THE UNIVERSITY OF CHICAGO

ADVERSARIAL ANALYSIS AND MOLECULAR COARSE-GRAINING

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF CHEMISTRY

BY
ALEKSANDER EVREN PAETZOLD DURUMERIC

CHICAGO, ILLINOIS
MARCH 2021

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

When I began graduate school I unsure as to what to expect in terms of support. As my family, friends, and mentors can attest to, I unhappy being told what to do, and this has often resulted in disagreements. Their continued support is more so a testament to their patience than my skills in conflict resolution. I have since realized that I was already quite well supported before starting and continued to be so while I completed my work.

First, I thank my family and friends. Their presence alone has gotten me through much of my time here, but more importantly has provided the opportunity to make it enjoyable. I am sure at this point that even if I had been able able to persevere alone it would not have been worth it. If you spent time with me, you helped me through this.

Second, I thank my advisor Gregory A. Voth, as there have been stretches of time where I have had little results to show. It is lost on me why I was given the freedom I was to do what I wanted. I am sure that many mentors would not have done so. I also thank the rest of my committee, Andrew Ferguson and Aaron Dinner, for being willing to entertain my writing and presentation style. I also thank my collaborators for being willing to converse to me.

Third, beyond their role as my friends, my coworkers have provided the conceptual backdrop for the work I present here and the researcher I have become. I have considerable gratitude to John Grime, James Dama, Glen Hocky, Alex Pak, and Andrew White (and others who I am clearly forgetting) for providing day to day mentorship at one point or another throughout my graduate career.

ABSTRACT

Models are common in chemistry. When these models can be described mathematically, their real world implications can often be simulated using computers, enabling the use of more complex models in hopes of improving scientific predictions. Prior to providing useful results these models must often be calibrated against existing scientific data. Separately, machine learning has recently gained significant traction in many applications. The algorithms underpinning machine learning often similarly require calibration prior to application. This work provides mathematical and numerical results connecting these two areas. Specifically, we consider novel applications of classification when creating molecular models such as those used in coarse-grained molecular dynamics. We focus on the concept of adversaries, a tactic that has recently gained traction in the machine learning community, and use this framework to analyze the difference between various coarse-grained ensembles and to parameterize new coarse-grained force-fields. Collectively, we show that classification is an effective tool for understanding the differences between high dimensional free energy surfaces and that adversarial parameterization strategies are theoretically and numerically feasible for coarse-grained models.

CHAPTER 1

INTRODUCTION

Models are fundamental to everyday life. Suppose you are cooking and you realize you need hot water. A reasonable next step is to put a pot of water on the stove and turn on the stove's heating element; this is done with the expectation that the water's temperature will increase. Here, we consider the expectation that that putting water on the stove will increase its temperature a model. This expectation is clearly a simplification of any physical mechanism that underpins the act of heating the water— no mention of molecules or the workings of the stove is present. However, the model is useful as it provides a prediction: the water will get hotter if we perform certain actions. In this way, a model is not necessarily embodied in equations; instead, it is simply a way to refer to an interpretation of events. With this definition of a model in hand, it is unsurprising that science is intertwined with these constructs: If science hopes describe how the world works, models seem a natural place to start.

Some models, especially those with quantitative predictions, require additional information to be useful. For example, we may want to know how long said stove will take to heat up our water. This naturally depends on the amount of water being heated, the temperatures considered, and the details of the stove. If we assume that the temperature of the water changes linearly as a function of time we can use our experience to predict how long it will take to heat up water to a given temperature. This is the idea of calibration: some models need to be "trained" to mimic the data we already have in order to be accurate. Our assumption of linearity implies we only need two parameters (the slope and intercept of our line) to describe our stove's behavior. Importantly, however, our model gives us more than just the prediction of the time needed for heating: it characterizes the stove itself.

If temperature goes up more quickly for a particular stove (this is characterized by the slope of our linear relationship, one of our parameters), it would be natural to say that said stove is more powerful. In other words, our model provides both predictions and understanding, and these are not the same thing. If we relate the intercept of our line to the initial temperature of the water we can go further: between different stoves our slope will likely change, but the intercept will probably only depend on our initial water temperature.

Importantly, however, our linear model is approximate. A better estimate could, for example, involve an exponential fit as suggested by Newton's law of cooling. If we base our modeling on the assumption that the true phenomenon is linear, any two distinct observations should provide the same inferred linear model. However, acknowledging that we are linearly approximating a more complex relationship implies that we must also specify how we want to estimate said true relationship. Given a set of observations we wish to approximate, a linear model parameterized to never overestimate the amount of time needed to heat our water will likely differ than one parameterized never to underestimate the same value. The rules we use to parameterize, or train, our model determine the types of errors the approximation entails. Furthermore, understanding the types of error made by a particular model is critical for being confident in its application: if we do not want to accidentally overheat our water, overestimating and underestimating the heating time naturally have different consequences.

At first this description of a model likely seems clear, if needlessly verbose. However, models change as they become more complex. This is quite evident when considering algorithms in machine learning. Our previously described model of our stove had two parameters (the slope and intercept); a modern neural network that can be used to discern the content in an image has approximately 25 million[1, 2]. The neural network's parameters do not easily lend to physical interpretation. The predictive value of the neural network is clear: for example, it can tell us if a dog is present in a picture with startling accuracy. However, it does not provide understanding: there is no slope, no intercept, that we can singly point at as representative of part of reality. Furthermore, calibration of said neural network requires more than a million data points— a far cry from the

minimal data needed to specify our linear model. Methods from machine learning have nevertheless found application in the molecular models common in chemistry. Understanding their application, however, first requires us to explain the corresponding molecular models in detail.

1.1 MOLECULAR MODELS

As alluded to previously, chemistry has a history with models. Chemical physics undertakes the task of using physics at the atomic scale to understand chemical phenomena. This task is difficult: electrons and their quantum mechanics aside, most chemical phenomena involve many atoms. Understanding why, for example, a bubble nucleates in a liquid involves the simultaneous behavior of these many atoms. And yet, if we include too many variables, our model will not be interpretable and will be difficult to parameterize. In order to satisfy our need for transparent models we focus on simple explanations, and indeed, classical nucleation theory (CNT)[3] describes such a many-body event without worrying about the behavior of each atom individually. These models have worth in both their predictive value and their chemical insight. Their transparent nature also makes it clear when they break down. For example, despite CNT's success in describing certain cases of vapor-liquid condensation, its central assumptions are qualitatively incompatible with crystal nucleation[4].

Sometimes, however, scientific applications do not mesh well with simple explanations. One pertinent way this arises is when we want a single parameterized model for a large number of different phenomena. For example, while the macroscopic progress of a particular chemical reaction can be predicted by using a small number of experimental measurements combined with simple differential equations, models which understand the motion and interaction of individual atoms can aim to characterize rates of many reactions after a single calibration step is performed[1]. Molecular dynamics (MD)[5–10], an example of such a model on the atomistic scale, is central to the studies presented in this work. MD computationally simulates the motion of each individual atom

[1]For certain types of models (generally referred to as *ab initio*, although this term is used loosely) no calibration is necessary. The domains and time/space -scales we discuss in this work do not have applicable models of this type, usually due to computational considerations.

present in the system under study, often scaling to millions of atoms in each simulation performed. These simulations provide a large number of sequential discrete snapshots, or configurations, of the modeled atoms evolving in time; these configurations are then analyzed statistically to provide macroscopic predictions. MD is primarily parameterized by controlling how atoms interact with each other: e.g., how atoms are chemically bonded and the large scale interactions characterizing nonbonded behavior. Collectively, these interaction rules are referred to as a force-field. While force-fields must still typically be calibrated to match experimental data, atomistic force-fields are stable over a large range of phenomena, which allows simulations to be constructed for a wide variety of physical settings using previously calibrated force-fields.

One one hand, these atomistic models do not have to be adjusted for each scientific question they are used to answer; models which do not require reparameterization can be used to predict results with less experimental data. On the other hand, this same property has the downside of obviating parameter inspection as a method of gaining insight. Fortunately, MD introduces a new avenue for interpretation: as these models simulate phenomena at the atomistic resolution, the physical atomistic mechanism underlying experimental predictions can be directly observed. That is, instead of considering learned parameters, the individual motion of atoms can be visualized and interpreted.

The atomistic description provided, however, reintroduces much of the problematic detail we sought to avoid in scientific models[11]: each snapshot produced by MD can be composed of the positions and velocities for over a million particles. This high dimensional data has provided many opportunities for analysis, ranging from techniques derived solely from classical statistical mechanics[7, 10] to the derivation of new reduced models that approximate and describe the observed simulation data. While the first case is extremely pertinent for connection to experimental measurements, the second category is more important to the work presented here, and has included a large number of techniques such as machine learning based dimensional reduction[12–14], the creation of effective Markov models[15–18], and the design of new MD models at a reduced physical resolution. These reduced models do not replace atomistic MD— after all, they post-process data from MD for their parameterization. Instead, they are trained to describe the patterns present in

MD data, and are often subsequently inspected similarly to how the parameters in our stove model were interpreted. One class of these models is central to the work at hand: the creation of new MD models at a coarser resolution. The approach of using the results of atomistic MD to parameterize coarse-grained (CG) MD, however, is best understood in a broader context.

Atomistic MD is computationally expensive[9, 11, 19–24]. Significant effort has been put forth to reduce its cost; however, state of the art simulations can only simulate small solvated proteins for a matter of milliseconds[8] or billions of atoms at significantly shorter timescales[25], and many researchers do not have access to computational resources necessary to perform such simulations. MD simulation cost is strongly related to the physical dimensions of the problem considered as it proceeds iteratively through time: at each iteration the forces on every atom simulated are calculated; this information is then used to update the position of each atom as time evolves. A large number of atoms creates a more complex force calculation; a longer timescale necessitates more iterations.

