_{ij} r_{jk}}\right), \quad \theta_{ikj} = \arccos\left(\frac{\mathbf{r}_{ik} \cdot \mathbf{r}_{jk}}{r_{ik} r_{jk}}\right)$$

$$h(r_{ij}, r_{ik}, \theta_{jik}) = f_{c,3}(r_{ij}) f_{c,3}(r_{ik}) g(\theta_{jik})$$

$$g(\theta) = (\cos\theta - \cos\theta_0)^2$$

$$f_{c,3}(r) = \begin{cases} \exp\left(\dfrac{\gamma}{r-a} + \dfrac{\gamma}{a}\right) & r < a \\ 0 & r \geq a \end{cases}$$

Parameters: First, note that we do not employ the original SW parameters because they predict the cohesive energy of diamond to be approximatedly -4.3364 eV, in contrast to the experimental value of -4.63 eV. Rather, we use the parameters put forth by Balamane in [261], which differ only from the original set by scaling the original parameter ε so that $\varepsilon = 1.07\varepsilon_{\text{orig}}$. Further, notice from the functional form above that we have eliminated the nondimensionalization employed in the original publication, where the inputs to two-body and three-body energies are divided by σ. Finally, as done in the EDIP potential, in order to identify two-body and three-body cutoff functions $f_{c,2}(r)$ and $f_{c,3}(r)$ which are equal to unity at $r = 0$, we have incorporated the appropriate normalization factors into the parameter $A = \exp(-\sigma/a)A_{\text{orig}}$ and an additional parameter $C = \exp(-2\gamma/a)$.

$A = 4.04470702861$	$B = 0.6022245584$	$C = 0.263597138116$
$p = 4$	$q = 0$	$a = 3.7712\,\text{Å}$
$\cos\theta_0 = -1/3$	$\lambda = 21.0$	$\sigma = 2.0951\,\text{Å}$
$\varepsilon = 2.315\,\text{eV}$	$\gamma = 2.5141\,\text{Å}$	

C.5 Stillinger–Weber for silicene (SWS1)

Literary reference: [44]
OpenKIM references: [145, 263]
Functional form: This EP utilizes a functional form identical to that of the SW potential given above.
Parameters: Similar to the SW potential, we have defined parameters $A = \exp(-\sigma/a)A_{\text{orig}}$ and $C = \exp(-2\gamma/a)$.

$A = 3.41542104922$	$B = 0.618328$	$C = 0.263987654672$
$p = 4$	$q = 0$	$a = 3.55546917008\,\text{Å}$
$\cos\theta_0 = -0.44561015$	$\lambda = 15.662962$	$\sigma = 2.00336\,\text{Å}$
$\varepsilon = 2.1683\,\text{eV}$	$\gamma = 2.3676810328\,\text{Å}$	

C.6 Tersoff (T2)

Literary reference: [124]
OpenKIM references: [257, 264]
Functional form: While the original form given by Tersoff for the T2 and T3 potentials was given in Section 5.1, we adopt here the notation used in the publication of TMOD [266] in order to make the relation between the three transparent. Also, note that $b_{ij} \neq b_{ji}$, in general, which prevents the use of a half summation.

$$\mathcal{V} = \frac{1}{2}\sum_i \sum_{j\neq i} V_{ij}$$

$$V_{ij} = f_c(r_{ij})[f_R(r_{ij}) - b_{ij} f_A(r_{ij})]$$

$$f_R(r) = A\exp(-\lambda_1 r), \quad f_A(r) = B\exp(-\lambda_2 r)$$

$$f_c(r) = \begin{cases} 1 & r \leq R_1 \\ \frac{1}{2}\left[1 + \cos\left(\pi\frac{r-R_1}{R_2-R_1}\right)\right] & R_1 < r < R_2 \\ 0 & r \geq R_2 \end{cases}$$

$$b_{ij} = (1 + \zeta_{ij}^{\eta})^{-\delta}$$

$$\zeta_{ij} = \sum_{k\neq i,j} f_c(r_{ik}) g(\theta_{ijk}) \exp\left[m(r_{ij} - r_{ik})^n\right]$$

$$
\begin{aligned}
g(\theta) &= c_1 + g_o(\theta) g_a(\theta) \\
g_o(\theta) &= \frac{c_2 (h - \cos\theta)^2}{c_3 + (h - \cos\theta)^2} \\
g_a(\theta) &= 1 + c_4 \exp[-c_5 (h - \cos\theta)^2]
\end{aligned}
$$

Parameters:

$$
\begin{aligned}
A &= 3264.7\,\text{eV} & B &= 95.373\,\text{eV} \\
\lambda_1 &= 3.2394\,\text{Å}^{-1} & \lambda_2 &= 1.3258\,\text{Å}^{-1} \\
\eta &= 22.956 & \eta \times \delta &= 0.5 \\
m &= 2.3304191695120005 & n &= 3 \\
c_1 &= 0.33675 & c_2 &= 1.890921187895526 \\
c_3 &= 4.16853889 & c_4 &= 0 \\
c_5 &= 0 & h &= 0 \\
R_1 &= 2.8\,\text{Å} & R_2 &= 3.2\,\text{Å}
\end{aligned}
$$

C.7 Tersoff (T3)

Literary reference: [144]

OpenKIM references: [257, 265]

Functional form: The functional form of this EP is identical to that of the T2 potential given above.

Parameters:

$$
\begin{aligned}
A &= 1830.8\,\text{eV} & B &= 471.18\,\text{eV} \\
\lambda_1 &= 2.4799\,\text{Å}^{-1} & \lambda_2 &= 1.7322\,\text{Å}^{-1} \\
\eta &= 0.78734 & \eta \times \delta &= 0.5 \\
m &= 5.197495270248 & n &= 3 \\
c_1 &= 1.0999 \cdot 10^{-6} & c_2 &= 42.14436536402729 \\
c_3 &= 263.023524 & c_4 &= 0 \\
c_5 &= 0 & h &= -0.59826 \\
R_1 &= 2.7\,\text{Å} & R_2 &= 3.0\,\text{Å}
\end{aligned}
$$

C.8 Modified Tersoff (TMOD)

Literary reference: [266]
OpenKIM references: None
Functional form:
The functional form of this EP is identical to that of the T2 potential given above, except that the cutoff function $f_c(r)$ is modified to be

$$f_c(r) = \begin{cases} 1 & r \le R_1 \\ \frac{1}{2} + \frac{9}{16}\cos\left(\pi\frac{r-R_1}{R_2-R_1}\right) - \frac{1}{16}\cos\left(3\pi\frac{r-R_1}{R_2-R_1}\right) & R_1 < r < R_2 \\ 0 & r \ge R_2 \end{cases}$$

Parameters:

$$A = 3281.5905\,\text{eV} \qquad B = 121.00047\,\text{eV}$$
$$\lambda_1 = 3.2300135\,\text{Å}^{-1} \qquad \lambda_2 = 1.3457970\,\text{Å}^{-1}$$
$$\eta = 1.0000000 \qquad \eta \times \delta = 0.53298909$$
$$m = 2.3890327 \qquad n = 1$$
$$c_1 = 0.20173476 \qquad c_2 = 730418.72$$
$$c_3 = 1000000.0 \qquad c_4 = 1.0000000$$
$$c_5 = 26.000000 \qquad h = -0.36500000$$
$$R_1 = 2.7\,\text{Å} \qquad R_2 = 3.3\,\text{Å}$$

C.9 Analytical definition of atomic energies

In Section 2.2.5, it was remarked that many EPs are cast in terms of individual bond energies rather than atomic energies, and that this is manifested in the fact that the summations carried out by these EPs to arrive at the total energy of an atomic configuration are reduced based on symmetry considerations. Thus, the summation which features the two-body contribution of many EPs is written as a half summation $\sum_{\alpha<\beta}$ rather than a full summation $\sum_{\alpha\neq\beta}$, so that each individual bond between a pair of atoms is considered only once. Similarly, the sums over triplets of atoms are performed under the constraint $\sum_{\alpha<\beta<\gamma}$ rather than the full summation $\sum_{\alpha\neq\beta\neq\gamma}$, so that each unique triplet is considered only once in the summation. As a result, EPs which leverage this methodology do not define atomic energies which are invariant to permutations of the indices used to refer to each atom. However, while such EPs do not natively define atomic energies, it is possible in most cases to derive an alternative form of them which eases the above summation constraints, and thus permits the definition of the latter. In this section, we perform a specific reformulation of this type for each of the EPs thus far presented, although we stress that the particular decompositions carried out here are not the only possible course of definition.

C.9.1 Tersoff-type EPs (EA, T2, T3, TMOD)

Following Balamane [261], we choose to define the two-body energy V_2 and three-body energy V_3 of the Tersoff-type potentials by rewriting their total energy as

$$\begin{aligned}
\mathcal{V} &= \frac{1}{2}\sum_{\alpha}\sum_{\beta\neq\alpha} f_c(r_{\alpha\beta})\left[f_R(r_{\alpha\beta}) - b_{\alpha\beta}f_A(r_{\alpha\beta})\right] \\
&= \frac{1}{2}\sum_{\alpha}\sum_{\beta\neq\alpha} f_c(r_{\alpha\beta})\left[f_R(r_{\alpha\beta}) - f_A(r_{\alpha\beta}) + f_A(r_{\alpha\beta}) - b_{\alpha\beta}f_A(r_{\alpha\beta})\right] \\
&= \frac{1}{2}\sum_{\alpha}\sum_{\beta\neq\alpha} f_c(r_{\alpha\beta})\left[f_R(r_{\alpha\beta}) - f_A(r_{\alpha\beta})\right] + \frac{1}{2}\sum_{\alpha}\sum_{\beta\neq\alpha} f_c(r_{\alpha\beta})f_A(r_{\alpha\beta})\left[1 - b_{\alpha\beta}\right],
\end{aligned}$$

whence we choose to define

$$V_2(r_{\alpha\beta}) \triangleq \frac{1}{2}f_c(r_{\alpha\beta})\left[f_R(r_{\alpha\beta}) - f_A(r_{\alpha\beta})\right], \tag{C.1}$$

$$V_3(r_{\alpha\beta}, b_{\alpha\beta}) \triangleq \frac{1}{2}f_c(r_{\alpha\beta})f_A(r_{\alpha\beta})\left[1 - b_{\alpha\beta}\right], \tag{C.2}$$

so that

$$\mathcal{V} = \sum_{\alpha}\sum_{\beta\neq\alpha} V_2(r_{\alpha\beta}) + \sum_{\alpha}\sum_{\beta\neq\alpha} V_3(r_{\alpha\beta}, b_{\alpha\beta}).$$

The motivation for the above partitioning of the energy is that the two-body energy $V_2(r_{\alpha\beta})$ is independent of the bond order $b_{\alpha\beta}$, which incorporates three-body contributions. The two-body energy associated with atom α is then given by

$$\mathcal{V}_{2,\alpha}^{\mathrm{EP}} = \sum_{\beta \neq \alpha} V_2(r_{\alpha\beta}), \tag{C.3}$$

while the three-body energy associated with atom α is given by

$$\mathcal{V}_{3,\alpha}^{\mathrm{EP}} = \sum_{\beta \neq \alpha} V_3(r_{\alpha\beta}, b_{\alpha\beta}). \tag{C.4}$$

C.9.2 EDIP

The two-body summations in EDIP, including those carried out for the coordination Z_α of each atom, are already taken over all indices $\beta \neq \alpha$, and thus the two-body energy of atom α can simply be defined as

$$\mathcal{V}_{2,\alpha}^{\mathrm{EP}} \triangleq \sum_{\beta \neq \alpha} V_2(r_{\alpha\beta}, Z_\alpha). \tag{C.5}$$

The three-body energy of EDIP is written as a nested summation of the form $\sum_{\beta\neq\alpha}\sum_{\gamma\neq\alpha,\gamma>\beta}$. The consequence of this form is that each atom which belongs to an arbitrary triplet possesses its own three-body contribution to its energy (this can be verified by writing out the associated valid index permutations for a cluster of three atoms). Thus, the three-body energy of atom α can also be defined directly as

$$\mathcal{V}_{3,\alpha}^{\mathrm{EP}} \triangleq \sum_{\beta \neq \alpha} \sum_{\substack{\gamma \neq \alpha \\ \gamma > \beta}} V_3(\mathbf{r}_{\alpha\beta}, \mathbf{r}_{\alpha\gamma}, Z_\alpha). \tag{C.6}$$

C.9.3 LSA

The three-body interactions of LSA are present only in the embedding density ϱ^{CF}, i.e. the argument of the functional U. The two-body energy of atom α is

$$\mathcal{V}_{2,\alpha}^{\mathrm{EP}} \triangleq \sum_{\beta \neq \alpha} \frac{1}{2}\phi(r_{\alpha\beta}), \tag{C.7}$$

while the embedding energy associated with atom α is

$$\mathcal{V}_{U,\alpha}^{\mathrm{EP}} \triangleq U\left[\sum_{\beta\neq\alpha}\varrho(r_{\alpha\beta}) + \sum_{\substack{\beta\neq\alpha\\ \gamma>j}} f_c(r_{\alpha\beta})f_c(r_{\alpha\gamma})g(\cos\theta_{\beta\gamma})\right] - U[0], \tag{C.8}$$

so that $\mathcal{V} = \sum_\alpha \mathcal{V}_{2,\alpha}^{\mathrm{EP}} + \sum_\alpha \mathcal{V}_{U,\alpha}^{\mathrm{EP}}$.

C.9.4 Stillinger–Weber-type EPs (SW, SWS1)

Rewriting the two-body energy of SW is trivially accomplished by writing

$$\sum_\alpha \sum_{\alpha<\beta} V_2(r_{\alpha\beta}) = \sum_\alpha \sum_{\beta\neq\alpha} \frac{1}{2} V_2(r_{\alpha\beta}),$$

so that the two-body energy of atom α can be defined as

$$\mathcal{V}_{2,\alpha}^{\mathrm{EP}} \triangleq \sum_{\beta\neq\alpha} \frac{1}{2} V_2(r_{\alpha\beta}). \tag{C.9}$$

Next, one may observe that, although the summation

$$\sum_\alpha \sum_{\alpha<\beta} \sum_{\beta<\gamma} V_3(\mathbf{r}_{\alpha\beta}, \mathbf{r}_{\alpha\gamma}, \mathbf{r}_{\beta\gamma})$$

runs over each unique triplet of atoms in a configuration only once, three terms are considered in each such instance. One may thus rewrite the reduced three-body summation in the form

$$\sum_\alpha \sum_{\alpha<\beta} \sum_{\beta<\gamma} V_3(\mathbf{r}_{\alpha\beta}, \mathbf{r}_{\alpha\gamma}, \mathbf{r}_{\beta\gamma}) = \sum_\alpha \sum_{\beta\neq\alpha} \sum_{\substack{\gamma\neq\alpha\\ \gamma>\beta}} \tilde{V}_3(\mathbf{r}_{\alpha\beta}, \mathbf{r}_{\alpha\gamma}, \mathbf{r}_{\beta\gamma}),$$

where the newly defined function $\tilde{V}_3(\mathbf{r}_{\alpha\beta}, \mathbf{r}_{\alpha\gamma}, \mathbf{r}_{\beta\gamma})$ includes only one of the three terms which comprise $V_3(\mathbf{r}_{\alpha\beta}, \mathbf{r}_{\alpha\gamma}, \mathbf{r}_{\beta\gamma})$:

$$\tilde{V}_3(\mathbf{r}_{\alpha\beta}, \mathbf{r}_{\alpha\gamma}, \mathbf{r}_{\beta\gamma}) \triangleq \varepsilon C\lambda h(r_{\alpha\beta}, r_{\alpha\gamma}, \theta_{j\alpha\gamma}). \tag{C.10}$$

The three-body energy of atom α is then

$$\mathcal{V}_{3,\alpha}^{\mathrm{EP}} = \sum_{\beta\neq\alpha} \sum_{\substack{\gamma\neq\alpha\\ \gamma>\beta}} \tilde{V}_3(\mathbf{r}_{\alpha\beta}, \mathbf{r}_{\alpha\gamma}, \mathbf{r}_{\beta\gamma}). \tag{C.11}$$

	Ref.[a]	EA[b]	EDIP[c]	LSA[b]	SW[c]	T2[c]	T3[c]	TMOD[a]
E_{coh} (eV)	-4.63	-4.63	-4.65	-4.61	-4.63	-4.63	-4.63	-4.63
a_0 (Å)	5.429	5.429	5.430	5.430	5.431	5.431	5.432	5.429
B	99	99	99	110	108	98	98	99
C_{11}	167	167	175	165	161	127	143	166.4
C_{12}	65	65	62	82	82	86	75	65.3
C_{44}	81	72	71.7	72	60	10	69	77.1
C_{44}^0	106	111	112	-	117	92	119	120.9
C_{11}-C_{12}	102	102	113	83.1	79	41	68	101.1
C_{12}-C_{44}	-16	-7	-9	10	22	76	6	-11.8
ζ	0.533	0.52	0.497[e]	-	0.63[f]	0.83[f]	0.6746[a]	0.5526
T_m (K)	1687	2125-2175	1350-1390	1200-1300[d]	1776-1867[f]	-	2391-2569[a,b]	1681

Table C.1: Properties of diamond silicon according to experiment, first-principles calculations, and as predicted by the EPs presented in this section. The bulk modulus B and all elastic constants are given in units of GPa. The reference value for B was computed from the reference values of C_{11} and C_{12} using the relation $B = (\mathrm{C}_{11} + 2\,\mathrm{C}_{12})/3$. The Kleinman parameter is denoted ζ, while the quantity C_{44}^0 denotes the elastic constant C_{44} calculated under the constraint $\zeta = 0$. Values for the SWS1 potential, as well as entries for which a – is listed could not be found in the literature.