Unfortunately, the scale of many biological processes easily exceeds the scales accessible to atomistic MD: for example, the reproductive processes of the HIV-1 virion occur on the timescale of tens of minutes and involve more than one billion atoms[26–28]. Reducing the cost of these simulations either reduces the number of needed iterations or reduces the number of particles present[24]. The first category provides methods critical to contemporary MD studies; however, the effective CG MD models alluded to in the previous paragraph are related to the second category[2], and are the topic of the current work. While an atomistic (or all-atom, AA) MD simulation considers the time evolution of every atom, CG MD simulations consider the evolution of a reduced description of the system. For example, an atomistic simulation of a protein could consider every atom of the specified protein, the water solvating it, and the ions present in the water. A CG simulation could reduce the cost of this system by only considering the center of mass of each amino acid present in the protein. If the conclusions that are to be drawn from the simulation can be made using only this reduced representation, it would then seem appropriate to by default use a CG model. However, as

[2]CG MD models also reduce the number of needed iterations; however, this effect is not often explicitly incorporated into the methods designing these models, and we do not discuss it in this document.

one might expect, this decision can have problematic consequences.

A large number of contemporary techniques exist to design a CG MD simulation [11, 13, 14, 21, 28–37]. Broadly speaking, the rules one uses to define the (effective) interactions among CG particles are a matter of debate. Unlike atomistic force-fields, CG force-fields often do not produce accurate results over a broad range of phenomena; this is referred to as the issue of transferability. Nevertheless, one straightforward option is to treat the creation of a CG force-field similar to atomistic force-fields: in this case, a set of rules describing the force-field for CG molecules is fit to known experimental values or chemical structures. This approach has an extensive history relating back to the theory of simple liquids, where radial distribution functions (a measurement that can be gleaned from both experiment and simulation) were used to create pairwise interactions describing liquids[29]. The information used for parameterization can either be specific to the particular system of interest or may pertain to multiple systems in hopes of producing a broadly transferrable model. Unfortunately, using information solely related to a particular system requires reparameterization for each physical setting considered, removing the advantage seen with generally applicable atomistic force-fields. On the other hand, issues with the intrinsic achievable transferability of CG force-fields suggest that parameterizing against the properties of multiple systems is at odds with accuracy in any given physical setting.

Physical source of the reference data aside, many CG force-field parameterization strategies aim to reproduce the results of experimentally viable measurements. This approach encourages force-fields to match scientifically pertinent properties; however, multiple force-fields may match the obtained measurements creating ambiguity in calibration, and each viable force-field may exhibit distinct atomistic behavior underpinning the observed values. The approach central to this document pursues a different route based on the following observation: in an ideal setting, a CG model should act identically to an atomistic one viewed at the appropriate resolution. In other words, if a model is created at the resolution of the centers of mass of the amino acids comprising a given protein, then its generated configurations should be indistinguishable from those obtained by calculating the statistics of the corresponding centers of mass from atomistic information. This configurational

information is not directly available from experimental measurements— it is available, however, if an atomistic MD simulation is performed. The resulting atomistic configurations are then mapped to the CG resolution and used as a high dimensional parameterization target to create the CG force-field.

The approach of emulating mapped atomistic data has motivated a number of techniques (a non exhaustive list of methods includes references [38–68]) which we will refer to as bottom-up methods. These methods are designed to produce effective CG models from high dimensional configurations produced by atomistic MD, similar to the post-processing methods discussed previously. Under certain assumptions, these approaches produce models whose statistics are indistinguishable from those of the mapped atomistic system. The task of matching the full mapped atomistic distribution is inherently a high dimensional task: while the CG resolution is lower in dimensionality than the atomistic system, it easily reaches more than 100 dimensions. We note, however, that some of these methods can additionally be described using lower dimensional correlations. For example, references [47, 69] showed that certain high dimensional fits ideally produce the same results as those matching experimental measurements[39–41], making some of these methods similarly valid when considering experimental reference data.

At first, bottom-up methods seem unhelpful: in order to perform a efficient CG simulation, one must first perform an expensive atomistic simulation. How do these methods then enable the study of new systems? First, the CG models themselves, when applied in the post-processing context above, can extract information from the atomistic model and communicate it through parameter introspection, similar to our stove model. For example, changes in the CG force-field parameters upon alterations in atomistic physical conditions can provide insight into molecular organization[70]. Second, and more importantly, bottom-up CG models are generally calibrated on a small system and then applied to a larger system using chemical intuition. For example, bottom up coarse-graining can be used to define harmonic bonds which describe the tertiary structure of a single protein in solution. This bonded network can be used to model hundreds of these proteins in tandem to understand their collective aggregation behavior on a membrane[28, 71].

Unfortunately, CG models often do not reproduce results from the corresponding atomistic simulations. In the case of bottom-up methods, this may initially seem surprising given their parameterization; however, realistic use of CG models differs from the conditions implying their accuracy. First, while there typically exists a perfect CG force-field that reproduces the atomistic statistics at the CG resolution, the force-fields used by CG models are often constrained to low dimensional interactions: a common example is to limit the force-field to contributions similar to those found in classical atomistic force-fields, such as pairwise nonbonded and bonded interactions. This constraint prevents the CG model from capturing the observed statistics, but is useful to improve the interpetability and computational efficiency of the resulting model. Furthermore, even if the atomistic force-field used to parameterize a CG model does not change based on physical conditions (e.g., volume or ambient temperature), the optimal effective potential characterizing the CG model likely does. This variation causes difficulties when measuring thermodynamic variables[72, 73]. Together, these sources of error are referred to as issues of representability[3]. Second, as mentioned previously, the utility of CG models stems from applying them in physical settings different than those underpinning their parameterization; accuracy in these modified conditions is termed transferability. The dependence of CG potentials on physical state is fundamentally tied to issues of transferability: state dependence must be taken into account in order to adapt said potential to new thermodynamic settings.

This approximate nature of CG models makes it ambiguous which CG force-field is optimal with respect to an atomistic system. Similar to the model of our stove, if it were possible to perfectly capture the analyzed data no such ambiguity would exist. However, the limited selection of CG force-fields forces us to only consider imperfect models, and subjective preferences for different types of errors dictate which of these model is an optimal approximation. Unfortunately, unlike our one dimensional stove model, CG models are high dimensional, and the nature of this error is difficult to visualize. The work presented considers a single type of error: the mismatch in the probability distribution given by the CG force-field and that implied by the atomistic simulation at a

[3]These two issues are ameliorated via different strategies, but are referred to by the same term for historical reasons[72].

single state point. While MD iteratively simulates how every particle in a system moves in time, the system as a whole will generally occupy various states, or configurations, with a certain frequency if the simulation is long enough. This property is assumed to hold for both the atomistic and CG models considered, although these probability distributions are present at different resolutions. After adjusting for the difference in resolution, both simulations implicitly provide a probability density at the resolution of the CG model that describes the frequency of visiting a particular CG system configuration. This probabilistic interpretation is standard in MD analysis, but is mentioned here explicitly as it forms the basis for the connections made to machine learning, a subfield of artificial intelligence which designs algorithms which learn from data. A central approach in the presented work is the use of classification, a technique form machine learning, to discern and summarize this error.

1.2 MACHINE LEARNING

Machine learning (ML) has progressed significantly in the last 60 years[74–76]. Signatures of early research into decision trees and neural networks still can still be seen in contemporary algorithms which can extract accurate models form large amounts of data. These algorithms have been applied in a number of domains, ranging from recidivism prediction[77] to playing Go[78]. ML methods are well suited to high dimensional tasks; the high dimensional nature of MD, along with the large amount of data produced, has therefore naturally attracted the use of various methods from ML[13, 14, 79]. The perhaps most salient example is the creation of force-fields which are more expressive then those traditionally used: force-fields are mathematically specified as a function from the system configuration to a scalar energy value, naturally providing a high dimensional setting for regression methods such as neural networks and Gaussian processes. Another prominent application is dimensional reduction: reduced descriptions provide more information on emergent behavior than the simultaneous description of every particle in a system. These techniques have proven to be useful at both the atomistic and CG resolution, in many cases progressing from proof-of-concept approaches to established computational tools.

The work presented here makes heavy use of a particular task studied by ML: classification. Classification is task of predicting the class, or label, associated with a data point[75]. For example, one might want to predict the particular number present in a given picture of a handwritten digit. Classification algorithms are trained to do so by studying an already labeled auxiliary set of samples: in this example, said data would be a set of pictures that have already had the correct number associated with each picture. In the present work, classification is used to define the difference between two probability distributions[80, 81]. The approach of using classification to quantify the difference between probability distributions was popularized by generative adversarial networks (GANs)[82], a type of generative model, and roughly proceeds as follows:

1. Obtain n samples from distribution A
2. Obtain n samples from distribution B
3. Label each sample with which distribution generated it (A or B)
4. Train a classifier to predict the label of each sample

In other words, a classifier is trained to determine which distribution "owns" a particular sample. The classifier provides a guess for each sample— we then average the accuracy of the classifier trying to predict the source of each sample over all the samples to get a single value characterizing the two distributions. If the classifier is 100% accurate, the distributions are very different; if the classifier is 50% accurate, the distributions are the same[4]. The term "adversary" refers to the task of classification: the generator in a GAN can be viewed as fighting against an "adversary" which performs said classification to quantify the quality of the generator. If the generator can fool the adversary— that is, if classification cannot differentiate the samples produced by the generator and those used as reference— the generator has defeated the adversary.

Generative models, such as GANs, produce samples which mimic a set of reference samples[83, 84]. For example, if shown a set of pictures of the handwritten digit five, a trained GAN produces

[4] 50% (and not 0%) corresponds to identical distributions as in the worst case scenario, we can simply guess A for all samples, which will be correct for 50% of the samples.

new pictures of fives; these examples would be similar, but not identical, to the examples used for training. Generative models are deeply tied to the work presented: MD algorithms similarly produce samples of the configurations typical to a molecular system. Generative models in ML have been used in a variety of areas[85] ranging from generating photos of imaginary celebrities[86], increasing the resolution of a given photo[87], generating captions which match a given picture[88], to the generation of molecular configurations[89–91]. Despite the similarities between MD and GANs, however, the methods used to calibrate each of these models differ. Focusing on bottom-up parameterization strategies, MD force-fields are often trained to reproduce configuration-wise energies and forces from more expensive reference simulations. This involves directly evaluating the force-field characterizing the model, a task which is not possible for the architecture used by GANs. In contrast, GANs traditionally use the classification mechanism described previously as a method to quantify how well they mimic the reference samples. The difference in parameterization strategies, along with the similarity in the end goal of sample generation, suggests a natural question that is discussed in the presented work: can these two techniques be formally linked?

1.3 MOLECULAR DYNAMICS AND ADVERSARIES

This setting is where this work begins. High fidelity MD models have the possibility of enabling many scientific discoveries. Unlike their atomistic counterparts, the approximate nature of typical CG MD models makes it difficult to treat them as though they perfectly describe reality, and their high dimensional behavior can be difficult to analyze in its entirety. Furthermore, their approximate nature implies that new parameterization strategies, such as those used by GANs, may create models with favorable approximations. We here use classification as basis for creating methods which begin to address these difficulties.

As discussed previously, performing classification between samples generated by two different distribution characterizes the differences between these distributions. For example, if one of these distributions is the output of our CG force-field and the other is the reference data from an atomistic simulation, classification analyzes the configurational errors made by the CG force-field.

Algorithms for performing classification are already extensively optimized; the task we consider here is how to mathematically use the results of classification to understand and optimize CG models. In this context, the main output of classification is function which maps each CG molecular configuration to a number characterizing whether the CG model over- or under- stabilizes that point in phase space, a function we refer to (after transformation) as ΔW. When viewed as a measurement defined on each molecular configuration, ΔW is a collective variable whose reported overlap of our two distributions matches that given in original high dimensional configurational phase space, providing a low dimensional visualization of the high dimensional error present. We continue further by analyzing the decision the classification method used to construct ΔW using methods from explainable artificial intelligence, allowing us to understand the variety of errors made by the CG model. Finally, we show that ΔW can be used to adversarially perform bottom-up CG model parameterization, and in doing so produce variational training formulations generalizing relative entropy minimization[47], a common method for CG model parameterization.

The remainder of this document is structured as follows. Chapter 2 describes the mathematical notation we will use in the later sections. Chapter 3 describes and demonstrates classification's role when comparing free energy surfaces by examining the errors in CG models. Chapter 4 discusses using classification-based adversaries to calibrate CG force-fields, and chapter 5 contains concluding remarks.

CHAPTER 2

NOTATION AND SETTING

I have never been a fan of algebra. If there is been one theme throughout the work presented here, it is that all of the requisite conceptual progress has already been written down but resides in a different field. Algebra shines in its concision but creates barriers to communication, whether it be from the manipulations used or the implied frame of reference the concision imposes. However, I do not see a more suitable language to describe the content of this work. This chapter provides much of the terminology and external results central to chapters 3 and 4; an additional theorem pertinent to chapter 4 is provided in appendix C.

Section 2.1 describes the coarse-grained (and to some extent, atomistic) molecular dynamics (MD) models which are discussed in this work. Unless otherwise specified, all fine-grained and coarse-grained (CG) models discussed are MD models. There are a large number of analyses that can be performed on results from these models; however, for our purposes they are simply ways to draw samples from a high dimensional probability distribution. Section 2.2 provides a mathematical description of classification (along with extensions to regression), which is the basis for analysis performed throughout this work.

2.1 COARSE-GRAINING

MD simulations, when performed in the canonical ensemble, iteratively produce samples which ideally follow the Boltzmann distribution with respect to their Hamiltonian. As we will see later, the portion of the Hamiltonian which typically requires approximation is the force-field. Throughout

this work we only characterize the fine-grained (FG, e.g. atomistic) and CG models by these corresponding probability distributions. If a CG model is consistent with a FG model in this regard, configurations at the CG resolution will be visited with equal frequency by both the FG and CG models.[1] The resolution of a CG model is determined by the mapping operator, a function that transforms a configuration at the FG resolution to one at the CG resolution. In this section we describe these criteria and functions in detail.

We consider a FG probability density $p_{\text{ref}}^{\text{FG}}$ and a mapping operator $\mathcal{M}$ that maps a FG configuration to a CG configuration. The FG simulation is constructed such that it produces samples from the Boltzmann distribution with respect to a FG Hamiltonian giving the following probability density:

$$p_{\text{ref}}^{\text{FG}}(\mathbf{r}^{3n},\mathbf{p}^{3n}) := {Z_{\text{ref},\mathbf{r}}^{\text{FG}}}^{-1}{Z_{\text{ref},\mathbf{p}}^{\text{FG}}}^{-1}\exp\left[-\beta\left(\sum_{i=1}^{n}\frac{\mathbf{p}_i^2}{2m_i}+U_{\text{ref}}^{\text{FG}}(\mathbf{r}^{3n})\right)\right] = p_{\text{ref},\mathbf{p}}^{\text{FG}}(\mathbf{p}^{3n})p_{\text{ref},\mathbf{r}}^{\text{FG}}(\mathbf{r}^{3n}) \quad (2.1)$$

where β is $\frac{1}{k_b T}$ with the temperature T set by the simulation protocol, m_i are the FG masses, $\mathbf{r}^{3n}$ and $\mathbf{p}^{3n}$ are the FG positions and momenta variables, and our partition functions are defined as expected[2] such that

$$Z_{\text{ref},\mathbf{r}}^{\text{FG}} = \int_{\mathcal{X}_{\mathbf{r}}^{\text{FG}}}\exp\left[-\beta U_{\text{ref}}^{\text{FG}}(\mathbf{r}^{3n})\right]\mathrm{d}\mathbf{r}^{3n} \quad (2.2)$$

$$Z_{\text{ref},\mathbf{p}}^{\text{FG}} = \int_{\mathcal{X}_{\mathbf{p}}^{\text{FG}}}\exp\left[-\beta\sum_{i=1}^{n}\frac{\mathbf{p}_i^2}{2m_i}\right]\mathrm{d}\mathbf{p}^{3n} \quad (2.3)$$

where the integrals are taken over the full domains of the position and momentum variables (denoted via $\mathcal{X}_{\mathbf{r}}^{\text{FG}}$ and $\mathcal{X}_{\mathbf{p}}^{\text{FG}}$). Note that $\left(\sum_{i=1}^{n}\frac{\mathbf{p}_i^2}{2m_i}+U_{\text{ref}}^{\text{FG}}(\mathbf{r}^{3n})\right)$ collectively corresponds to the FG Hamiltonian; the first term refers to the distribution over the momentum variables ($\mathbf{p}^{3n}$) while the second term corresponds to the distribution over the configurational ($\mathbf{r}^{3n}$) variables. The second

[1] This property is sometimes termed thermodynamic consistency; however, it does not imply that thermodynamic measurements on the two ensembles are equivalent.

[2] Throughout this work we omit proportionality constants related to, for example, indistinguishability and unit systems (including the factors of Planck's constant often introduced through quantum mechanical limits); reintroduction of these constants is straightforward. The term partition function is used to generally refer to normalization constants.

term is also referred to as the force-field. The application of the $\mathcal{M}$ produces CG configurations that follow an implied probability distribution. $\mathcal{M}$ is constrained such that it is linear and can be decomposed into momentum and position components, i.e., $\mathcal{M}(\mathbf{r}^{3n}, \mathbf{p}^{3n}) = [\mathcal{M}_{\mathbf{r}}(\mathbf{r}^{3n}); \mathcal{M}_{\mathbf{p}}(\mathbf{p}^{3n})]$,[3] implying a factorizable probability density $p_{\text{ref}}(\mathbf{R}^{3N}, \mathbf{P}^{3N}) := p_{\text{ref},\mathbf{R}}(\mathbf{R}^{3N}) p_{\text{ref},\mathbf{P}}(\mathbf{P}^{3N})$ over the CG variables defined as

$$p_{\text{ref},\mathbf{R}}(\mathbf{R}^{3N}) := \int_{\mathcal{X}_{\mathbf{r}}^{\text{FG}}} p_{\text{ref},\mathbf{r}}^{\text{FG}}(\mathbf{r}^{3n}) \delta(\mathcal{M}_{\mathbf{r}}(\mathbf{r}^{3n}) - \mathbf{R}^{3N}) \mathrm{d}\mathbf{r}^{3n} \tag{2.4}$$

$$p_{\text{ref},\mathbf{P}}(\mathbf{P}^{3N}) := \int_{\mathcal{X}_{\mathbf{p}}^{\text{FG}}} p_{\text{ref},\mathbf{p}}^{\text{FG}}(\mathbf{p}^{3n}) \delta(\mathcal{M}_{\mathbf{p}}(\mathbf{p}^{3n}) - \mathbf{P}^{3N}) \mathrm{d}\mathbf{p}^{3n}, \tag{2.5}$$

where δ corresponds to the multidimensional Dirac delta function. Bottom-up CG models aim to directly produce samples from the distribution described by p_{ref}[44, 45]. Ideally, this is achieved by defining a model CG Hamiltonian $\left(\sum_{i=1}^{N} \frac{\mathbf{P}_i^2}{2M_i} + U_{\text{mod}}(\mathbf{R}^{3N})\right)$ such that the corresponding Boltzmann statistics

$$\begin{aligned} p_{\text{mod}}(\mathbf{R}^{3N}, \mathbf{P}^{3N}) &:= {Z_{\text{mod},\mathbf{R}}}^{-1} {Z_{\text{mod},\mathbf{P}}}^{-1} \exp\left[-\beta\left(\sum_{i=1}^{N} \frac{\mathbf{P}_i^2}{2M_i} + U_{\text{mod}}(\mathbf{R}^{3N})\right)\right] \\ &= p_{\text{mod},\mathbf{R}}(\mathbf{R}^{3N}) p_{\text{mod},\mathbf{P}}(\mathbf{P}^{3N}) \end{aligned} \tag{2.6}$$

are ideally identical to the mapped FG statistics, criteria expressed with the following CG consistency equations[45]

$$p_{\text{ref},\mathbf{R}}(\mathbf{R}^{3N}) = p_{\text{mod},\mathbf{R}}(\mathbf{R}^{3N}) \tag{2.7}$$

$$p_{\text{ref},\mathbf{P}}(\mathbf{P}^{3N}) = p_{\text{mod},\mathbf{P}}(\mathbf{P}^{3N}). \tag{2.8}$$

These equations characterize the ideal CG model with respect to its probability distributions. The ideal CG force-field which satisfies configurational consistency is often referred to as the many-

[3] $\mathcal{M}$ is also typically[45] additionally constrained such that the resulting coordinates are linearly independent and unambiguously associate at most one CG site to each atom. These constraints are mostly unimportant to the work at hand except for when momentum consistency is considered, but some care must be taken so that the corresponding densities exist.

body potential of mean force (manybody-PMF). Critically, while the manybody-PMF shows up in theoretical formulations, it is not possible to evaluate it on molecular configurations due to the high dimensional integral in its definition.