[a] [266].
[b] [147].
[c] [258].
[d] [179].
[e] [289].
[f] [261].

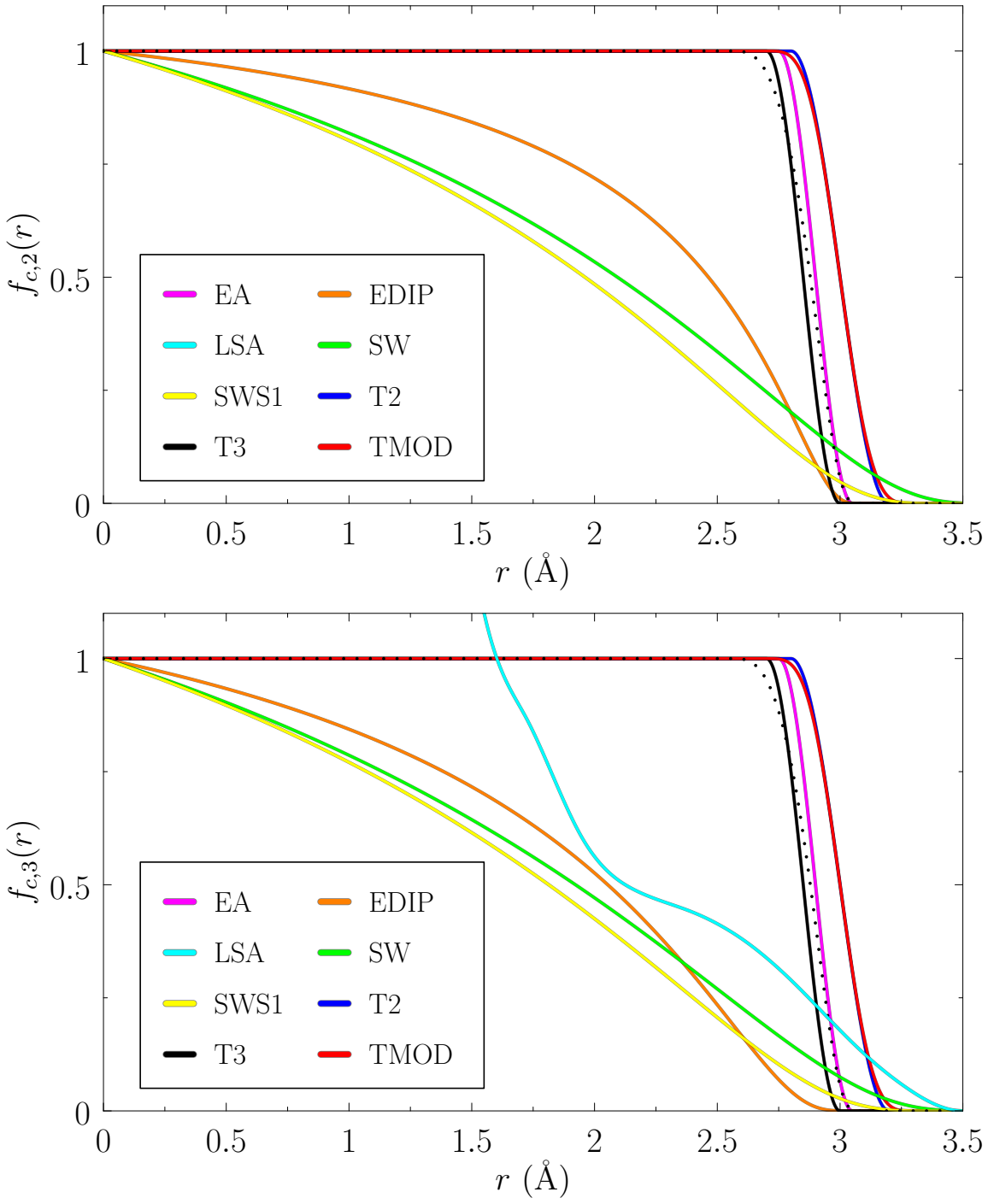


Figure C.2: (Top) Radial cutoff functions used to control the two-body interactions of each potential other than LSA, which has no such explicit cutoff function. (Bottom) Radial cutoff functions used to control the three-body interactions of each potential. The black dotted line shown in both figures indicates the cutoff function $f_{c,Z}$ used in the EDIP potential to calculate the coordination of a given atom.

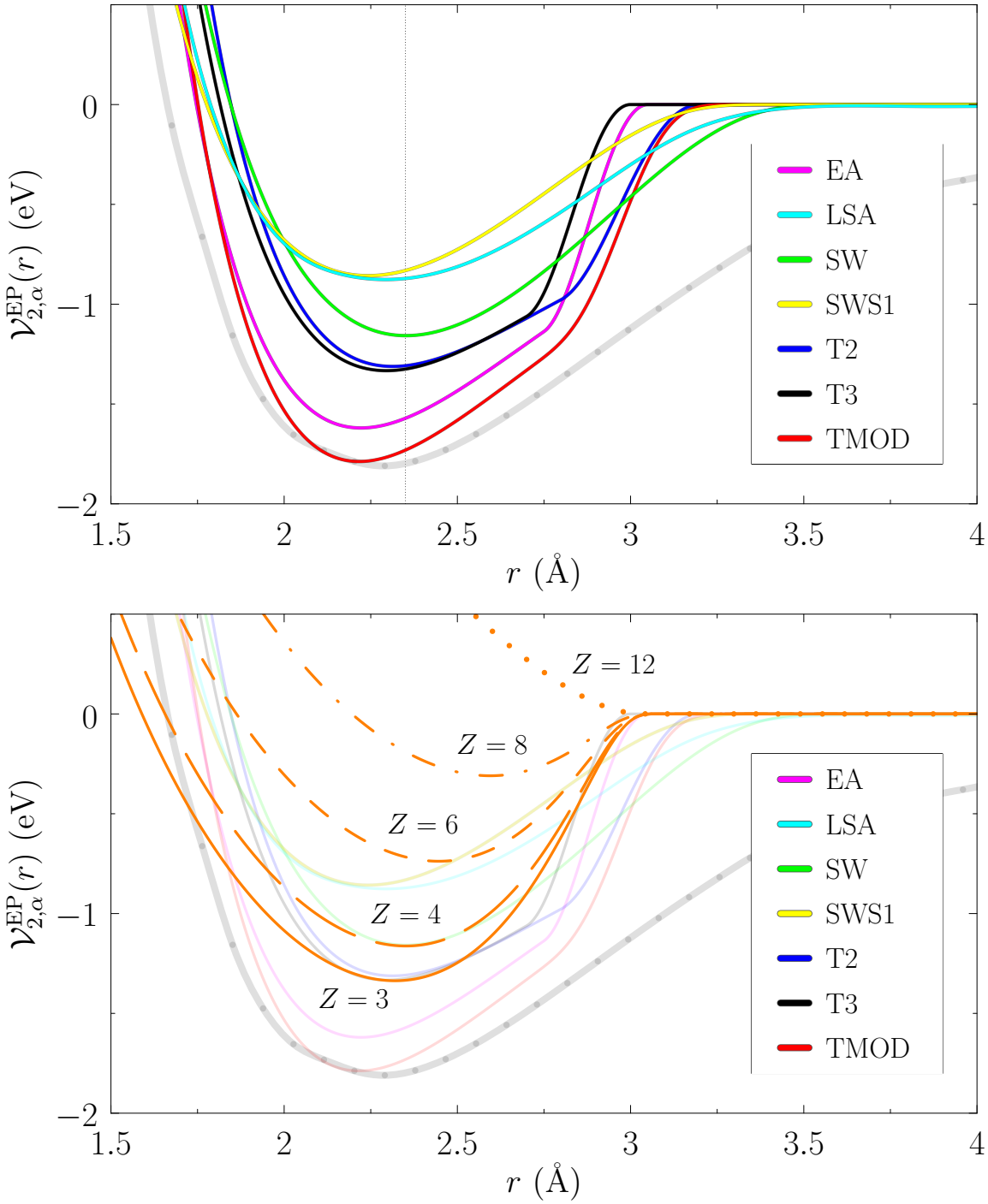


Figure C.3: (Top) Two-body energy $\mathcal{V}_{2,\alpha}^{\mathrm{EP}}$ of the EPs other than EDIP for one atom of a dimer with separation distance r. The black dotted line sits at 2.35 Å, the first nearest-neighbor distance in diamond. (Bottom) Two-body energy $\mathcal{V}_{2,\alpha}^{\mathrm{EP}}$ of EDIP for coordinations Z =3, 4, 6, 8, and 12. The two-body energies of the remaining potentials are transparently superposed. In both figures, the light grey line represents a cubic spline interpolation of one-half of the total DFT energy of dimers with bond lengths indicated by grey dots.

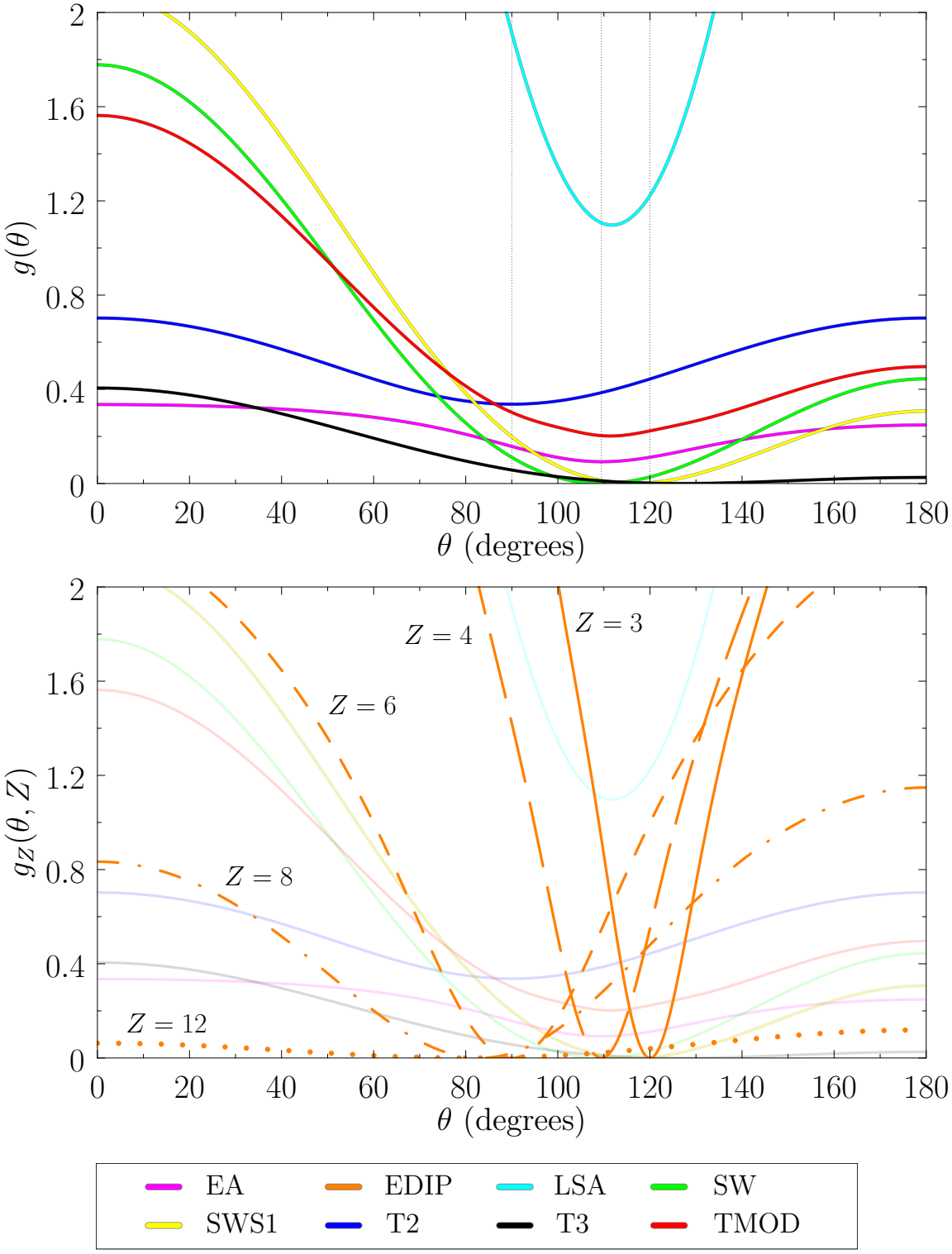


Figure C.4: (Top) Angular functions $g(\theta)$ used in the EPs other than EDIP, none of which has any coordination dependence. Black dotted lines are shown at 90°, 109.47°, and 120° for reference. (Bottom) Angular function $g_Z(\theta, Z)$ of EDIP for coordinations Z=3, 4, 6, 8, and 12. The angular functions of the remaining potentials are transparently superposed.

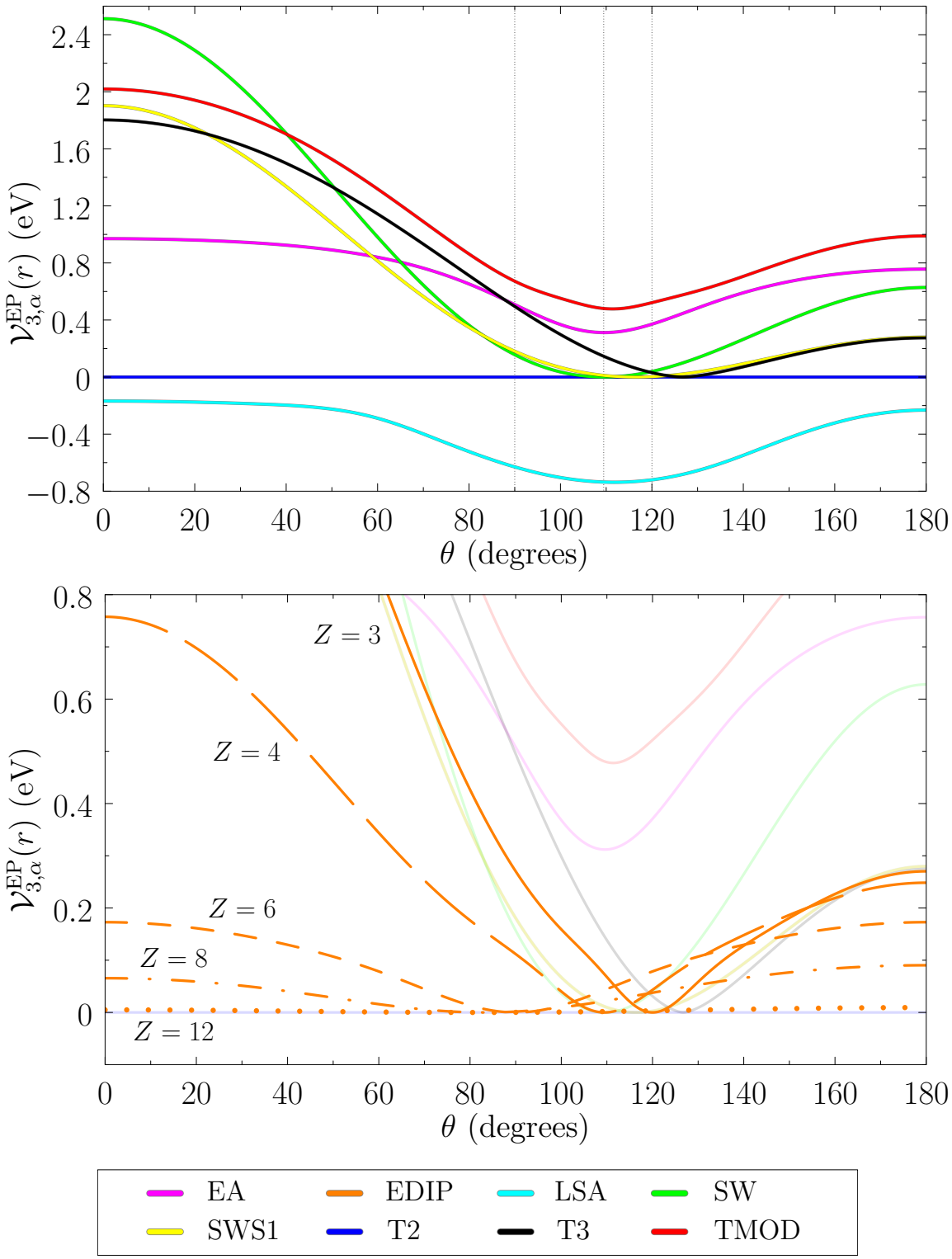


Figure C.5: (Top) Three-body energy $\mathcal{V}^{\rm EP}_{3,\alpha}$ of the apex atom of a trimer containing two bonds of length 2.35 Å separated by an angle θ for the EPs other than EDIP. The curve shown for the LSA potential is the embedding energy $\mathcal{V}^{\rm EP}_{U,\alpha}$ of the apex atom. See Section C.9 for details. (Bottom) The three-body energy $\mathcal{V}^{\rm EP}_{3,\alpha}$ of EDIP for the apex atom of the same trimer, but where the coordination has been set equal to Z=3, 4, 6, 8 and 12.

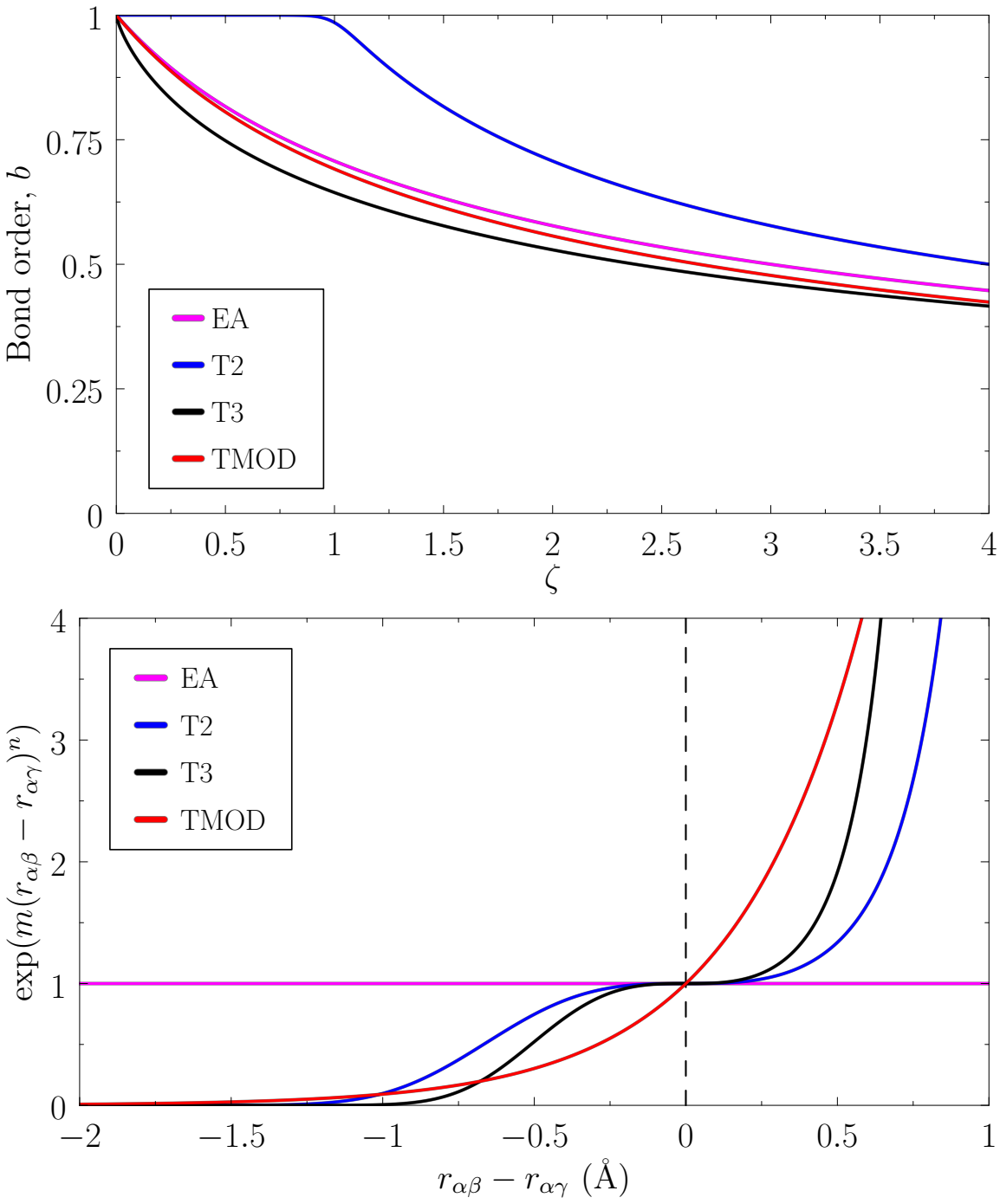


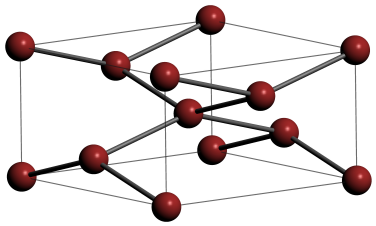
Figure C.6: (Top) Bond order b as a function of effective coordination ζ for the Tersoff-type potentials: EA, T2, T3, and TMOD. (Bottom) The term $\exp(m(r_{\alpha\beta} - r_{\alpha\gamma})^n)$ which modulates the three-body interactions in the Tersoff-type potentials.

Appendix D

Bulk structures

Below, we give the primitive unit cells of the ten bulk crystal structures considered in Chapter 6, as well as accompanying illustrations which, in most cases, correspond to the conventional unit cell of each. The "ideal" lattice parameters are taken from [261].

D.1 β-Sn



Lattice vectors:

$$\mathbf{A}_1 = (a, 0, 0) \qquad \mathbf{A}_2 = (0, a, 0) \qquad \mathbf{A}_1 = \frac{1}{2}(a, a, c)$$

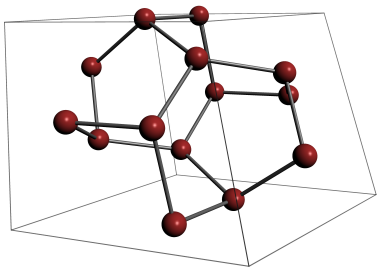
Basis atom coordinates:

$$\mathbf{b}_1 = \frac{-1}{8}\mathbf{A}_1 - \frac{3}{8}\mathbf{A}_2 + \frac{1}{4}\mathbf{A}_3 \qquad \mathbf{b}_2 = \frac{1}{8}\mathbf{A}_1 + \frac{3}{8}\mathbf{A}_2 - \frac{1}{4}\mathbf{A}_3$$

Ideal lattice parameters in this work:

$$a = 4.822\text{Å} \qquad c = 0.552a$$

D.2 bc8



Lattice vectors:

$$\mathbf{A}_1 = \frac{1}{2}(-a, a, a) \qquad \mathbf{A}_2 = \frac{1}{2}(a, -a, a) \qquad \mathbf{A}_3 = \frac{1}{2}(a, a, -a)$$

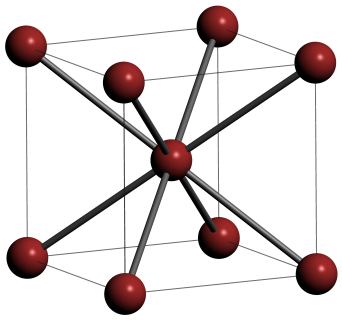
Basis atom coordinates:

$$\begin{aligned}
\mathbf{b}_1 &= 2x\mathbf{A}_1 + 2x\mathbf{A}_2 + 2x\mathbf{A}_3 \\
\mathbf{b}_2 &= -2x\mathbf{A}_1 - 2x\mathbf{A}_2 - 2x\mathbf{A}_3 \\
\mathbf{b}_3 &= \left(\frac{1}{2} - 2x\right)\mathbf{A}_1 + \frac{1}{2}\mathbf{A}_2 \\
\mathbf{b}_4 &= -\left(\frac{1}{2} - 2x\right)\mathbf{A}_1 - \frac{1}{2}\mathbf{A}_2 \\
\mathbf{b}_5 &= \left(\frac{1}{2} - 2x\right)\mathbf{A}_2 + \frac{1}{2}\mathbf{A}_3 \\
\mathbf{b}_6 &= -\left(\frac{1}{2} - 2x\right)\mathbf{A}_2 - \frac{1}{2}\mathbf{A}_3 \\
\mathbf{b}_7 &= \frac{1}{2}\mathbf{A}_1 + \left(\frac{1}{2} - 2x\right)\mathbf{A}_3 \\
\mathbf{b}_8 &= -\frac{1}{2}\mathbf{A}_1 - \left(\frac{1}{2} - 2x\right)\mathbf{A}_3
\end{aligned}$$

Ideal lattice parameters in this work:

$$a = 6.67\text{Å} \qquad x = 0.1003 \text{ (unitless)}$$

D.3 bcc



Lattice vectors:

$$\mathbf{A}_1 = \frac{1}{2}(-a, a, a) \qquad \mathbf{A}_2 = \frac{1}{2}(a, -a, a) \qquad \mathbf{A}_3 = \frac{1}{2}(a, a, -a)$$

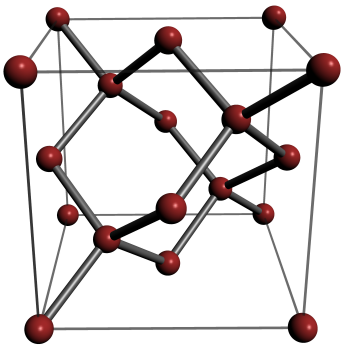
Basis atom coordinates:

$$\mathbf{b}_1 = \mathbf{0}$$

Ideal lattice parameters in this work:

$$a = 3.088\text{Å}$$

D.4 diamond



Lattice vectors:

$$\mathbf{A}_1 = \frac{1}{2}(0, a, a) \qquad \mathbf{A}_2 = \frac{1}{2}(a, 0, a) \qquad \mathbf{A}_3 = \frac{1}{2}(a, a, 0)$$

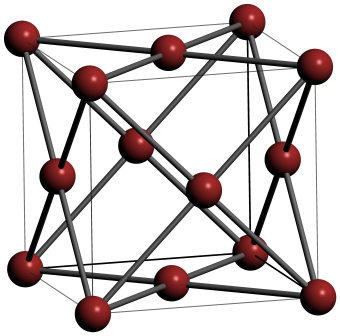
Basis atom coordinates:

$$\mathbf{b}_1 = \frac{1}{8}\mathbf{A}_1 + \frac{1}{8}\mathbf{A}_2 + \frac{1}{8}\mathbf{A}_3 \qquad \mathbf{b}_2 = -\frac{1}{8}\mathbf{A}_1 - \frac{1}{8}\mathbf{A}_2 - \frac{1}{8}\mathbf{A}_3$$

Ideal lattice parameters in this work:

$$a = 5.429\text{Å}$$

D.5 fcc



Lattice vectors:

$$\mathbf{A}_1 = \frac{1}{2}(0, a, a) \qquad \mathbf{A}_2 = \frac{1}{2}(a, 0, a) \qquad \mathbf{A}_3 = \frac{1}{2}(a, a, 0)$$

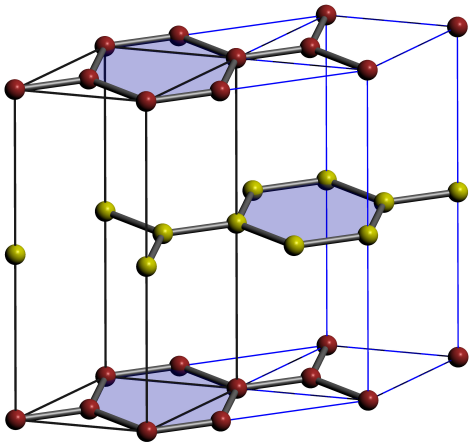
Basis atom coordinates:

$$\mathbf{b}_1 = \mathbf{0}$$

Ideal lattice parameters in this work:

$$a = 3.885\text{Å}$$

D.6 graphite



Lattice vectors:

$$\mathbf{A}_1 = \frac{1}{2}(\sqrt{3}\,a, -a, 0) \qquad \mathbf{A}_2 = \frac{1}{2}(\sqrt{3}\,a, a, 0) \qquad \mathbf{A}_3 = (0, 0, c)$$

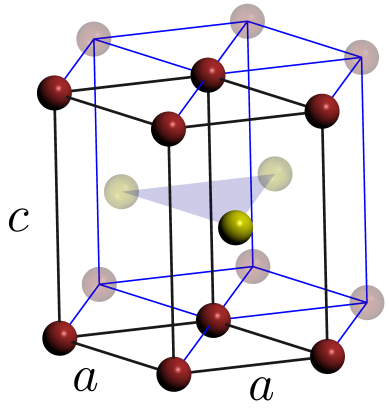
Basis atom coordinates:

$$\mathbf{b}_1 = \mathbf{0} \qquad \mathbf{b}_3 = \frac{1}{2}\mathbf{A}_3$$
$$\mathbf{b}_2 = -\frac{1}{3}\mathbf{A}_1 + \frac{2}{3}\mathbf{A}_2 \qquad \mathbf{b}_4 = \frac{1}{3}\mathbf{A}_1 - \frac{2}{3}\mathbf{A}_2 + \frac{1}{2}\mathbf{A}_3$$

Ideal lattice parameters in this work:

$$a = 3.895\text{Å} \qquad c = 2.726a$$

D.7 hcp



Lattice vectors:

$$\mathbf{A}_1 = (a, 0, 0) \qquad \mathbf{A}_2 = \frac{1}{2}(-a, \sqrt{3}\,a, 0) \qquad \mathbf{A}_3 = (0, 0, c)$$

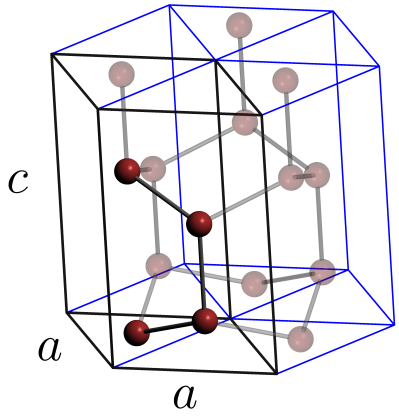
Basis atom coordinates:

$$\mathbf{b}_1 = \mathbf{0} \qquad \mathbf{b}_2 = \frac{2}{3}\mathbf{A}_1 + \frac{1}{3}\mathbf{A}_2 + \frac{1}{2}\mathbf{A}_3$$

Ideal lattice parameters in this work:

$$a = 2.735\text{Å} \qquad c = 1.633a$$

D.8 hexagonal diamond (hd)



Lattice vectors:

$$\mathbf{A}_1 = \frac{1}{2}(a, -\sqrt{3}\,a, 0) \qquad \mathbf{A}_2 = \frac{1}{2}(a, \sqrt{3}\,a, 0) \qquad \mathbf{A}_3 = (0, 0, c)$$

Basis atom coordinates:

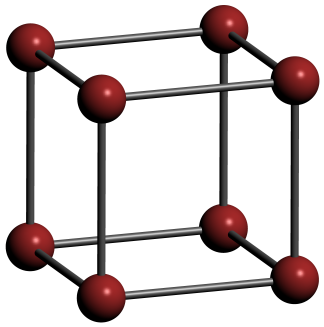
$$\mathbf{b}_1 = \frac{1}{3}\mathbf{A}_1 + \frac{2}{3}\mathbf{A}_2 + u\mathbf{A}_3 \qquad \mathbf{b}_3 = \frac{1}{3}\mathbf{A}_1 + \frac{2}{3}\mathbf{A}_2 + \left(\frac{1}{2} - u\right)\mathbf{A}_3$$

$$\mathbf{b}_2 = \frac{2}{3}\mathbf{A}_1 + \frac{1}{3}\mathbf{A}_2 + \left(\frac{1}{2} + u\right)\mathbf{A}_3 \qquad \mathbf{b}_4 = \frac{2}{3}\mathbf{A}_1 + \frac{1}{3}\mathbf{A}_2 - u\mathbf{A}_3$$

Ideal lattice parameters in this work:

$$a = 3.858\text{Å} \qquad c = 1.633a \qquad u = \frac{1}{16}$$

D.9 sc



Lattice vectors:

$$\mathbf{A}_1 = (a, 0, 0) \qquad \mathbf{A}_2 = (0, a, 0) \qquad \mathbf{A}_3 = (0, 0, a)$$

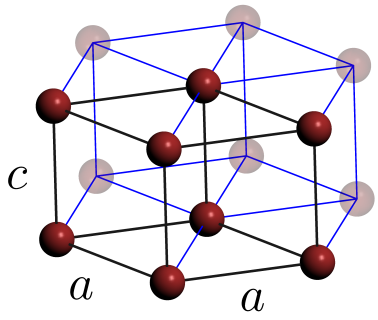
Basis atom coordinates:

$$\mathbf{b}_1 = \mathbf{0}$$

Ideal lattice parameters in this work:

$$a = 2.528\text{Å}$$

D.10 sh



Lattice vectors:

$$\mathbf{A}_1 = \frac{1}{2}(a, -\sqrt{3}\ a, 0) \qquad \mathbf{A}_2 = \frac{1}{2}(a, \sqrt{3}\ a, 0) \qquad \mathbf{A}_3 = (0, 0, c)$$