It is important to note that while we have used the symbol U when describing the Boltzmann distributions of both the CG force-field and the manybody-PMF, these distributions correspond to free energy surfaces when $\mathcal{M}$ is not invertible (assuming the corresponding probability density exist); this notation is used for similarity with the existing systematic CG literature (e.g., Dama *et al.* [52]). This is in contrast to the typical thermodynamic definition of U as the average energy. Nowhere in this document do we estimate the difference in the conditional average energies of two free energy surfaces. We note that setting $\mathcal{M}$ to be invertible would allow one to match energies using the proposed algorithms for atomistic systems, but that this is not usually necessary as said difference can be calculated directly.

The above characterization is sufficient for chapter 3; however, the generative adversarial network related connections present in 4 require us to describe the optimization of various force-fields, a procedure we describe here. Momentum and configurational consistency are generally treated separately, with momentum consistency exactly satisfied through direct definition of CG masses M_i and configurational consistency approximated through a variational statement (as the corresponding integral is not generally tractable)[45]. Momentum consistency is further discussed in section 4.3.5. This variational statement central to configurational consistency transforms the difficult task of high dimensional probability density estimation into an optimization problem; the quantity optimized can be considered a characterization of the aggregate error of a candidate force-field. Said variational statement utilizes a variety of information depending on the method considered. Generally, knowledge of $U_{\text{ref}}^{\text{FG}}$, U_{mod}, and $\mathcal{M}$ are used. In many cases the corresponding variational principle can be considered in the following form

$$\theta^{\dagger} := \underset{\theta}{\operatorname{argmin}}\, \mathcal{F}\left[p_{\text{mod},\mathbf{R},\theta}; p_{\text{ref},\mathbf{r}}^{\text{FG}}, \mathcal{M}\right] \tag{2.9}$$

Figure 2.1: Various digits from the MNIST dataset[92, 93].

where θ denotes the finite parameterization of our CG potential, $\theta^{\dagger}$ parameterizes our ideal model, and $\mathcal{F}$ is a function characterizing the quality of our model. Often[38, 43–48, 55], the exact form of the variational statement contains intractable integrals which are approximated via empirical averages from atomistic and CG trajectories.

Importantly, while the models discussed in chapter 4 fit into this framework, they differ in one important respect to many previous CG parameterization strategies: they introduce a variational definition of $\mathcal{F}$ itself, resulting in two nested variational statements in the numerical optimization procedure.

2.2 CLASSIFICATION

Classification is central to many of the analyses presented in this work, and entails training an algorithm to predict which class is associated with a particular sample. Regression and classification are similar: whereas regression often aims to predict a real number, classification predicts a discrete label. A common example is predicting the content present in various images in the Modified National Institute of Standards and Technology (MNIST) data set[92], which is composed of small black and white pictures of handwritten digits (Fig 2.1). Each of these pictures corresponds to a single digit. In the classification setting, half of the pictures in the dataset could first be associated

with the correct label (e.g., 1) as obtained from human inspection. This labeled data set would then be used to train a classification algorithm such as a perceptron. Validation of the accuracy of the perceptron would be performed by applying the algorithm to the second half of the MNIST data set: the predicted labels would be compared to external labels again obtained from human inspection.

The example of classifying digits in the MNIST dataset had ten classes (one for each possible digit). The classification problems central to the work presented in chapters 3 and 4 have two classes. This is analogous to only predicting the digit present in the MNIST dataset if we limit ourselves to pictures solely of 0 or 1. Mathematically, we can describe how a possible classifier performs in the two class case via the following expression[81, 94]:

$$\sum_{y\in\{0,1\}} \int_{\mathcal{X}} l(f(x),y)P(x,y)\mathrm{d}x \tag{2.10}$$

where y denotes our label (0 or 1), x denotes the values of the pixels of the image, and P[4] corresponds to the joint probability of a particular image content with a possible label. f corresponds to our hypothesis; in the previous example, it would correspond to a perceptron, linking images (x) to digits (y). l corresponds to our loss function, which characterizes how we penalize an incorrect prediction. Formally, classification has a loss as follows

$$l^{0-1}(y',y) := \begin{cases} 0 & \text{if } y' = y \\ 1 & \text{if } y' \neq y. \end{cases} \tag{2.11}$$

Unfortunately, this loss is often difficult to optimize. In place of the l^{0-1} loss many classifiers

[4]This section abuses notation slightly for brevity. Note that $P(x|0)$ and $P(x|1)$ are formally distinct densities, while $P(0)$ and $P(1)$ are distinct evaluations of a mass function. P is most easily fully defined conditioning on the label. Readers are referred to references [76, 81, 94, 95] for a more rigorous introduction. M and η are later defined to reduce this ambiguity.

optimize smoother surrogate losses such as the log loss:

$$l^{\log}(y',y) := \begin{cases} -\log(1-y') & \text{if } y = 0 \\ -\log(y') & \text{if } y = 1. \end{cases} \tag{2.12}$$

Importantly, when using the l^{0-1} loss our prediction y' is in the set $\{0,1\}$, but when using surrogate losses, such as $l^{\log}$, y' is often in the set $[0,1]$. Additionally, when the surrogate loss function is a strictly proper loss function (a condition satisfied by $l^{\log}$ above), the intermediate output of the classifier encodes the probability of each class; typical classification behavior can be regained by selecting the most probable label. The intermediate probabilistic output, however, results in class probability estimation, providing information central the work presented. We will informally include class probability estimation under the umbrella of classification. A more complete characterization of proper losses is given in chapter 4 and references [81, 94, 95].

In general, a joint probability function (density or mass) $p(x,y)$ can be factorized as $p(x|y)p(y)$[96]. This can be used to characterize Eq. (2.10) in two ways. First, one can condition on x, the image content, resulting in

$$\int_{\mathcal{X}} \left[\sum_{y \in \{0,1\}} l(f(x),y) P(y|x) \right] P(x) \mathrm{d}x. \tag{2.13}$$

This factorization focuses on the view that we are looking for a hypothesis f that provides a guess, even when certain images may have ambiguous content. Alternatively, one can condition on the y, the label, giving

$$\sum_{y \in \{0,1\}} \left[\int_{\mathcal{X}} l(f(x),y) P(x|y) \mathrm{d}x \right] P(y). \tag{2.14}$$

This viewpoint focuses on characterizing the images specific to either label. While Eq. (2.13) is useful for understanding the behavior of proper losses, Eq. (2.14) allows us to formulate the comparisons used in this work. For the remainder of this document we assume that equal proportions

of each label are present, i.e. $P(Y=0)=P(Y=1)=\frac{1}{2}$. This implies

$$M(x) := P(x) = \frac{1}{2}\left[P(x|0)+P(x|1)\right] \tag{2.15}$$

and

$$\eta := P(1|x) = \frac{P(x|1)}{P(x|1)+P(x|0)} \tag{2.16}$$

where we have defined M and η. η is notable in that it is, among all functions mapping x to $y' \in [0,1]$, the global minimizer of Eq. (2.17)[81, 94, 95] assuming a strictly proper loss; i.e.

$$\eta = \underset{f\in[0,1]^{X_X}}{\operatorname{argmin}} \left[\sum_{y\in\{0,1\}} \int_X l(f(x),y)P(x,y)\mathrm{d}x\right] \tag{2.17}$$

$M(x)$ is defined simply to disambiguate the many uses of P. The generality of these expressions is primarily rooted in the exchange of the minimum and integral operations. Note that for some losses use of inf in place of min is required. These restrictions do not play into the results of this work, and we do not mention them again; we refer the reader to references [81, 94, 95] for more details. Eq. (2.13), when combined with our assumption of equal labels and when written using the brackets typical to statistical mechanics, results in

$$\frac{1}{2}\left\langle l(f(x),1)\right\rangle_{P(x|1)} + \frac{1}{2}\left\langle l(f(x),0)\right\rangle_{P(x|0)} \tag{2.18}$$

where $\langle\cdot\rangle$ denotes the integral of the function over the specified probability distribution (in other words, the expectation value). These expressions are used explicitly in chapter 4.

2.2.1 REGRESSION

Eq. (2.13) can easily be extended to regression:

$$\int_X \int_{\mathcal{Y}} l(f(x),y)P(y|x)M(x)\mathrm{d}x\mathrm{d}y \tag{2.19}$$

Here, however, l is often limited to the square loss (the square loss is similarly valid for two-class classification[81]):

$$l^{\mathrm{sq}}(y',y) := (y'-y)^2 \tag{2.20}$$

The global minimizer for the square loss similarly is known[96] to be the conditional mean:

$$\operatorname*{argmin}_f \int_{\mathcal{X}}\int_{\mathcal{Y}} l^{\mathrm{sq}}(f(x),y)P(y|x)M(x)\mathrm{d}x\mathrm{d}y = \int_{\mathcal{Y}} yP(y|x)\mathrm{d}y \tag{2.21}$$

An important secondary property is given by the law of conditional expectation, which for continuous distributions can be stated as

$$\int_{\mathcal{X}}\int_{\mathcal{Y}} f(x,y)P(x,y)\mathrm{d}x\mathrm{d}y = \int_{\mathcal{X}}\int_{\mathcal{Y}}\left[\int_{\mathcal{Y}} f(x,y)P(y|x)\mathrm{d}y\right]P(x,y)\mathrm{d}y\mathrm{d}x \tag{2.22}$$

We note that Eq. (2.22), when combined with Eq. (2.21), implies recent results related to defining novel CG observables[72, 97], but does so without distributional assumptions.

CHAPTER 3

QUANTIFYING COARSE-GRAINED MODEL ERROR

3.1 MOTIVATION

As mentioned in the introduction, coarse-grained (CG) models are high dimensional. When we are pressed to consider their configurational behavior we often either resort to projecting their results to a lower dimension (e.g., radial or angle distribution functions or coordinates created via dimensional reduction) or using molecular visualization. As a result, when creating new CG models, the choice of the bottom-up algorithm selected for designing the force-field for complex systems (among other design choices such as those discussed in sections 3.5.1 and 3.5.2) has often relied upon said methods of inspection[98]. However, these have always seemed to me to be crude approaches. The behavior of CG models is complex; what other manybody errors are present? Alternatively, if multiple CGing algorithms give similar force-fields, how strongly does this imply that the true manybody potential of mean force (manybody-PMF) is well captured? Do different choices of mapping operators create simpler manybody-PMFs? These questions, as of now, seem to be rather open. As bottom-up CG models increase the computational cost of parameterization with the aim of being quantitatively accurate, it seems jarring that the absolute manybody error of various models is a difficult topic to discuss.