Basis atom coordinates:

$$\mathbf{b}_1 = \mathbf{0}$$

Ideal lattice parameters in this work:

$$a = 2.639\text{Å} \qquad c = 0.94a$$

Appendix E

Miscellany

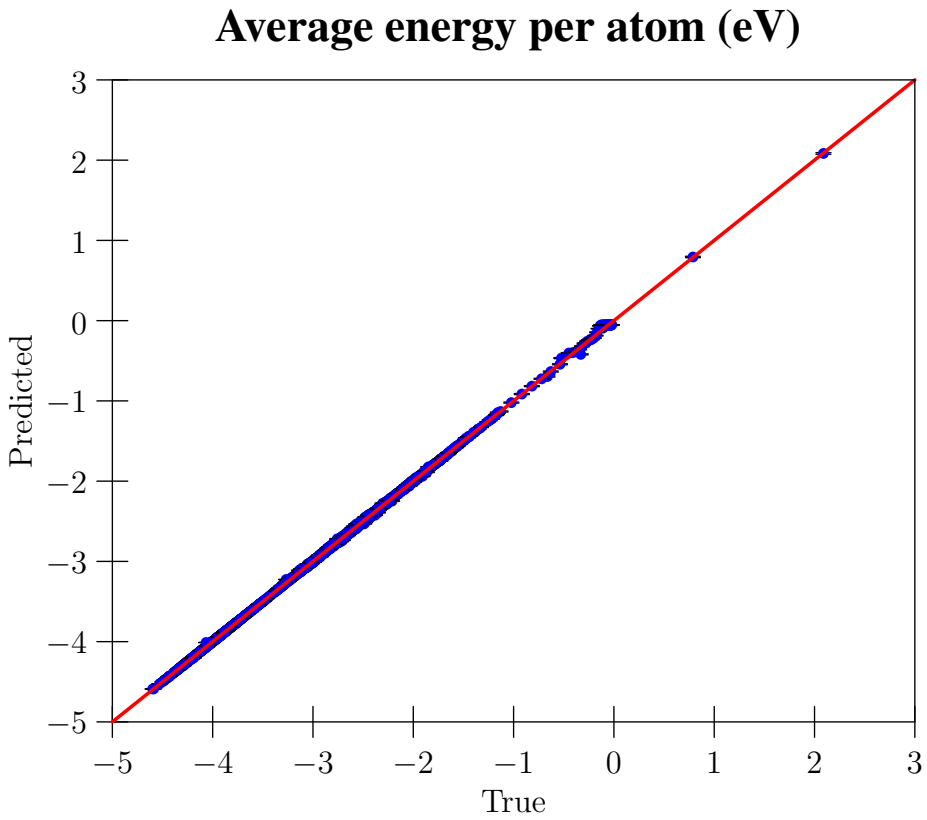


Figure E.1: Comparison of the true first-principles total energies computed for each the training set configurations with those obtained by summing the corresponding atomic energies of each configuration learned by the regression using the SOAP parameters in Table 6.2 and a noise parameter $\sigma_n = 0.01$.

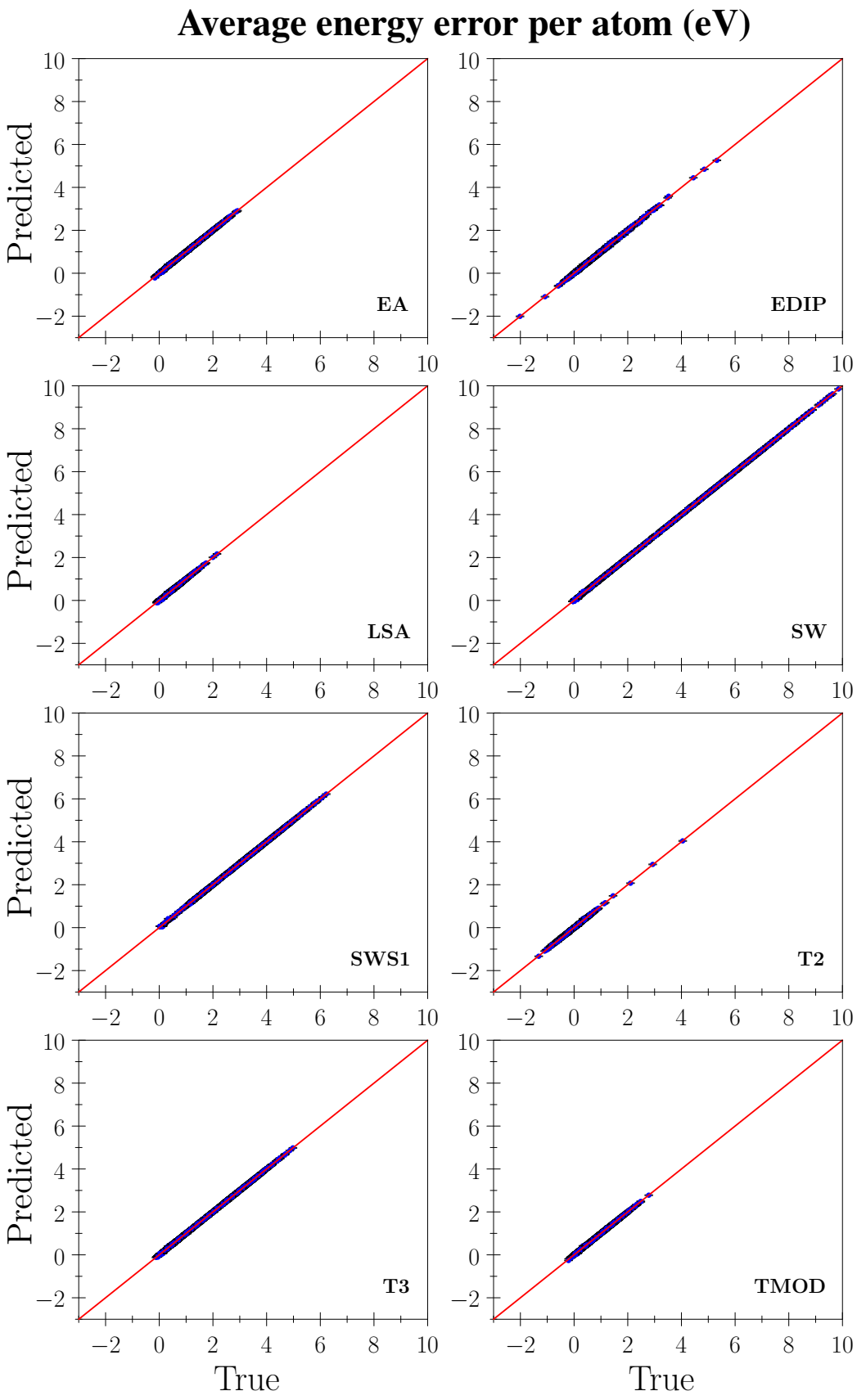


Figure E.2: Comparison of the true atomic energy errors for each EP with those obtained by summing the atomic energy errors of each EP learned by the regression.

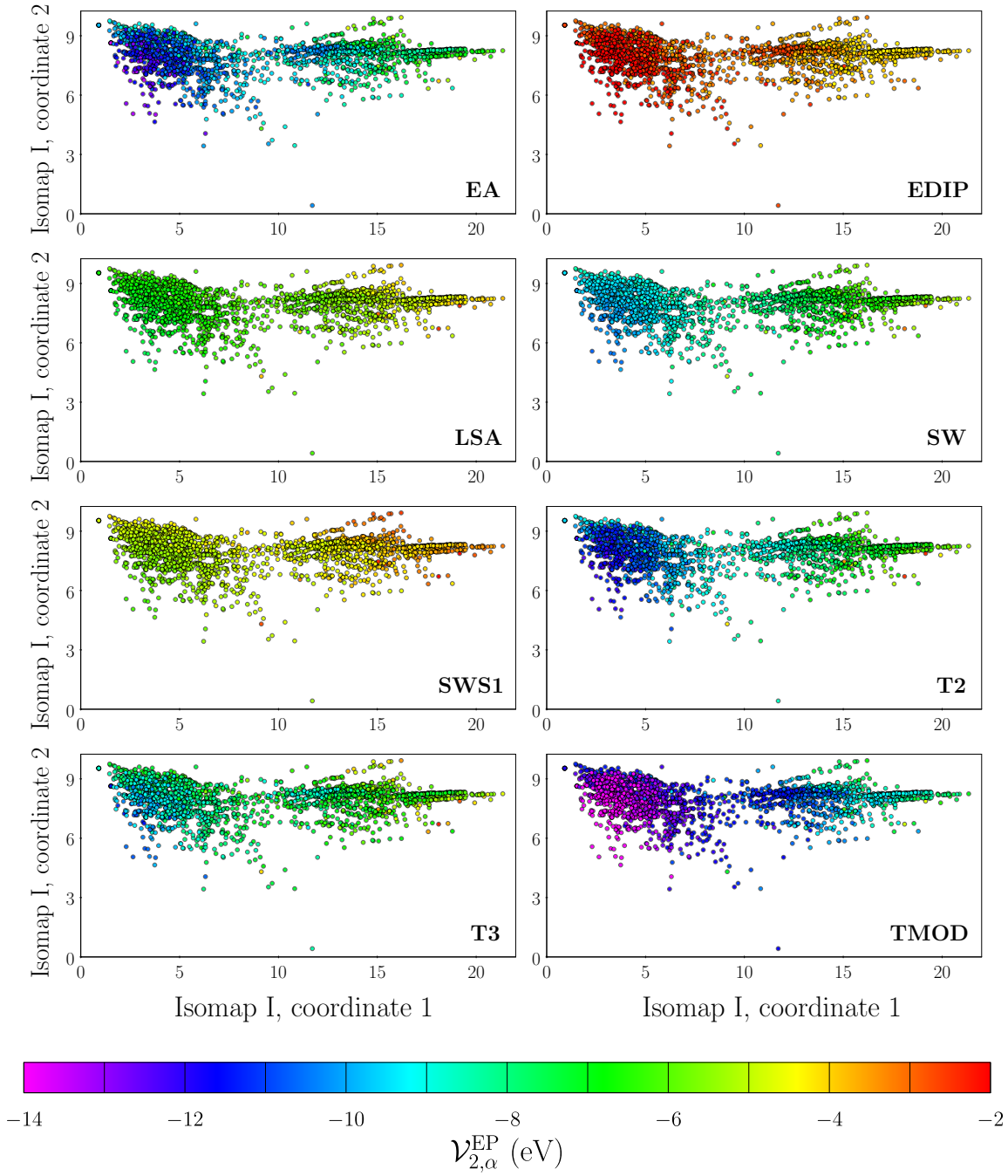


Figure E.3: Two-body contributions to the atomic energies defined in Section C.9 for each EP over isomap I, which contains the environments of the perturbed β-Sn, bcc, fcc, hcp, sc, and sh bulk configurations.

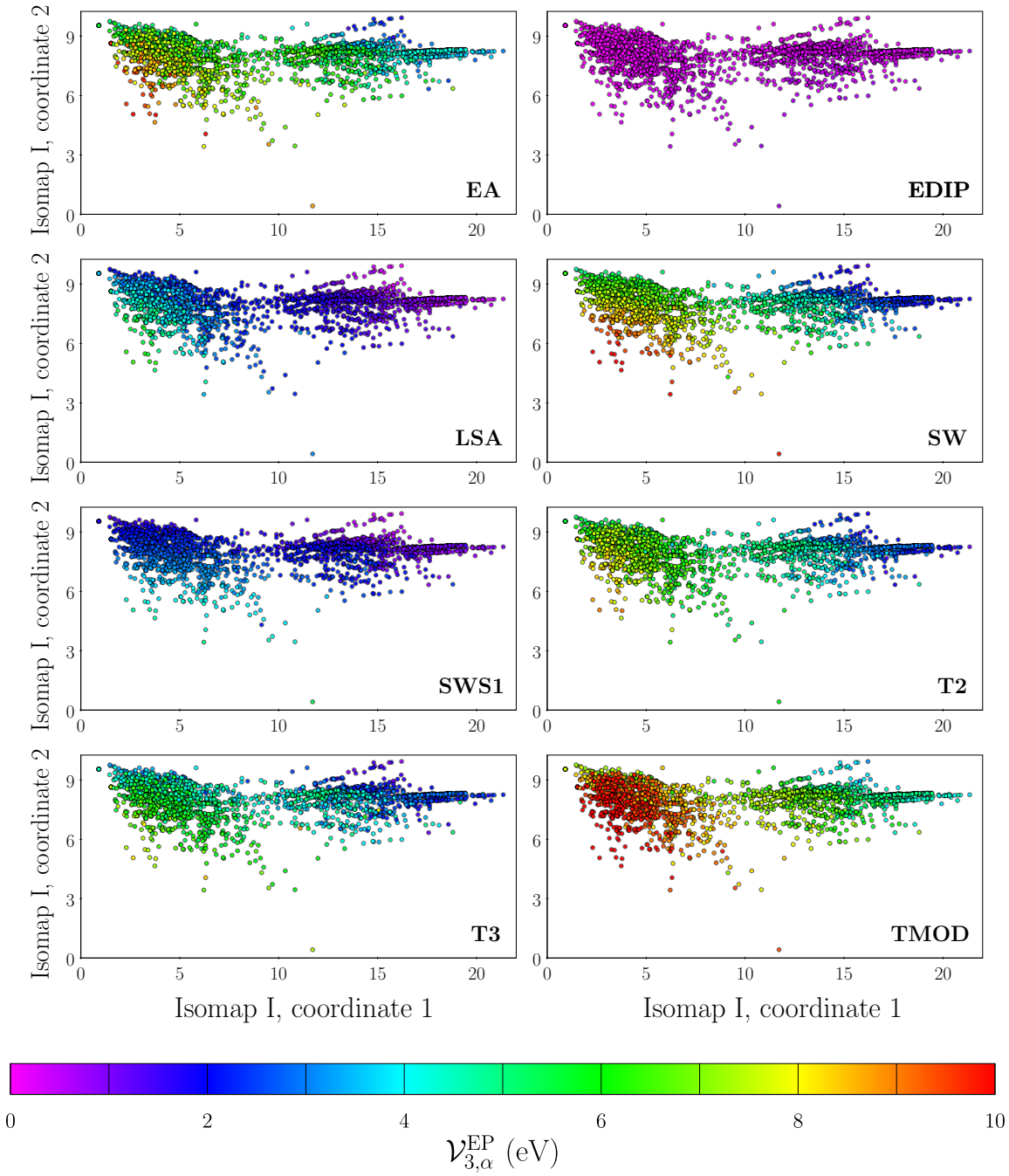


Figure E.4: Three-body contributions to the atomic energies defined in Section C.9 for each EP over isomap I, which contains the environments of the perturbed β-Sn, bcc, fcc, hcp, sc, and sh bulk configurations. The embedding energy $\mathcal{V}_{U,\alpha}^{\mathrm{EP}}$ is shown for the LSA potential.

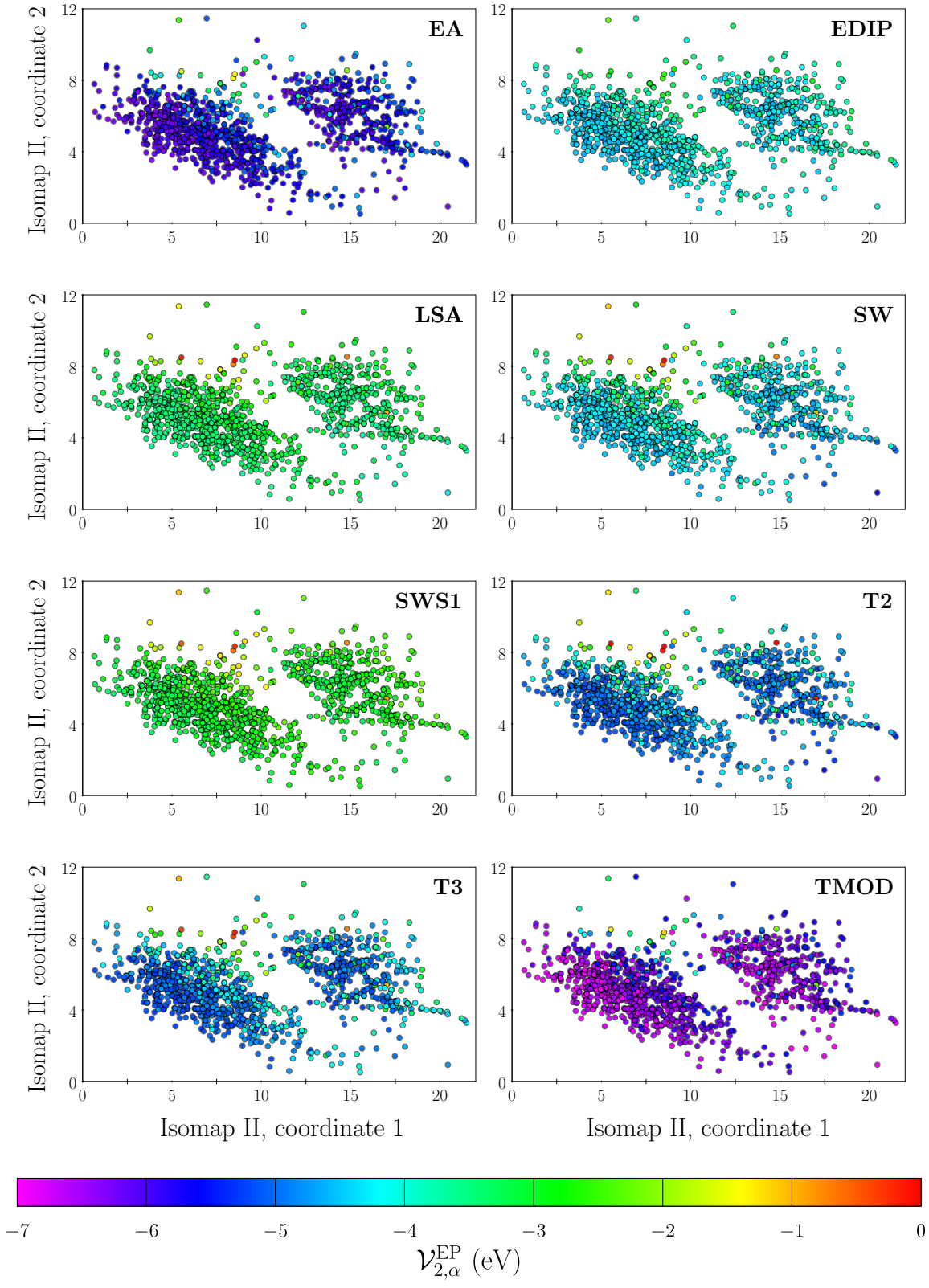


Figure E.5: Two-body contributions to the atomic energies defined in Section C.9 for each EP over isomap II, which contains the environments of the perturbed diamond, hexagonal diamond, and bc8 configurations.