If the manybody-PMF were generally evaluable, the pointwise difference between U_{mod} and U_{ref} could be mined for insight. However, it is not. The work presented here attempts to provide an initial numerical approach for capturing this pointwise error using classification. The initial insight is that the probabilistic description of the ideal η is easily transformable into a per configuration measure

of error. This, alone, is not particularly groundbreaking: the resulting term is proportional to the log odds ratio, a quantity well known to statisticians. Additionally, the pointwise error given by η is still high dimensional. The task, however, of understanding η can be reformulated as a question of explainable artificial intelligence. If we can probe the algorithm learned when approximating η, this provides a window into the error characterizing the CG model. This is the crux of this chapter: we propose that explainable machine learning can convey high-dimensional error to scientists and use Shapley additive explanations do so in CG protein models. We additionally provide initial work using the same method to uncover the differences between various atomistic ensembles.

3.2 INTRODUCTION

Atomistic molecular dynamics (MD) has provided scientific insight into many problems[99–103]. As discussed previously, however, atomistic MD is still limited in terms of the time- and space-scales it can access. These limitations have motivated the development of CG models that simulate the original system at a minimal resolution, aiming to reduce computational cost while maintaining quantitative accuracy[11, 13, 14, 28, 30, 33–37]. As discussed in chapter 2, CG MD models describe molecular behavior using MD at a coarser resolution than their atomistic counterparts; the behavior of these simulations is primarily controlled via an effective Hamiltonian. For example, a CG model may simulate a solvated protein by propagating only the center of mass of each amino acid; the equilibrium distribution could then be controlled by a Hamiltonian defined at the resolution of these centers of mass. The behavior of these models can be divided into their dynamic and thermodynamic[1] properties. While accurately reproducing the dynamics of a reference atomistic system is an area of current interest[104], as noted in chapter 2, this work focuses on thermodynamic issues, and more specifically, the configurational distribution produced by a CG model. An extended description of these models is given in chapters 2 and 4; we provide a limited discussion here as background for the content of this chapter.

[1]Thermodynamic here refers to long-time behavior related to the equilibrium distribution of the model. It includes both issues related to estimating thermodynamic quantities, such as pressure, and averages of functions of individual microstates.

There are many ways to create the effective Hamiltonian characterizing a CG model[11, 13, 14, 28, 30, 33–37]; these various approaches often result in different effective force-fields. Top-down methods aim to reproduce observables that are coarser than the effective Hamiltonian, such as partition coefficients or interfacial tension. The coarse resolution of these observables makes it possible to obtain reference data from either experiment or simulation. In contrast, bottom-up methods tend to require samples from a reference atomistic simulation, which are mapped to the resolution of the CG Hamiltonian and used as a target for parameterization. The variety of possible parameterization techniques makes it valuable to understand how a proposed force-field approximates a reference atomistic simulation at the resolution of the CG model (i.e., that of the effective Hamiltonian), whether it be a reference simulation used to parameterize the model or one created solely for external validation.

However, while a CG model is intrinsically coarser than the atomistic model it represents, it is still high dimensional. For example, the CG molecules in this document are relatively small but easily reach 36 dimensions, which is well beyond what generic data visualizations (e.g., scatter plots or histograms) can communicate to humans. These models are often still amenable to visualization as groups of particles in 3D space[105] as the systems preferentially occupy a small portion of the possible phase space (for example, a protein does not dissociate into its constituent atoms— its behavior is strongly constrained by its primary and higher order structure). However, while visual inspection can detect some errors, it can be difficult to translate into quantitative evaluations of the model. Established dimensional reduction techniques[13, 14, 79] can be used to summarize the behavior present in the model and reference data, but are not typically designed to find differences between them. In contrast, the original task of designing said force-fields does involve quantitative comparison of candidate force-fields to reference data. In the case of top-down force-fields, optimization involves statistics that are readily intepretable by computational scientists; however, these observables are coarser than the resolution of the effective Hamiltonian. In the case of bottom-up force-fields, while the considered resolution is ideal, said optimization often utilizes specific (and often opaque) computational algorithms to optimize the Hamiltonian such that its high dimensional

configurational statistics approximate those of a reference model. As a result, it can be difficult to understand how the behavior produced by a CG force-field deviates from the ideal high dimensional behavior of the atomistic system, even when such reference data is available.

This chapter will provide a way to compare two samples from differing high dimensional free energy surfaces. The analysis presented here does not specify a particular process be used to generate these samples. However, the models we will study have been created using bottom-up coarse-graining techniques and we will borrow terminology from the bottom-up coarse-graining literature to express our ideas: for example, we will refer to the function which maps each atomistic configuration to its CG counterpart as the CG mapping operator, and we will refer to the ideal effective force-field perfectly reproducing the mapped atomistic statistics as the manybody potential of mean force (manybody-PMF). Furthermore, as the techniques we describe provide a way to compare two samples from differing high dimensional free energy surfaces, these approaches are especially pertinent to bottom-up parameterization strategies as a mapped atomistic reference sample is readily available.

As noted previously, the high dimensional nature of the data produced by CG models makes it difficult to directly visualize their full behavior. When parameterizing atomistic models using quantum mechanical data, computational scientists can often additionally compare the energies or forces produced by the reference method to those produced by the proposed atomistic force-field, and use this configuration-wise error (or atom-wise decompositions of this error) to study problematic areas of phase space (e.g. Bartók *et al.* [106]). This is difficult to do with CG models as the pointwise evaluation of the manybody-PMF is often not available and only a noisy estimate may be available for the forces of the manybody-PMF[2]. An attractive approach is to train multiple CG force-fields, each with increasingly complex manybody interactions: treating a higher order potential as if it were the true manybody-PMF allows one to estimate the conditional free energy

[2]It is possible to use constrained fine-grained simulation to estimate the derivative of the manybody-PMF[107]. The feasibility of this approach depends on the system at hand and the level of coarse-graining, but the results can analyzed using methods analogous to those described in this chapter.

and force differences at each configuration. In this chapter we show that a classifier can be used in lieu of training a more complex CG force-field to obtain the same estimate of force-field quality. While this alone allows the computational scientist to isolate problematic configurations based on force-field error, the larger contribution of this work is the realization that the classifier based approach, when combined with methods from explainable ML/AI (collectively referred to as XAI in this chapter), allows one to analyze the force-field errors in a novel way.

XAI is a subfield of AI under active development (for an overview of the corresponding definitions, see references [108–113]). This issue of algorithmic transparency is not new (see, e.g., references [114, 115]); however, as automated decisions become more common in everyday life, an increasing amount of scrutiny has been placed on providing justifications for the output of automated systems[77, 116]. This is required for a variety of reasons, whether it be regulatory compliance, ethical analysis, debugging, or further comprehension of the data used to train the algorithm. For the purposes of this chapter, we divide the algorithms in this field into two categories: transparent (or intepretable) models and post hoc explanations. Transparent models are algorithms that, once trained, can be intrinsically understood by a particular audience; examples include shallow decision trees or rule lists. Algorithms of this type are generally simpler than more opaque algorithms and do not include approaches such as deep neural networks. Post hoc explanations, on the other hand, are methods that digest and summarize information from an already optimized external algorithm (such as feature attribution methods). While transparent models are intrinsically interpretable, explainable models are those which have additional post hoc explanation methods to provide reduced representations of the knowledge present in the trained model.

The main contribution of this work is the following realization: extracting the knowledge present in a classifier trained to estimate force-field error provides a description of the errors present in the original model. The nature of the information provided depends on the particular methods from XAI we adopt. To illustrate this concept we take a particular modern explanation technique, Shapley additive explanations (SHAP values), and demonstrate how they can isolate which physical portions of two CG proteins (dodecaalanine and actin) are problematic for specific CG force-fields.

The goal of this chapter is not to model these two proteins accurately; instead, it is to show that non-ideal force-fields can be detected and understood. These insights are then aggregated using typical dimensional reduction techniques to produce collective variables (CVs) which optimally characterize the types of error exhibited by a candidate force-field parameterization. This approach is shown to be useful when considering the behavior of CG force-fields, and provides a conceptual basis for future bottom-up error analysis through classification. We then provide results comparing the free energy surfaces implied by various mapped atomistic ensembles: this initial work provides a way to understand both the impact of molecular changes at the subdomain resolution and provides a way to characterize heterogeneity in molecular assemblies.

3.3 THEORY

The techniques presented in this chapter compare two high dimensional free energy surfaces, which we refer to as $W_{\mathrm{PMF}}(x)$ and $W_{\mathrm{FF}}(x)$, where x represents a sample on the free energy surface. While W_{PMF} and W_{FF} may refer to the configurational portions of U_{ref} and U_{mod}, they can also refer to implied free energy surfaces at a coarser resolution. However, in the analysis that follows we will often assume said equivalence. Via the canonical ensemble these free energy surfaces define probability densities as

$$p_{\mathrm{PMF}}(x) = Z_{\mathrm{PMF}}^{-1} \mathrm{e}^{-\beta W_{\mathrm{PMF}}(x)}$$

and

$$p_{\mathrm{FF}}(x) = Z_{\mathrm{FF}}^{-1} \mathrm{e}^{-\beta W_{\mathrm{FF}}(x)},$$

with Z_{PMF} and Z_{FF} defined as the integral of $\mathrm{e}^{-\beta W_{\mathrm{PMF}}}$ and $\mathrm{e}^{-\beta W_{\mathrm{FF}}}$ over all possible x.