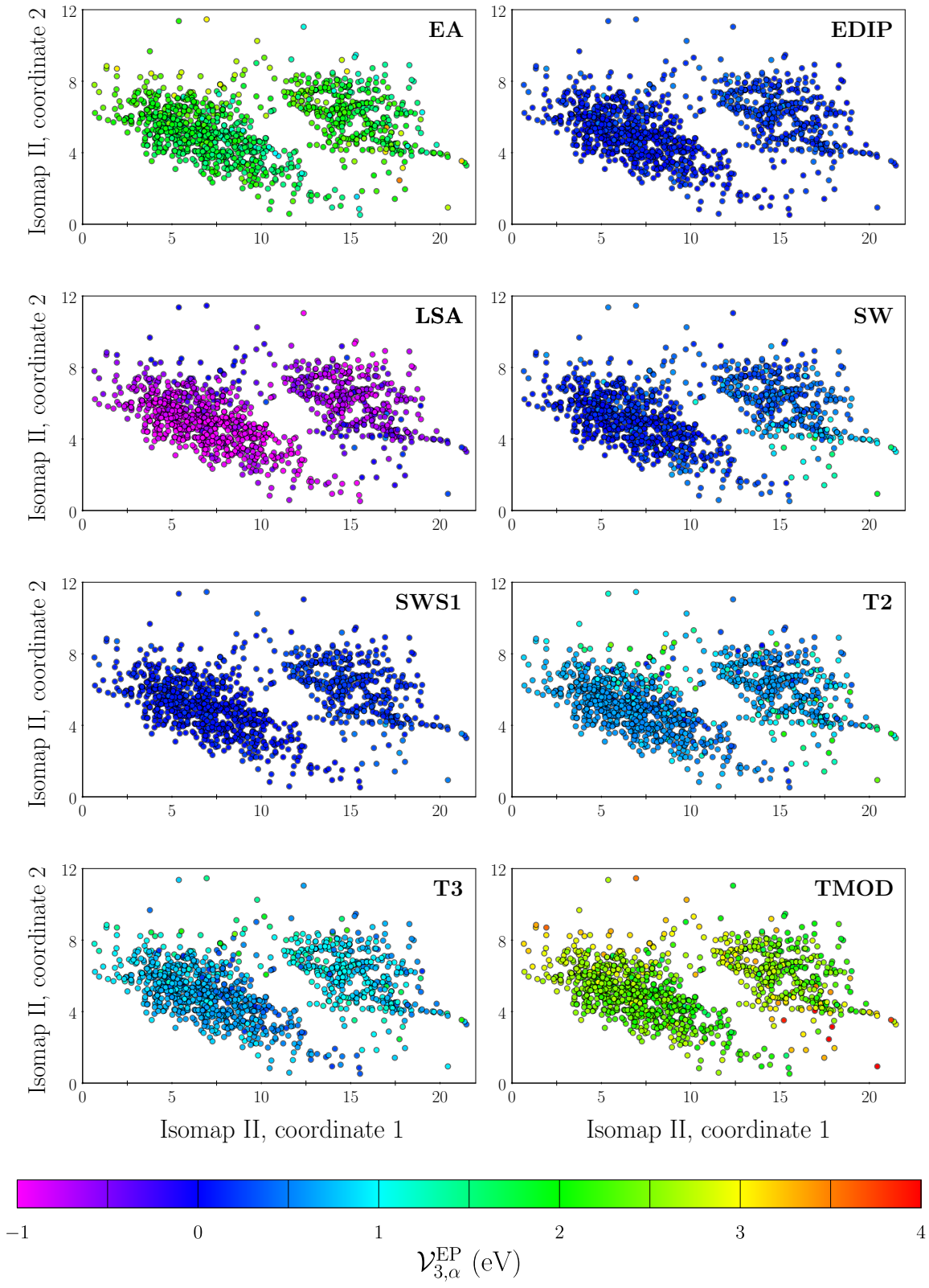


Figure E.6: Three-body contributions to the atomic energies defined in Section C.9 for each EP over isomap II. The embedding energy $\mathcal{V}_{U,\alpha}^{\mathrm{EP}}$ is shown for the LSA potential.

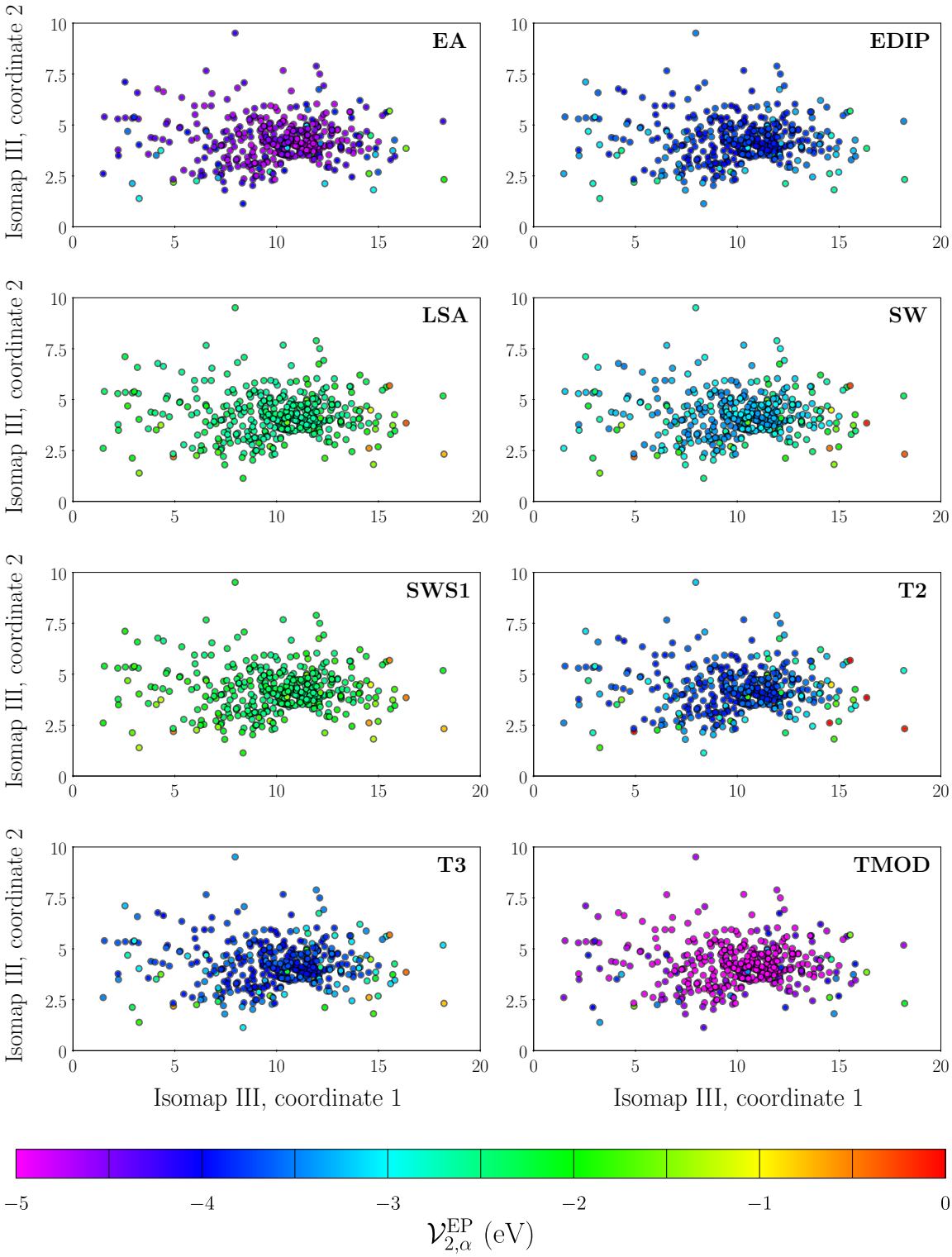


Figure E.7: Two-body contributions to the atomic energies defined in Section C.9 for each EP over isomap III, which contains the environments of the perturbed graphite bulk configurations.

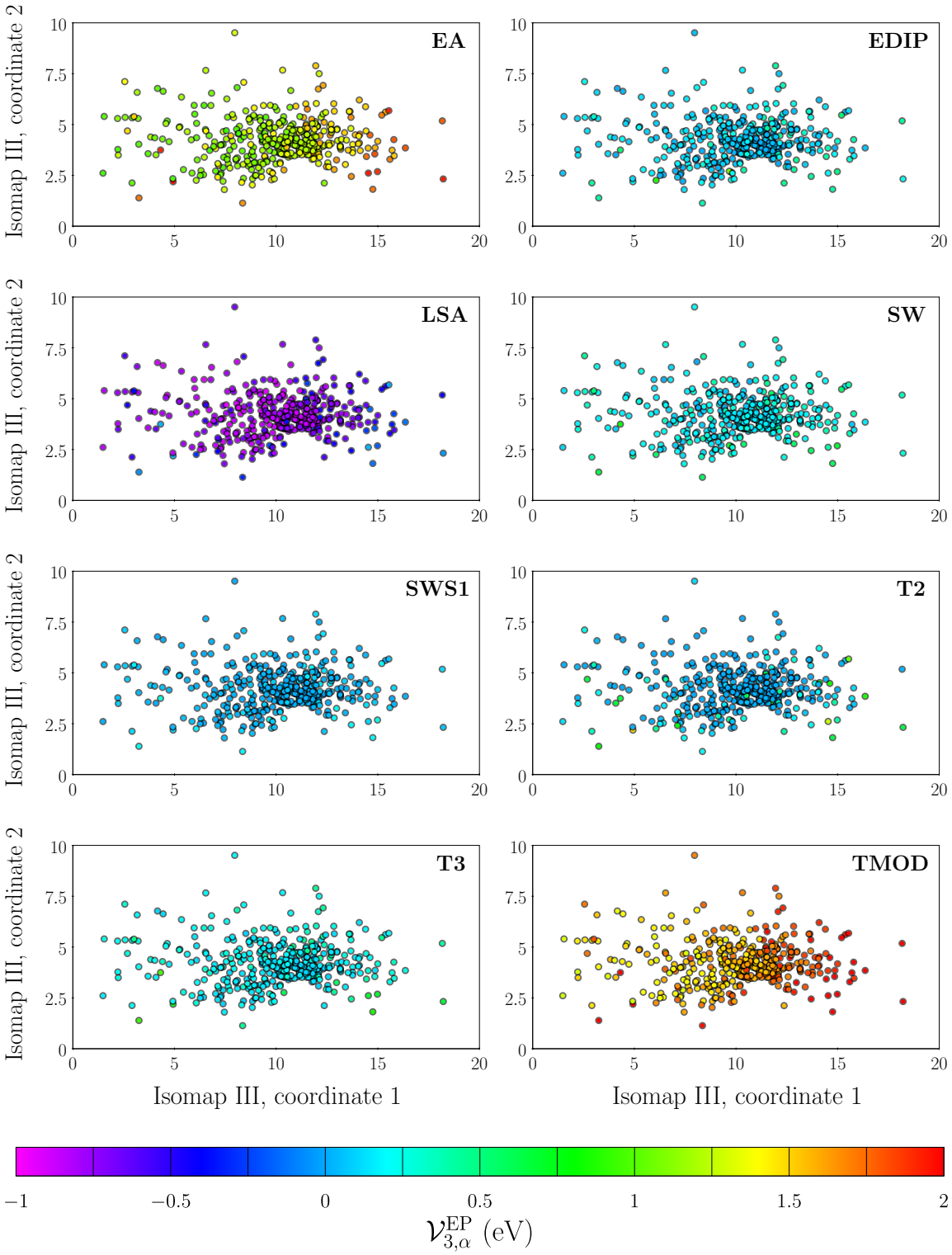


Figure E.8: Three-body contributions to the atomic energies defined in Section C.9 for each EP over isomap III, which contains the environments of the perturbed graphite bulk configurations. The embedding energy $\mathcal{V}_{U,\alpha}^{\mathrm{EP}}$ is shown for the LSA potential.

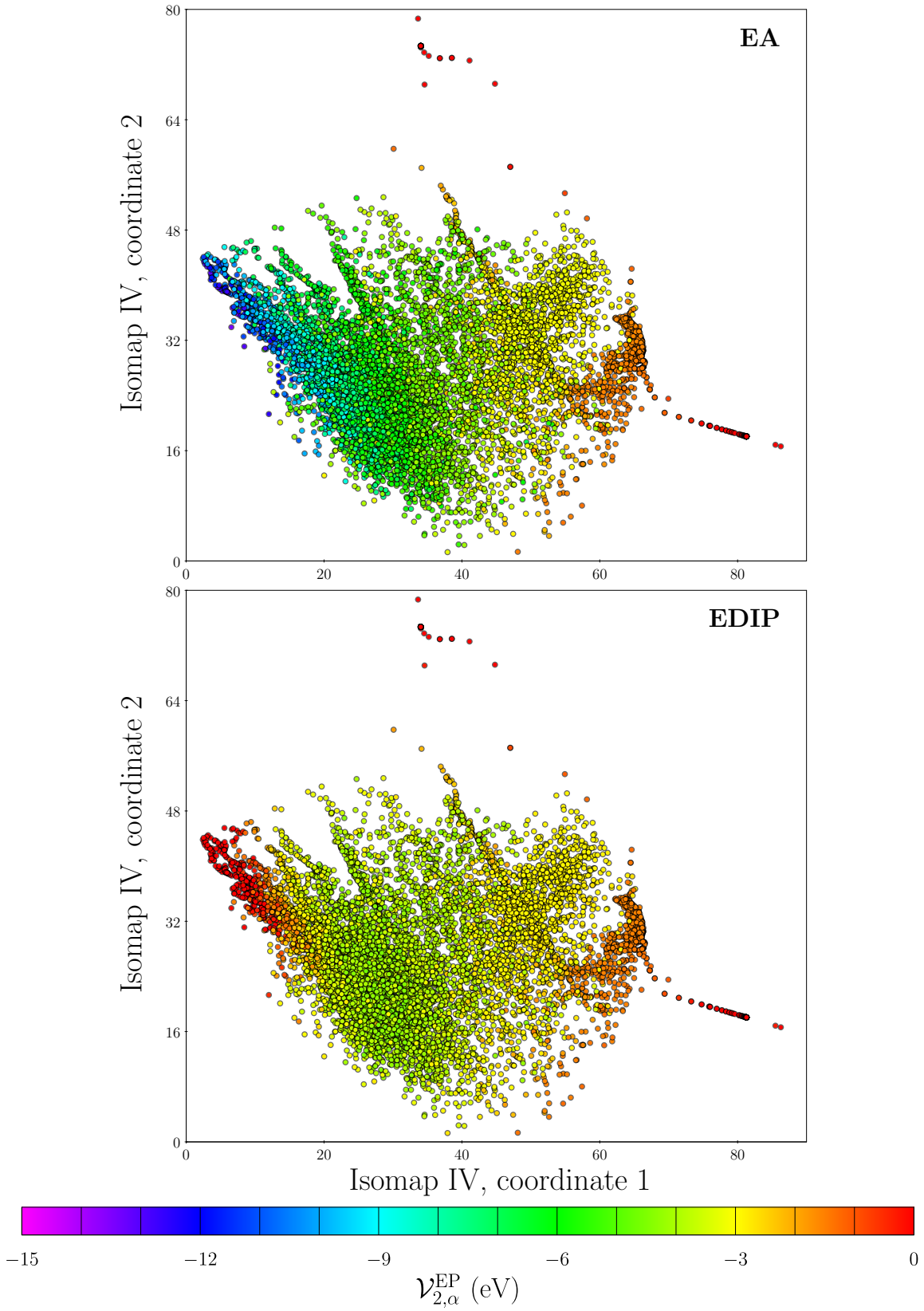


Figure E.9: Two-body contributions to the atomic energies defined in Section C.9 for the EA and EDIP potentials over isomap IV, which contains the environments of the cluster and dimer configurations.

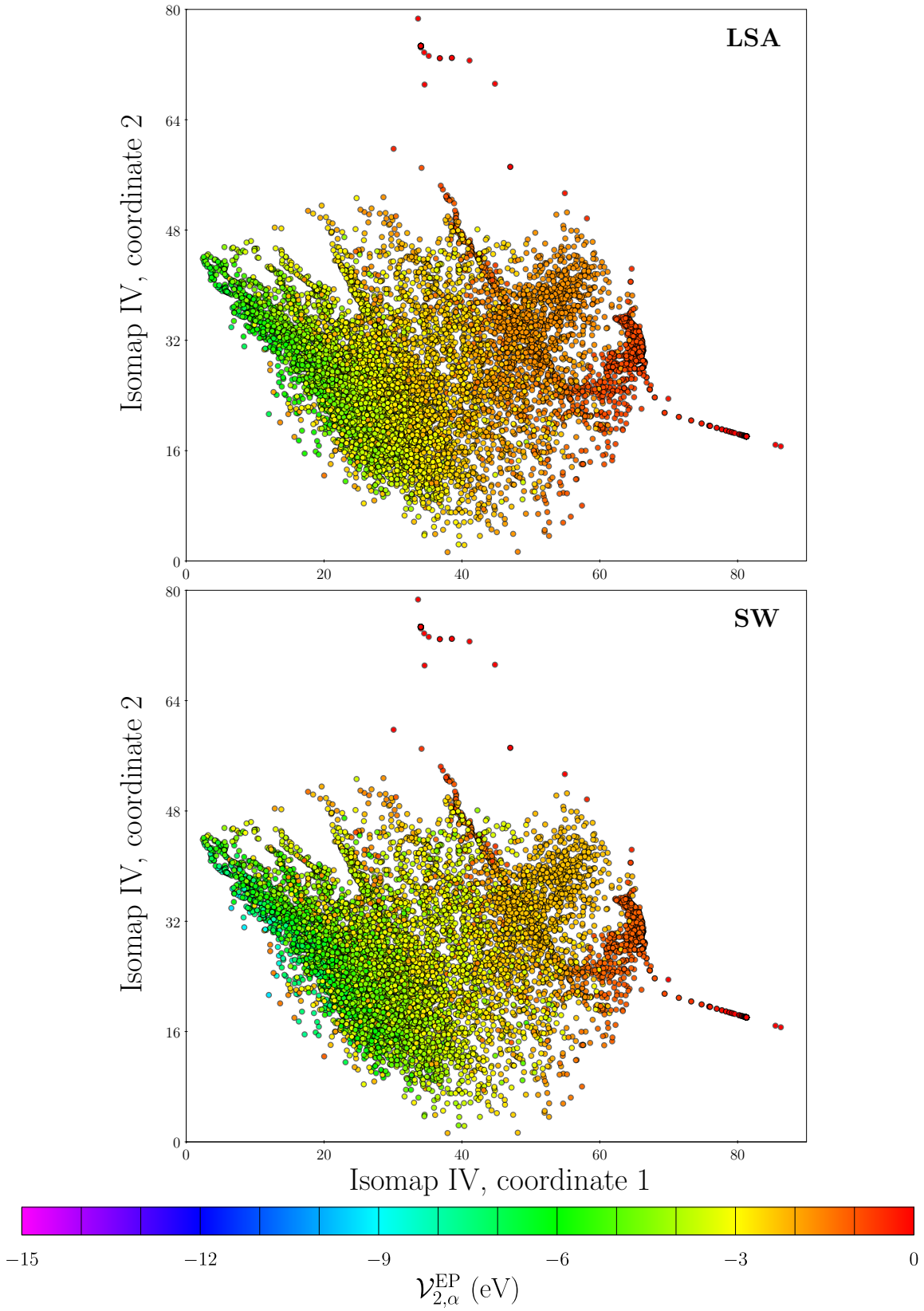


Figure E.10: Two-body contributions to the atomic energies defined in Section C.9 for the LSA and SW potentials over isomap IV, which contains the environments of the cluster and dimer configurations.

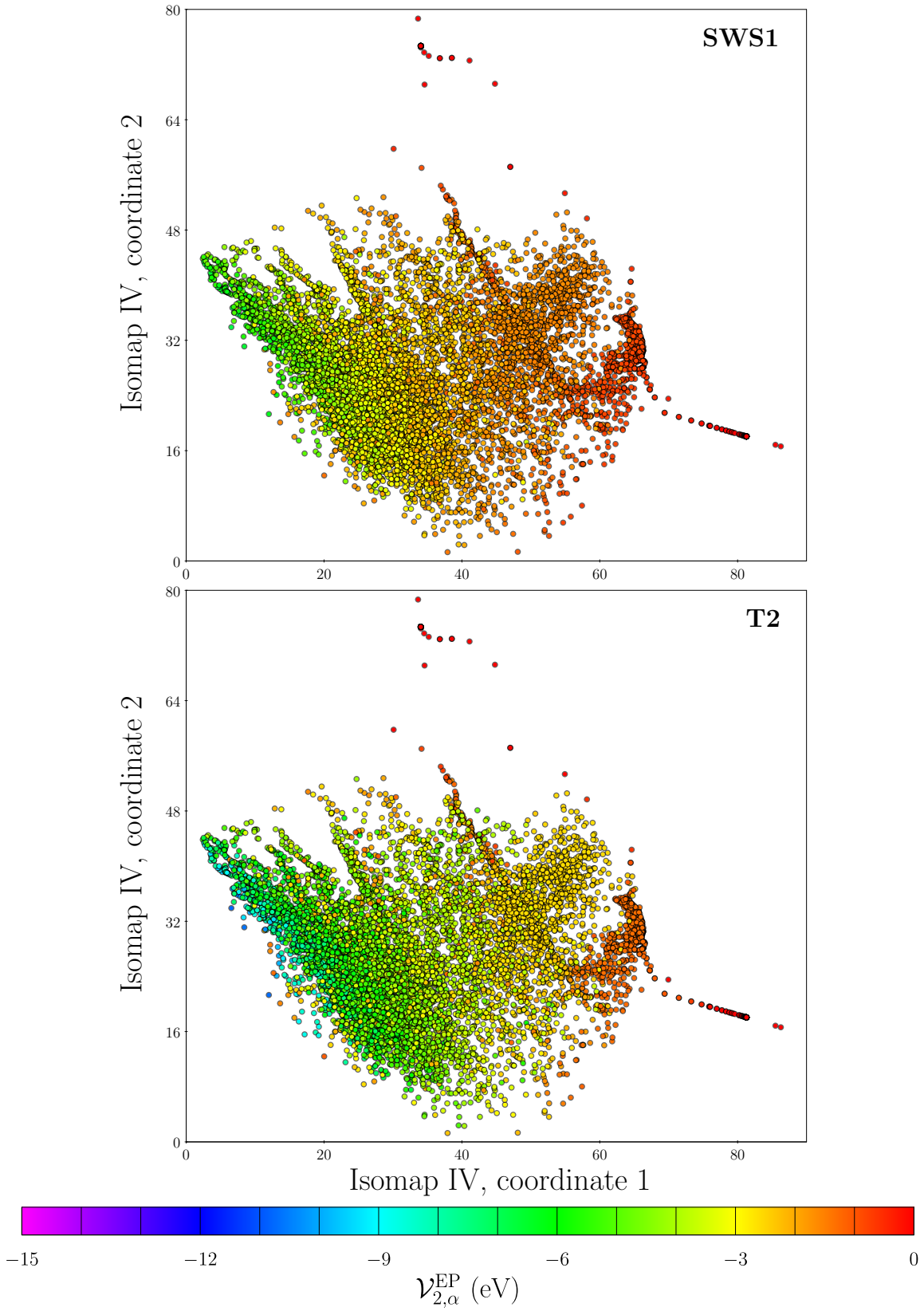


Figure E.11: Two-body contributions to the atomic energies defined in Section C.9 for the SWS1 and T2 potentials over isomap IV, which contains the environments of the cluster and dimer configurations.

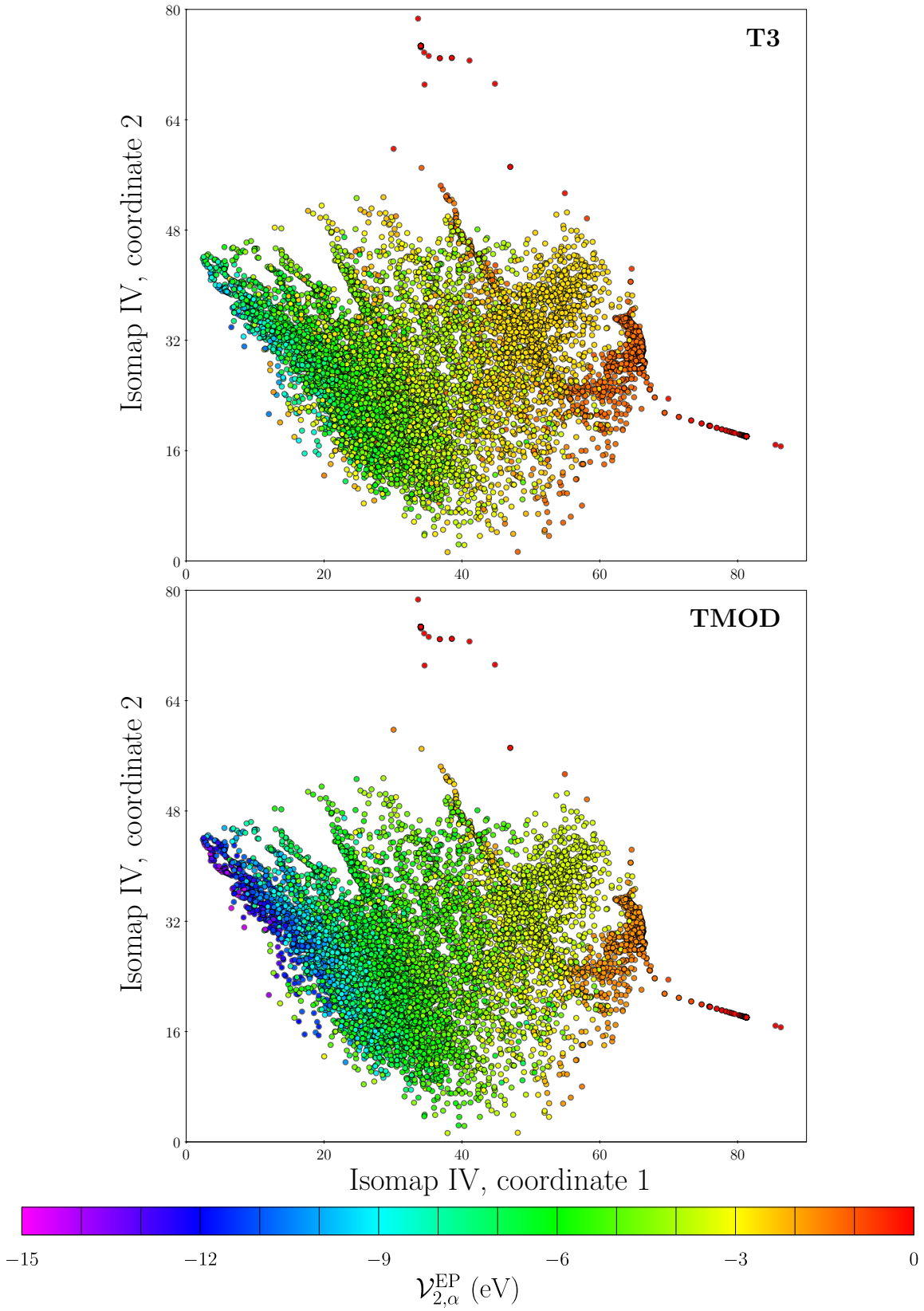


Figure E.12: Two-body contributions to the atomic energies defined in Section C.9 for the T3 and TMOD potentials over isomap IV, which contains the environments of the cluster and dimer configurations.

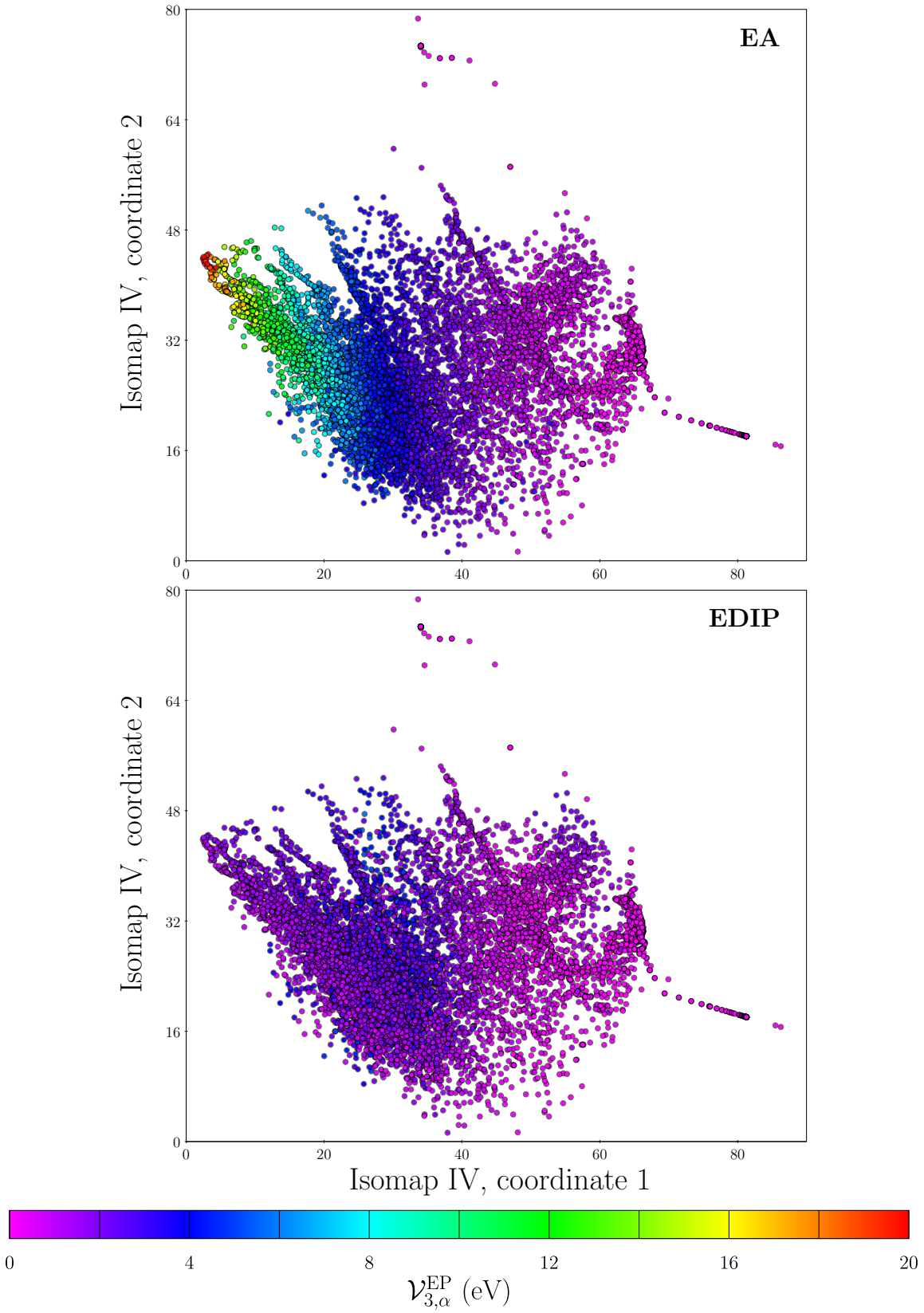


Figure E.13: Three-body contributions to the atomic energies defined in Section C.9 for the EA and EDIP potentials over isomap IV, which contains the environments of the cluster and dimer configurations.