As described in section 2.2, classification is the machine learning task of predicting the most probable class, or label, associated with a given data point[75]. For example, one might want to predict the particular number present in a given picture of a handwritten digit. Algorithms in supervised classification are trained to complete this labeling task by studying an already labeled data set: in the previous example, said data set would be a set of pictures that have had the correct

number already associated with each picture. In certain learning contexts various samples may not have a clear correct class. For example, certain hand written digits may be so messy that only the original writer knows the digit truly intended. In order to naturally adapt to ambiguous samples, classifiers can be designed to output a guess of the probability each possible class[75, 81, 94, 95]. Focusing on the case where we only have two possible classes to predict (for example, if we were only considering pictures of 4 or 5), this probabilistic estimate can be quantified by a single real number for each sample between 0 and 1 (which we refer to as η).

Classification is used in this work by setting the class conditional probabilities to p_{FF} and p_{PMF}. Using the connections from section 2.2, η then takes the following form:

$$\eta(x) = \frac{p_{\mathrm{PMF}}(x)}{p_{\mathrm{PMF}}(x) + p_{\mathrm{FF}}(x)} \tag{3.1}$$

A calibrated classifier can be used to approximate η using only samples from each free energy surface; importantly, this implies that it is not necessary to know the values of W_{FF} or W_{PMF} evaluated on samples in order to estimate η on each sample. This approach is summarized by the following algorithm:

1. Generate N samples from W_{PMF}
2. Generate N samples from W_{FF}
3. Label all samples from W_{PMF} with "A"
4. Label all samples from W_{FF} with "B"
5. Train a calibrated classifier to predict η

η, when combined with β, can be transformed into a offset pointwise difference between W_{FF} and W_{PMF}, which we refer to as ΔW:

$$\Delta W(x) := \mathrm{k_b}T \log \frac{\eta(x)}{1-\eta(x)} = W_{\mathrm{FF}}(x) - W_{\mathrm{PMF}}(x) + \mathrm{k_b}T \log \frac{Z_{\mathrm{FF}}}{Z_{\mathrm{PMF}}}. \tag{3.2}$$

When W_{FF} corresponds to the configurational distribution of a CG simulation and W_{PMF} corresponds to the manybody-PMF, we typically can evaluate W_{FF} on an arbitrary configuration but are unable to evaluate W_{PMF}. Additively combining ΔW and W_{FF} provides an estimate of W_{PMF}, one that would be exact with a perfect classifier and learning procedure. In practice, limited data and imperfect classification algorithms make this estimate only approximate. Using ΔW to form an additive update to W_{FF} has been performed in the past, see Lemke & Peter [59]. This estimation duality, combined with the realization that classification is a variational process with respect to the proposed hypothesis, establish that classification can here be viewed as a variational technique to estimate the manybody-PMF using W_{FF} as a reference. Note that ΔW is defined here such that there is no unknown global offset.

As ΔW precisely characterizes the pointwise difference between two free energy surfaces, evaluating it at a particular configuration quantifies the difference in the conditional free-energies at that point. Areas of high ΔW imply that one ensemble has considerably more population in said area than the other ensemble, while a largely negative ΔW implies the opposite. Equivalently, when used as a structural CV, ΔW describes which configurations occupy areas of high distributional overlap and which are specific to either free energy surface. In the context of W_{FF} approximating W_{PMF}, linking ΔW to other intuitive structural variables can characterize which errors a CG model is exhibiting. For example, if ΔW is negative whenever a particular bond distance is small, this implies that the CG model (W_{FF}) is over stabilizing small bond distances. The advantage relative to direct visualization of configurational statistics from either ensemble is that configurations occupying areas of high distributional overlap may be discarded prior to analysis. This approach, however, is still relatively tedious: it again requires human analysis of the resulting configurations, incurring all of the difficulties discussed in the introduction. An appealing alternative is to understand the algorithmic form of ΔW itself: for example, if it is a linear function, its learned coefficients may offer insight. However, if W_{FF} is composed of low order n-body terms while W_{PMF} contains higher order terms, ΔW will contain high order terms as well, and it may be difficult for intepretable models to provide a good estimate of ΔW. In this chapter we use techniques from XAI to overcome this

difficulty and extract configurational information from a complex estimate of ΔW.

3.3.1 FEATURE ATTRIBUTION

XAI includes of a large variety of methods. This chapter will focus on the use of a single method from this field: SHAP values[117–119]. SHAP values are a feature attribution method[108, 114, 115] with a strong mathematical underpinning. Feature attribution methods, or feature importance measures, provide a quantification of how informative a feature, or particular input variable, is to an algorithm. Some feature attribution methods are global, meaning that stated feature values are related to the aggregate behavior of the classifier over the entire data set. Other feature attribution methods, such as SHAP values, are local: every prediction made by a classifier can be associated with a particular set of SHAP values. When estimating ΔW, this means that a prediction of a large positive or negative ΔW can be analyzed to determine which features lead the classifier to that conclusion.[3]

The first set of classification examples in this chapter quantify the error present in various CG models of proteins. The classification algorithm is trained on the ordered distance matrix derived from each configuration. As a result, applying a feature attribution method to explain ΔW isolates which pairwise distances are most connected to the estimated error, and in doing so clarifies which configurational aspects of the protein are well captured by the CG model. The second set of examples considers the difference between two atomistic ensembles of different systems: here, feature attribution similarly describes how these ensembles differ.

3.3.2 SHAPLEY AND SHAP VALUES

SHAP values are based on Shapley values[108, 120] from cooperative game theory. We first explain Shapley values and how they relate to classification, and then provide a description of SHAP values.

[3]Here, η is not approximated as solely the output of the gradient boosted trees; it is the composition of said output with the distance matrix featurization. As a result, feature attributions information is produced on the resolution of the distance matrix, not the Cartesian coordinates.

3.3.2.1 SHAPLEY VALUES

Shapley values are a method to fairly distribute a reward among a group of individuals. Suppose a group of five scientists decides to create a product to bring to market. These five people have joined together because their individual knowledge, when combined, produces a better product then they could individually. However, suppose one of the five people has knowledge that is vital to the product: if they were not present, the total amount of profit would be greatly diminished. In contrast, the expertise of the remaining four people is largely, but not completely, shared. As a result, losing one of those four people would reduce the possible profit, but would not do so substantially. In this situation, how should the profit be fairly divided among the scientists? Allocation in these circumstances is answered by Shapley values.

The central calculation needed to define Shapley values is the ability to estimate the reward had some of the individuals in the group not been present. Suppose the five people present are referred to as A, B, C, D, and E. We then denote the reward when everyone is present $\mathcal{C}(\{A,B,C,D,E\})$. In order to calculate Shapley values we must be able to calculate, as an example, $\mathcal{C}(\{A,B,D,E\})$: the reward had individual C not been present. Shapley values then consider growing the number of present individuals incrementally, such as the (ordered) sequence $\mathcal{C}(\{B\}), \mathcal{C}(\{B,A\}), \mathcal{C}(\{B,A,D\})$..., and associating with individual D the term $\mathcal{C}(\{B,A,D\}) - \mathcal{C}(\{B,A\})$: the incremental increase that was seen when D was added in this particular sequence. The Shapley value averages over all such sequences. Mathematically, the Shapley value for player i is defined as

$$\phi_i = \frac{1}{n!} \sum_R \mathcal{C}(P_i^R \cup \{i\}) - \mathcal{C}(P_i^R) \tag{3.3}$$

where n is the number of individuals, R iterates over all possible orders (not subsets) of players and P_i^R is the subset of individuals that precedes player i in that particular R. It is important to note here that this sum is over all possible orderings, as where $\mathcal{C}$ only depends on the members present, not the order in which they were added. This calculation must be performed for each individual (or player) for which we wish to calculate a Shapley value.

Shapley values satisfy a number of intuitive mathematical properties[117, 120, 121] and in some cases are the only allocation method that does so. The most important property for the current application is that summing together the Shapley values for all players provides the original reward when the entire group is present.

3.3.2.2 SHAP VALUES

The connection between Shapley values and feature attribution is made by considering every individual prediction made by a classifier a game in which each feature is a player. The output of the game, in analogy with the total profit in the previous subsection, is the numerical prediction of the classifier. However, one important detail is absent when considering feature attribution: what does it mean for a feature to be missing? It is possible in some cases to retrain a model with only a subset of the original features[122]; however, the number of models required quickly becomes infeasible. Instead, Shapley Additive Explanation (SHAP) values train a single model and average said full model's predictions over the missing variables to represent missing features[117]. For example, consider hypothetical classifier $f(w,x,y,z)$ with four input variables. Suppose we wished to calculate the Shapley value of w for configuration (w_0,x_0,y_0,z_0): this would include calculating $f_{wz}(x_0,y_0)$, where w and z are "missing". SHAP values dictate that $f_{wz}(x_0,y_0) := \int f(w,x_0,y_0,z)p(w,z|x_0,y_0)\mathrm{d}w\mathrm{d}z$, where $p(w,z|x_0,y_0)$ is the distribution of w and z conditional[4] on $x = x_0$ and $y = y_0$. With this definition of "missing values" Shapley values are applied as before to give SHAP values for each feature. The algorithmic calculation of these values for arbitrary classification techniques is far from trivial, but is possible for tree ensembles such as those used in this chapter[118].

Despite the abstract description above, SHAP values can be physically interpreted in the current study. All the examples in this document perform calibrated classification using distance matrices as input. The signal additively explained by the given SHAP values corresponds to ΔW. The indi-

[4]Some implementations of SHAP values instead use the marginal expectation value, some label the choice of the marginal expectation as an approximation, and some argue the marginal value is the correct one to use from an interventional perspective[117, 122, 123]. The implementation and physical interpretation presented in this document use the conditional expectation value.

vidual terms in the Eq. (3.3) can be understood as follows: $\mathcal{C}(P_i^R \cup \{i\})$ corresponds to the mean ΔW found when holding the distances specified by $P_i^R \cup \{i\}$ constant and letting the remainder of the protein freely explore the canonical ensemble, and $\mathcal{C}(P_i^R)$ is the same but letting distance i also freely explore[5]. In this way, the SHAP values isolate which parts of a specific protein configuration contribute to its specified ΔW. The same idea can be applied to an feature set of an arbitrary free energy surface: the system is allowed to explore conditional to the given features under investigation.