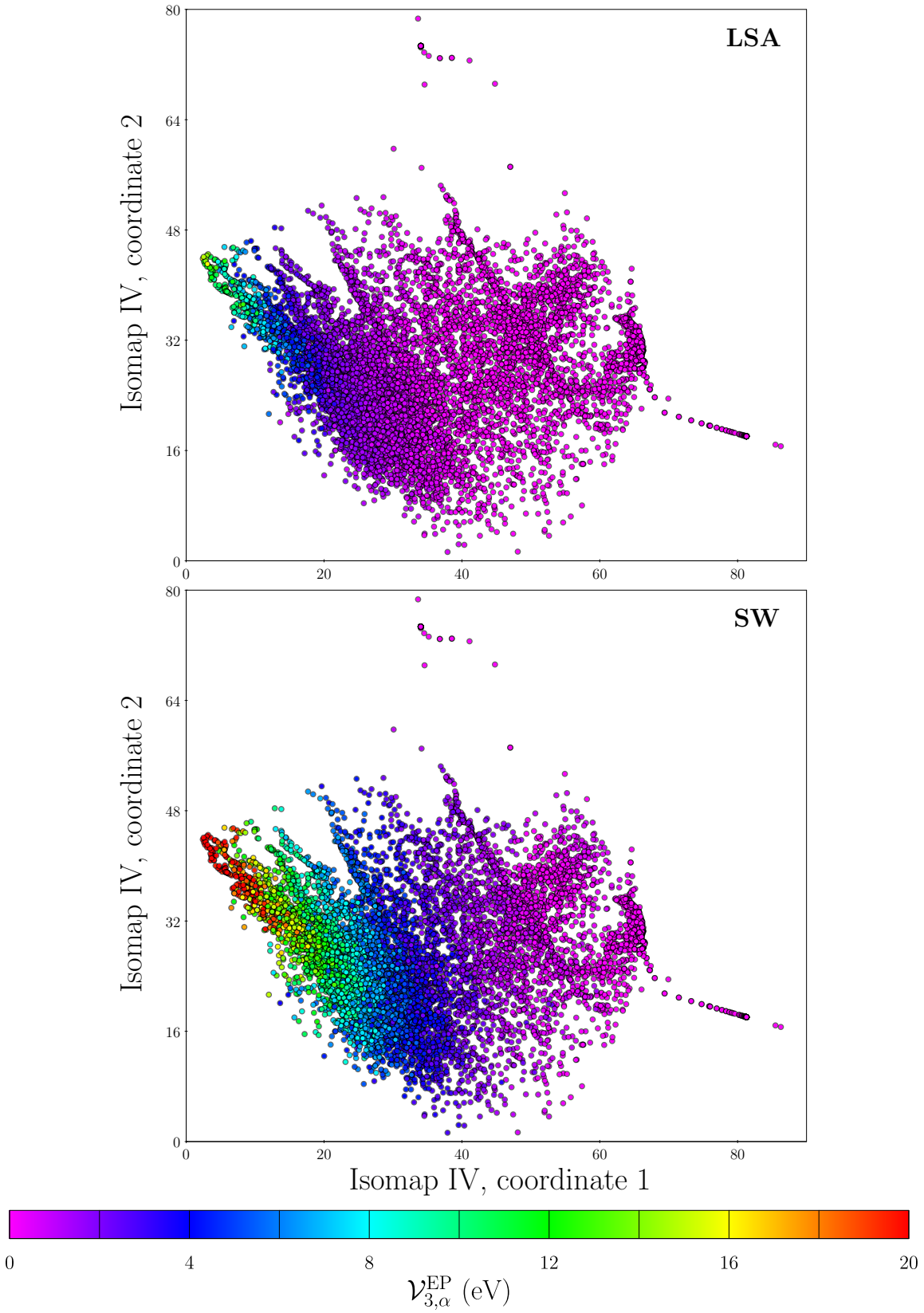


Figure E.14: Embedding energies of the LSA potential and three-body contributions of the SW potential defined in Section C.9 over isomap IV, which contains the environments of the cluster and dimer configurations.

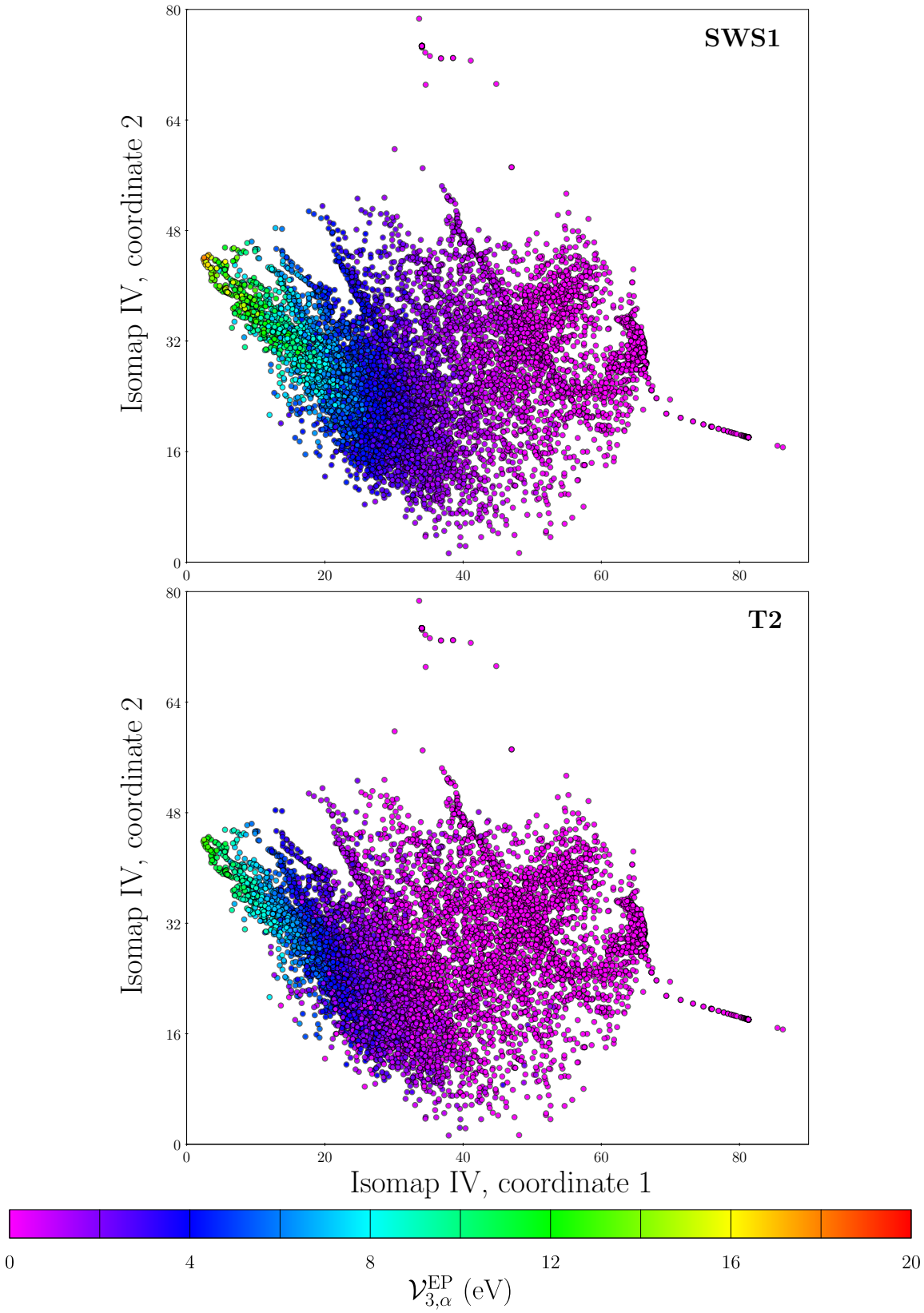


Figure E.15: Three-body contributions to the atomic energies defined in Section C.9 for the SWS1 and T2 potentials over isomap IV, which contains the environments of the cluster and dimer configurations.

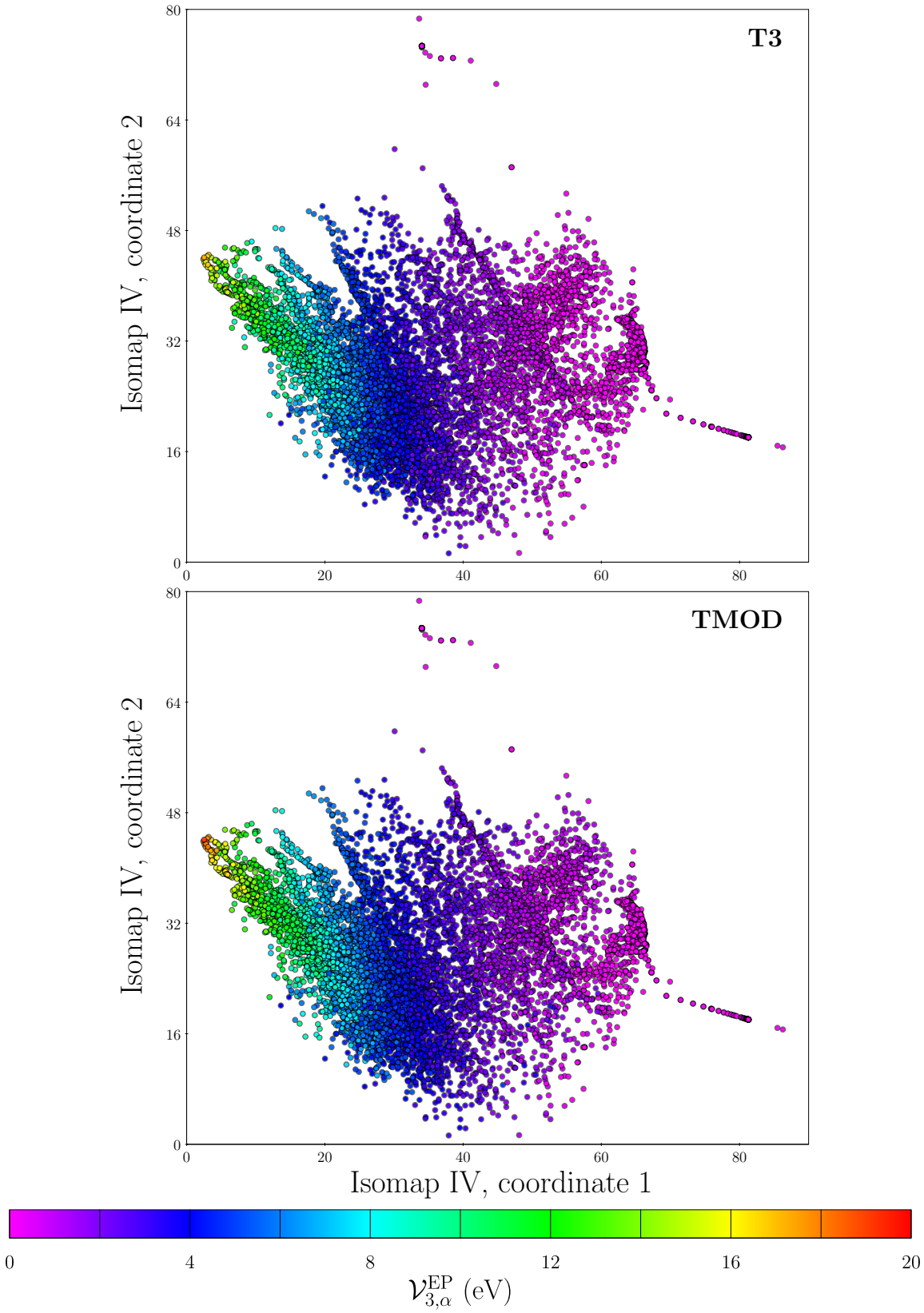


Figure E.16: Three-body contributions to the atomic energies defined in Section C.9 for the T3 and TMOD potentials over isomap IV, which contains the environments of the cluster and dimer configurations.

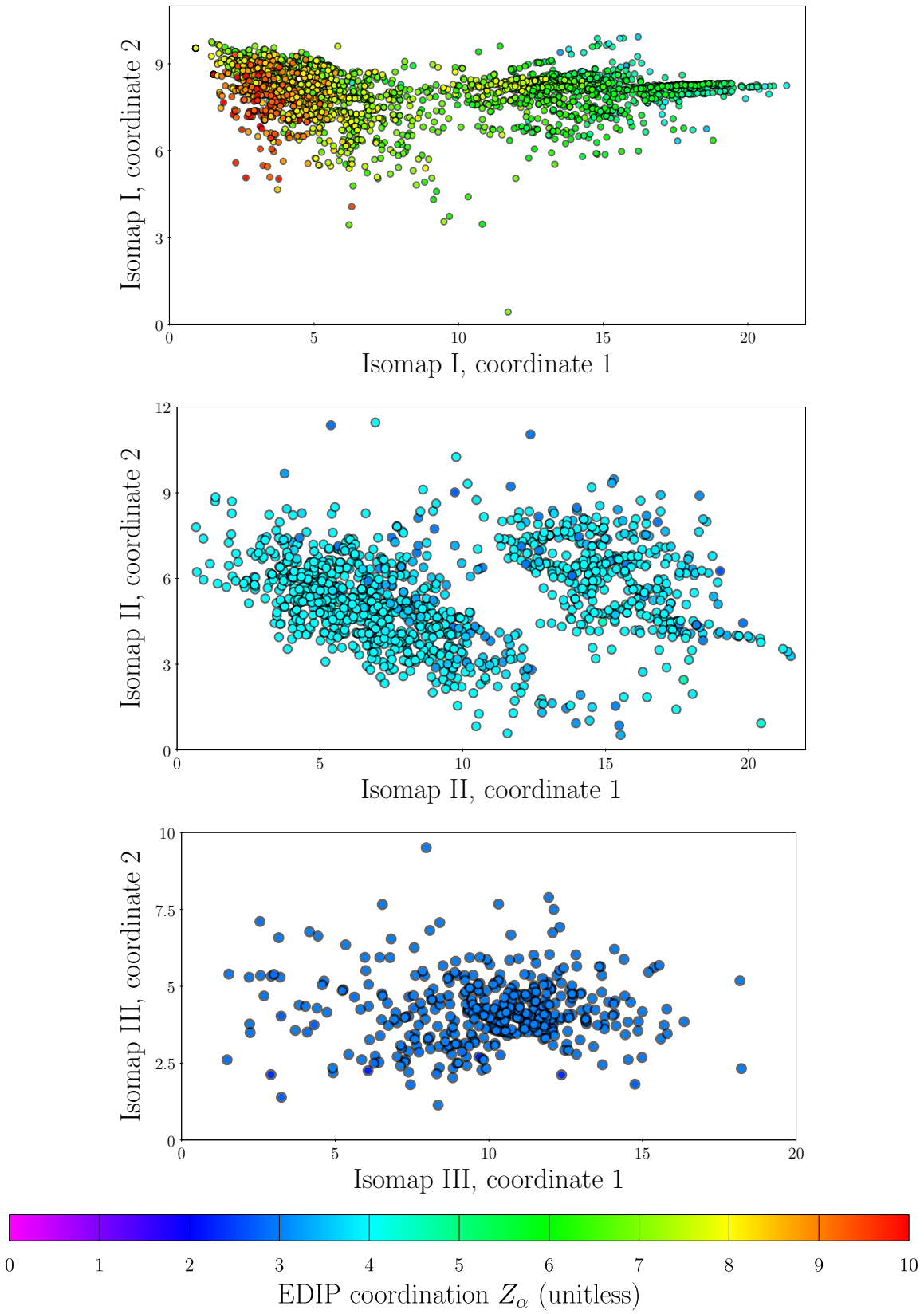


Figure E.17: Coordination parameter Z_α used in the EDIP potential displayed over isomaps I-III, which contain the environments of the bulk configurations.

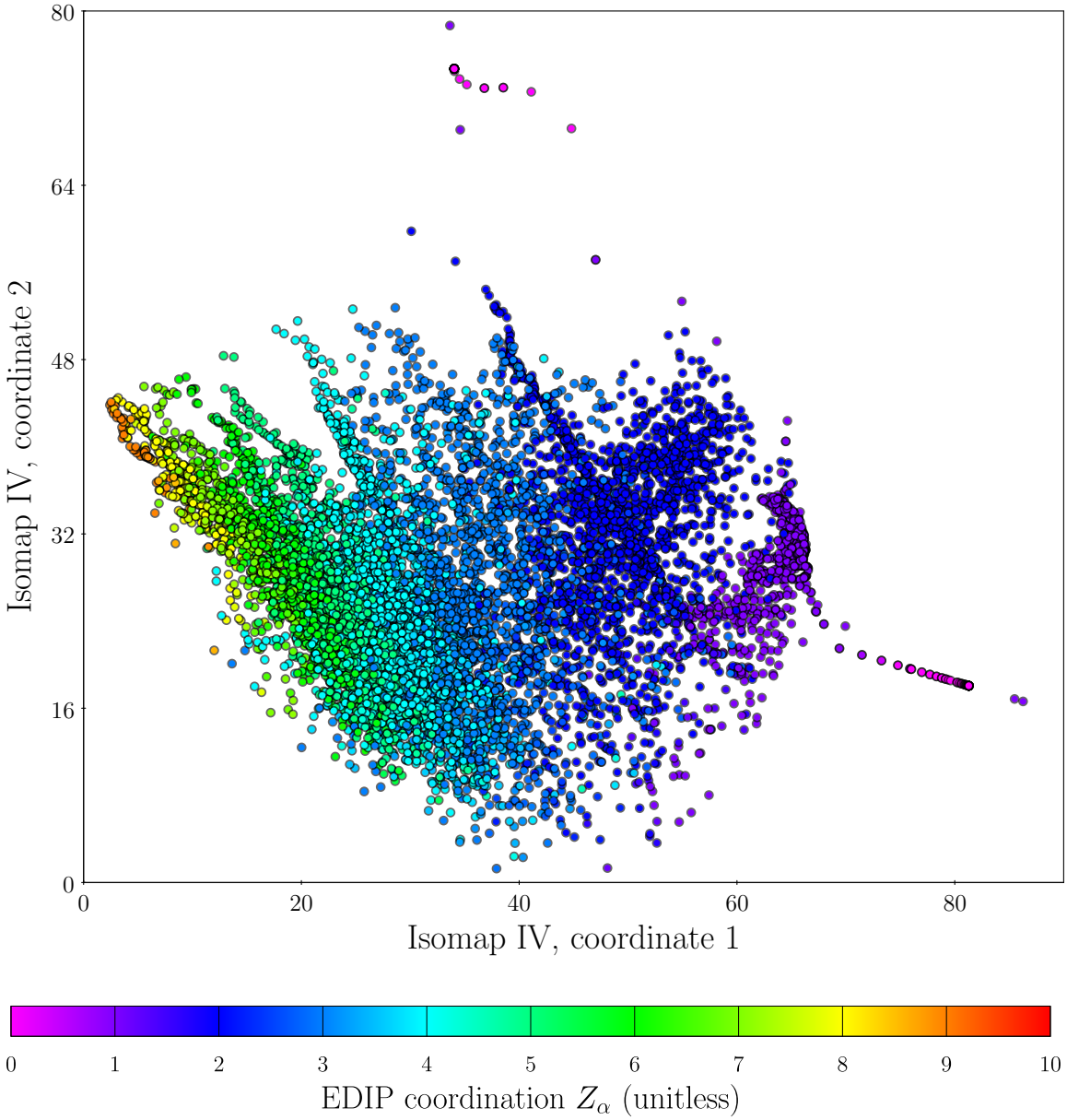


Figure E.18: Coordination parameter Z_α used in the EDIP potential displayed over isomap IV, which contains the environments of the cluster and dimer configurations.