SHAP values provide a real number for each feature and configuration considered. In this way they can be considered a new set of descriptors for each protein configuration. This set of descriptors is of equal dimensionality to the original data set, and at first seems to provide little advantage relative to the original coordinates. However, three properties of SHAP coordinates are more desirable than the original coordinate system:

1. SHAP values additively relate to ΔW.

2. SHAP coordinates are of the same scale for each configuration and feature. As a result, the relative importance of an inter-domain distance and an intra-domain distance can be directly compared in SHAP space.

3. SHAP values individually reflect manybody correlations in the original data.

As a result, quickly inspecting the individual ranges of SHAP values (using, for example, box plots) can produce insight about problematic aspects of W_{FF} in the original coordinate system. Additionally, we reduce the dimensionality of the generated SHAP coordinates to produce a set of CVs that summarizes the types and severities of errors present in a given CG model. These analyses suggest aspects of the CG model's force-field basis or resolution that may be modified to improve accuracy. It is important to note that SHAP values using conditional expectations, such as those used in this chapter, assign feature attributions using both correlations present in the learning

[5]SHAP values include a single global offset in their additive explanations; however, this distinction does not matter for applications presented in this chapter.

distribution (here, the combined model and reference ensembles) combined with information in ΔW[122, 123]. For example, consider the more generic case of the function $f(x,y) = 2x$ analyzed over a distribution of x and y where x and y are highly correlated. Conditional SHAP values will assign importance to both x and y: this is informally because knowing that y is large also implies that x is large due to the correlation: y contains information about x. As a result, it is difficult to infer the structure of a hypothetical f using said SHAP values. Other feature attribution methods based on Shapley values circumvent this limitation at the cost of considering unrepresentative samples[122]; we leave investigating these alternative methods to a future study.

It is important to note that the term "explanation" is ambiguous: its meaning depends on the method and setting in which it is considered[108–113]. In this work SHAP values provide us with a principled and tractable strategy to associate a particular estimate of ΔW with the intersite distances present in a configuration: each pairwise distance is associated with a additive contribution to ΔW. However, due to the large numbers of features, this procedure alone does not necessarily constitute a useful explanation; it is better instead viewed as a principled decomposition. Two additional analysis techniques are used to convey the information present in this decomposition. First, the mean absolute SHAP value is calculated for each feature in each study. This allows us to quickly isolate which distances typically contribute to ΔW, providing an order in which to investigate various correlations. Second, nonlinear dimensional reduction is used to communicate the global geometry and topology of this error characterization, and, in doing so, understand which errors are codependent. Examining these dimensionally reduced coordinates by visualizing both the physical distances and SHAP values associated with salient pairwise distances found via averaging leads to the plots and conclusions discussed in section 3.5. Examples of these visualizations may be found in Figs. 3.6, B.4, B.5, B.6, and B.7; while analogous plots were generated for all the systems considered they are omitted and conclusions summarized for brevity. As we will see in later sections and chapters, the selected dimensional reduction is also stable to perturbations, and as such also serves as a fingerprint for the error present: models which exhibit similar SHAP distributions through dimensional reduction are inferred to exhibit similar phase space errors (ΔW).

These secondary analyses are not without drawbacks. First, if errors always simultaneously dependent on a large number of features, the mean absolute SHAP values of each of these features alone will be relatively small: they must, after all, additively relate to ΔW. Similarly, if the sources of error are highly heterogeneous across each trajectory, these averages may occlude important patterns. In turn, while the nonlinear dimensional reduction ideally helps convey the relationship between the observed values it lacks direct interpetability in and of itself. Various other methods, such as principal component analysis (PCA), can additionally be used. We have found that in the case of PCA, while the coefficients provide information similar to the mean values discussed, the dataset is not well described in low dimensions as it is using non-linear techniques. Nevertheless, future studies may find other analysis approaches to be more fruitful depending on the feature set used.

3.4 METHODS

The methods used in this chapter can be reproduced using publicly available libraries. Classification was performed using the DART[124] (boosted decision trees with dropout) algorithm in the Light Gradient Boosting Machine (lightgbm) library[125]. This library supports a large number of hyperparameters. Unless otherwise specified, hyperparameters were set to 5 leaves, 5000 trees, and 6150 bins per feature; trees were dropped (DART) at a rate of 0.5 with a max drop of 1000, and each learned tree saw 80% of the feature space and 15% of the full data set (this random subset of data was redrawn every 50 trees). 50% of each molecular data set was used as a training set, while the other 50% were used as a test set. Trees were grown to minimize the log loss. Subsampling data and features sped up training. The results presented are obtained from the test set; however, due to the regularization imposed, performance on the train and test sets were close to identical. Qualitative conclusions made through CVs generated via dimensional reduction were stable to choices of hyperparameters. Estimated pointwise free energy differences were found to be somewhat sensitive to choices of hyperparameters. This is expected: large absolute differences in ΔW imply that a comparison is being made at a location with very low configurational density in one ensemble; the

precise level of population is difficult to accurately estimate without using enhanced sampling. As a result, if one wishes to make a quantitative comparison between models based on the pointwise free energy differences, a large amount of well sampled data must be put through a very careful train-test-validation based framework. In contrast, the approach in this document used classifiers with similar levels of regularization throughout.

Dimensional reduction was performed using PCA combined with the Uniform Manifold Approximation and Projection (UMAP) method[126]. This technique was selected due to observed computational efficiency and separation of clusters, not due to any physical argument. Results generated using t stochastic neighbor embedding[127] (t-SNE) appeared different, but lead to similar physical conclusions. It was observed that depending on the system under study, the structure of the high dimensional data varies greatly (for example, in terms of connectivity), suggesting that different models in the future may require different analysis strategies. Unless otherwise noted, 20 PCA coordinates were used as input to UMAP to create a 2 dimensional projection using 64 nearest neighbors (other parameters were left to their defaults). Qualitative aspects of the generated CVs were found to be consistent across a wide range of parameters; said high number of neighbors was used to avoid small clumps of points which did not improve physical interpretation. k-nearest neighbor regression (KNN regression) was used to map coordinates to UMAP coordinates using the following procedure. First, data was mapped to SHAP values; second, PCA was used to map the SHAP values onto the first 20 principal modes. UMAP was then applied to map these PCA coordinates to UMAP space. This data served as a training set for KNN regression, which was parameterized to map SHAP values to UMAP coordinates. New configurations were processed by first producing their SHAP values; these SHAP values were then transformed using the trained KNN regressor. When projecting data onto SHAP coordinates which were generated from other ensembles KNN regression was always used. For dodecaalanine, KNN regression was trained using $2e4$ data points; the remainder of the samples in each ensemble were also projected using KNN regression. In the case of actin, all $1e4$ samples were processed directly with PCA and UMAP. KNN regression was parameterized to use the 5 closest neighbors.

3.4.1 DODECAALANINE

Dodecaalanine (DDA) is a short polypeptide which adopts a variety of conformations in solution: a hairpin like conformation, a helical conformation, and an extended conformation (see, for example, Rudzinski & Noid [53]). DDA was simulated at the atomistic resolution using Amber18[128] and the Charmm36m[129] force-field. Each of two replicas was solvated with 7121 water molecules (TIP3P) and 20 sodium and chloride ion pairs. This system was relaxed using steepest descent for 5000 steps, followed by equilibration via NVT simulations and NPT simulations. Production samples were extracted every 50 ps from a 5.1 μs trajectory propagated using a 2 fs timestep in the NVT ensemble at 303K using a Langevin[130] thermostat with a damping parameter of 0.5 ps. Hydrogen bonds were constrained via SHAKE. The resulting DDA frames were mapped to a resolution of one CG site per amino acid; coordinates were determined by a center of mass mapping. CG simulations were propagated in LAMMPS[19] (version lammps-7Aug19, lammps.sandia.gov) and were started from an initial structure obtained from the mapped atomistic ensemble. The system was minimized for 5e5 NVE/limit steps, then propagated for 5e5 steps using a Nose-Hoover[131] thermostat with a 0.2 fs timestep and a coupling parameter of 0.5 fs. Production CG simulations were propagated using using a Langevin thermostat with a 2 fs timestep and a damping parameter of 0.1 ps. 1.8e5 total CG samples were taken by sampling every 0.4 ps. Each DDA classification task was performed on 3.6e5 samples: 1.8e5 were randomly sampled from the mapped atomistic replicas, and 1.8e5 were taken from a contiguous CG trajectory.

Relative entropy minimized (REM) models of DDA were iteratively parameterized using the Adversarial Residual Coarse-Graining[67] software framework (git repository available upon request) using the (first-order) gradient equations specified in Shell [47]. Analysis and visualization used the theano[132], numpy[133], lightgbm[125], scikit-learn[134], shap[117, 135], pandas[136], ggplot2[137], data.table[138], and pracma[139] libraries. Each iteration's CG simulation was performed using the settings described above for CG models. Marginal distributions were confirmed to well approximate those implied by the theory underpinning REM. All potentials (bonded and nonbonded) were represented via pairwise interactions. Nonbonded pairwise potentials were rep-

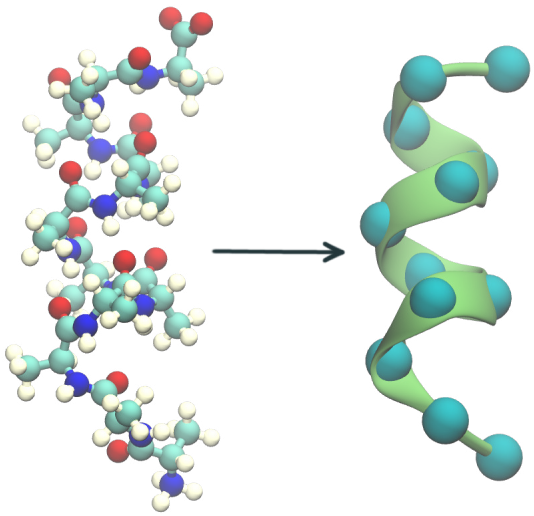

Figure 3.1: The map used to coarse-grain dodecaalanine in the current study. Each amino acid was mapped to a single CG site via a center of mass mapping. Solvent was present in the (not shown) atomistic system but not the CG system. The ribbon diagram is superimposed with the CG representation as a visual guide.

resented using a WCA potential summed with a 3rd order *b*-spline (the additive non-spline terms significantly improve stability during training); bonded potentials were represented using a harmonic potential summed with a 3rd order *b*-spline. 20 equally space spline knots were used for both the bonded and nonbonded splines; the nonbonded splines and tables ranged from 0.001 to 20 Angstroms, and the bonded splines and tables ranged from 0.001 to 10 Angstroms. Both the parameters controlling the WCA and harmonic terms were transformed through exp to enforce positivity. Bonds were present both between nearest neighbors along the CG backbone and neighbors separated by a single site; in the latter case, the bonded interactions emulated angle potentials. At each iteration $1.8e5$ configurations were randomly sampled from atomistic trajectory and used to calculate the REM derivative (along with the entire CG trajectory). Larger sets of configurations for derivatives were not seen to change the REM results, although smaller sets (e.g., $9e4$) were noted to give biased potentials as determined by correlations. For each iterative update, the next set of parameters was generated using the RMSprop algorithm[140] using a rate of 0.009 for all parameters (all parameterization was performed relative to LAMMPS real units). Resulting parameter updates were clipped to a absolute maximum of 0.07. Parameters were updated until no reliable change in

any potential was visible for 30 iterations.