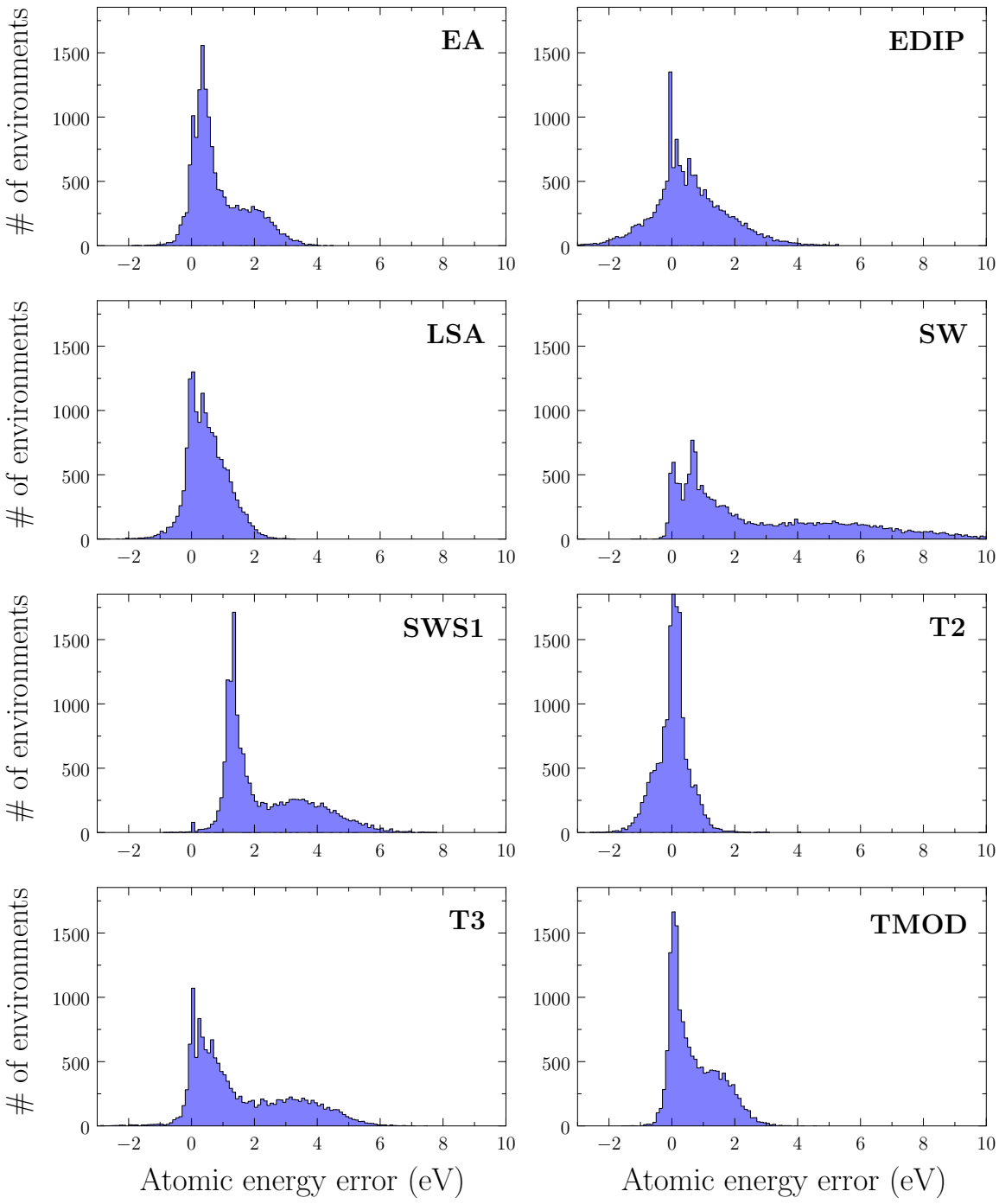


Figure E.19: Histograms of the atomic energy errors $\check{\varepsilon}_{\alpha}^{\mathrm{EP}}$ of the environments in the training set learned by RATE for each EP.

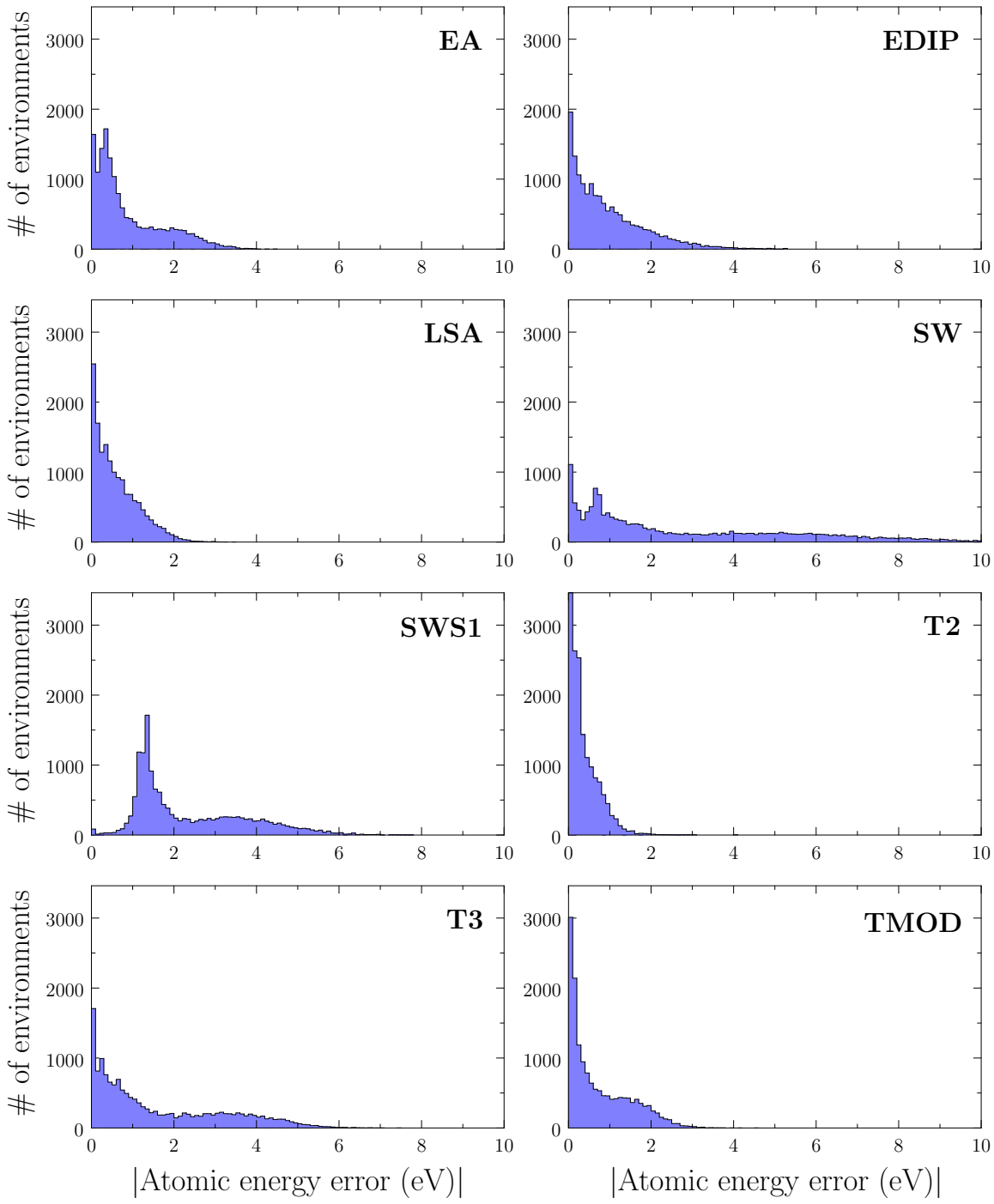


Figure E.20: Histograms of the absolute values of the atomic energy errors $\check{\varepsilon}_{\alpha}^{\mathrm{EP}}$ of the environments in the training set learned by RATE for each EP.

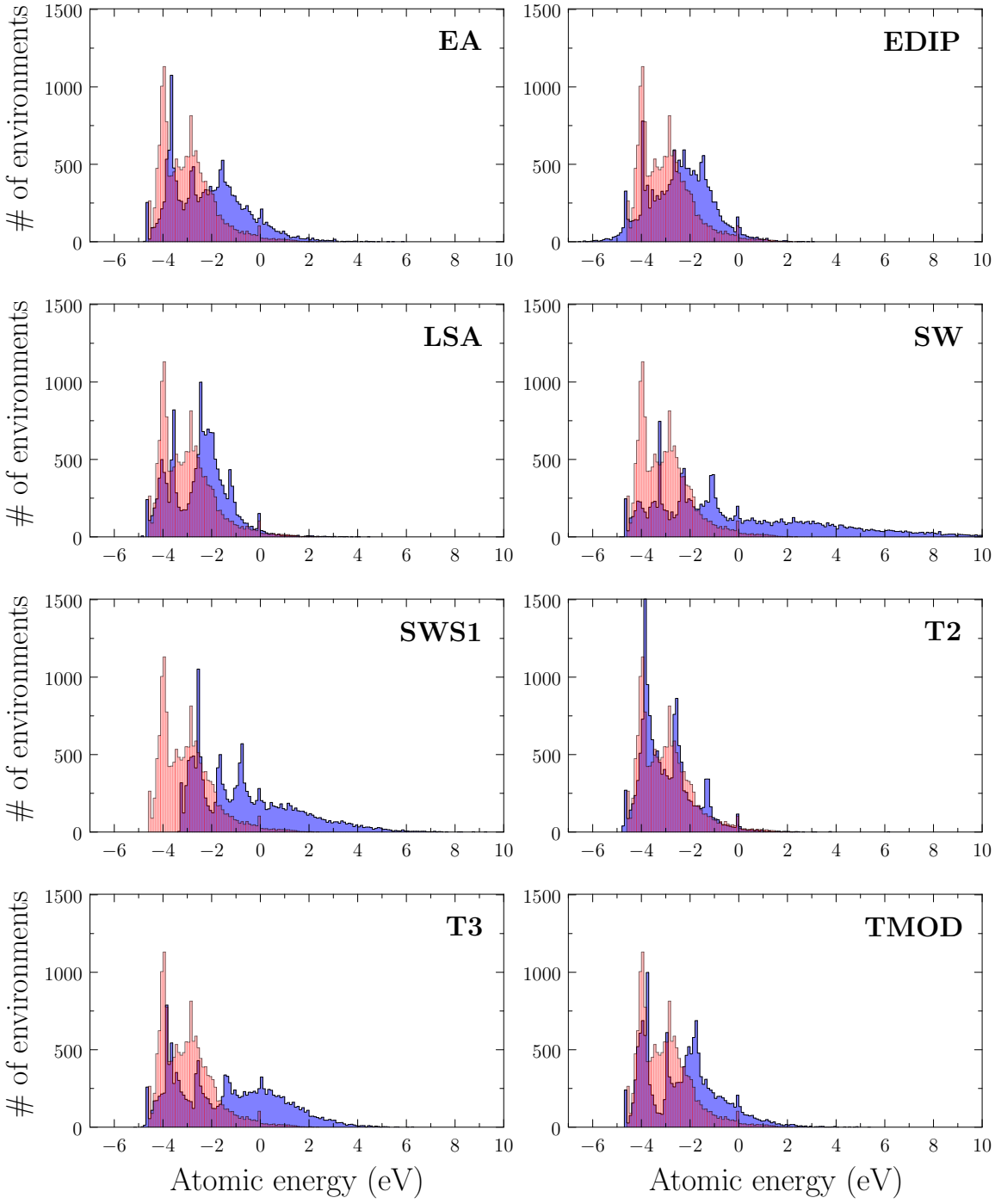


Figure E.21: Histograms of the atomic energies learned for each potential using the entire training set via regression. On each plot, the atomic energies $\varepsilon_\alpha^{\mathrm{DFT}}$ learned from the first-principles total energies for all atomic environments in the training set are overlaid transparently in red.

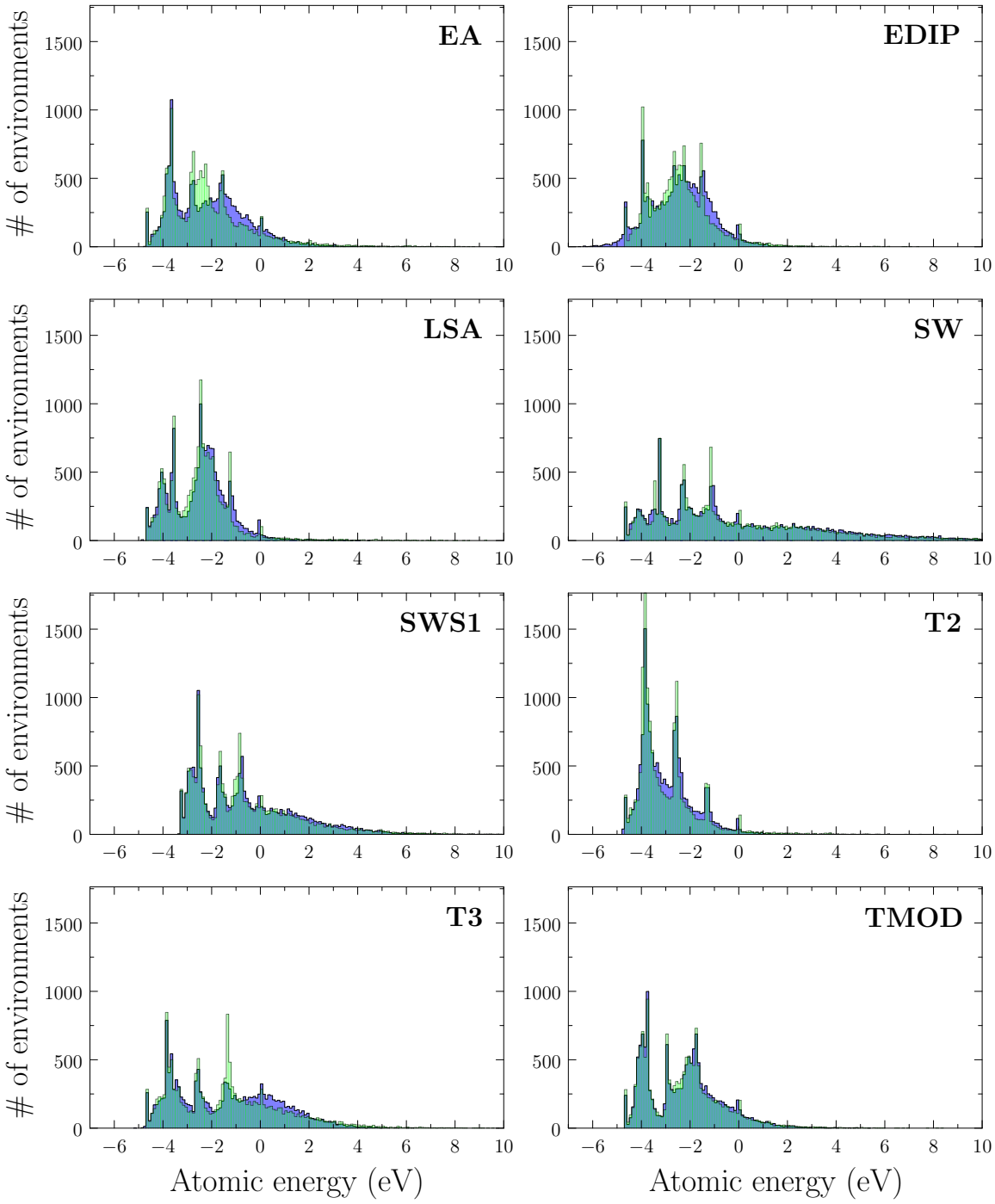


Figure E.22: Histograms of the atomic energies learned for each potential using the entire training set via regression. On each plot, the atomic energies $\mathcal{V}_\alpha^{\mathrm{EP}}$ defined in Section C.9 are overlaid transparently in green.