The radius of gyration and Q-helicity were used to quantify the behavior of DDA; their formulation can be found in Rudzinski & Noid [53]. Q-helicity quantifies the similarity of a given configuration to a helix: 0 corresponds to no helical character, while 1 corresponds to a completely helical polypeptide. Similar CVs are present in enhanced sampling studies, see Piana & Laio [141] and Prakash *et al.* [142].

3.4.2 ACTIN

ATP Actin was simulated at the atomistic resolution using GROMACS 5.1.4[143]; the protein was solvated with 29530 waters (TIP3P) and 107 pairs of potassium and chloride ions. Equilibration details may be found in Hocky *et al.* [144]. Production simulations were performed for 1 μs using CHARMM27+CMAP[145] in the NPT ensemble at 310K and 1 *bar* using the stochastic velocity rescaling thermostat[146] with a coupling parameter of 0.1 *ps* and a Parrinello-Rahman barostat[147] with a coupling parameter of 2 *ps*. Hydrogen bonds were constrained via LINCS. Samples were collected every 100 *ps*. The atomistic frames were mapped to the CG resolution using the map found in Saunders & Voth [148]. Briefly, sites indexed 1-4 represent actin's four main subdomains which are approximately arranged at the 4 corners of a square; site 9 represents the nucleotide ADP situated at the center of these 4 subdomains, and site 5 represents the D-loop, a semistructured region connected to site 2. The map is characterized in Fig. 3.2 (adapted from Saunders & Voth [148] with permission). Hetero-elastic network models (hENMs) were created using the procedure in Lyman *et al.* [149] using this atomistic trajectory. In certain portions of the results, models are parameterized using all 12 of these sites. In other portions only sites 1-4 (1 based indexing) are used. ADP actin was additionally equilibrated using the techniques described in Hocky *et al.* [144], and propagated using an approach identical to that described above except that 10000 production samples were acquired by sampling every 50 *ps* providing 500 *ns* of sampling. CG samples of actin were generated using LAMMPS (version 17Nov2016). Initial configurations were taken from the mapped atomistic trajectory. The system was propagated using a timestep of 1

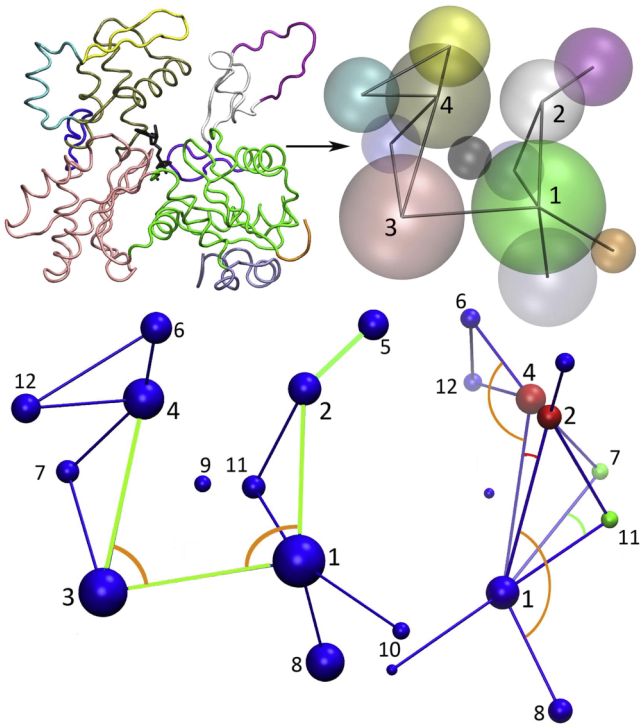

Figure 3.2: The map used to coarse-grain actin in the current study. Each set of atoms was mapped to a position via a center of mass mapping. CG models of actin were simulated and modeled at both the resolution of all 12 sites presented and at the resolution of sites 1-4.

fs via a Langevin thermostat with a damping parameter of 50 fs. $2e5$ CG samples were obtained by sampling every 10 ps. The relaxation steps used for actin were less aggressive than those used in DDA as no REM optimization was needed. When performing classification based comparisons, similar procedures were followed except the size of the data was significantly smaller: $1e4$ samples were used from each data source (the entirety of the mapped atomistic simulation was used, and the CG trajectory was randomly subsampled). All ADP-Pi actin data used for filament analysis was prepared as according to reference [150].

3.5 RESULTS

SHAP coordinates were first used to explain errors in models of DDA and actin. As elaborated on in section 3.4.1, when considering DDA a single resolution of 12 sites was used with multiple pairwise spline force-fields, each of which was minimized using REM. The CG resolution was

defined as the center of mass of each amino acid. In the case of actin, models consisting of fully connected networks of harmonic springs at two resolutions (12 and 4 sites) were studied, each of which were parameterized using the hENM method as described in section 3.4.2; as a result, all harmonic springs present differed. Section 3.5.1 studies the effect of various force-field bases on DDA, providing examples of using SHAP based analysis to understand and improve existing force-fields. Section 3.5.2 analyzes the role of the CG mapping operator when modeling actin, providing an example of how ΔW and the resulting SHAP coordinates relate to selecting model resolution. Section 3.5.2.2 adapts these analysis techniques to investigate the effect of the hydrolysis of ATP to ADP on subdomain behavior; finally, section 3.5.2.3 describes an initial application of classification to understanding molecular assemblies.

3.5.1 DODECAALANINE

Dodecaalanine was modeled using five different force-fields. Each force-field was composed solely of pairwise interactions at the bonded and nonbonded level. Sites adjacent along the backbone were connected via bonds; sites separated by a single site were connected via an additional bond to emulate an angle potential. Within each type of interactions (bonded, angle-bonded, and non-bonded), a unique interaction type was defined for each possible pair of site types. Each model can be considered as extending the model before it.

The first model was composed of a single site type. The pairwise nonbonded interactions were set to be constant, i.e. nonbonded pairwise forces were uniformly zero. This resulted in two unique pairwise interactions (one bonded and one angle bonded). The second model was identical to the first model, but included nonzero pairwise nonbonded interactions. The third model then included a single additional collective site type for the termini, resulting in two site types total. The fourth extended the model by considering the site types at each termini to be distinct, resulting in three site types. The final model was extended to five site types by providing additional unique site types to each CG bead adjacent to a termini bead. Only a subset of these models is analyzed in certain sections for brevity.

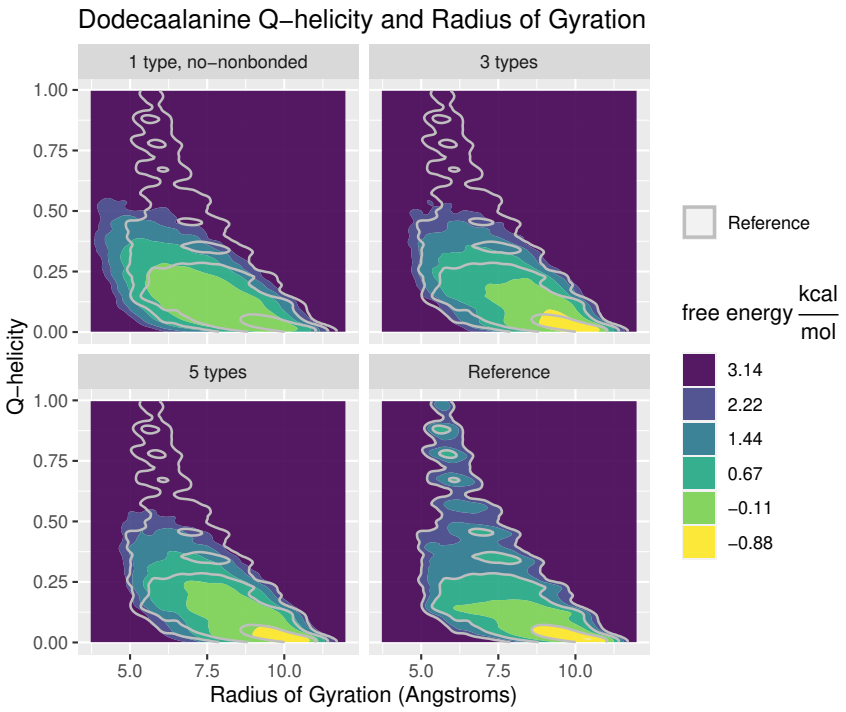

Figure 3.3: Free energy surfaces of the radius of gyration and Q-helicity for various dodecaalanine models and the reference distribution. The grey overlay (and lower right panel) is given by the reference density and is present as a visual guide. Clear differences exist between the models and the reference distribution, but little difference can be seen between various models, despite significant differences in the force-field basis.

Established CVs, such as the radius of gyration and Q-helicity (see section 3.4.1), show significant differences between the reference atomistic data and the various models: highly helical states are not captured. The difference between a select few models and the reference free energy surface is shown in Fig. 3.3. However, little difference is apparent between various CG force-fields, save for a small amount of extra distortion present in the single site model lacking nonbonded interactions.

Classification was performed between each model and the reference data, with the results shown in Fig 3.4. We note that Fig. 3.4 characterizes the distribution of errors at the resolution of the CG Hamiltonian, but that ΔW is also a valid structural CV[151] that is optimized to separate the pair of ensembles it is generated from. ΔW additionally numerically corresponds to the difference in pointwise free energy values at the resolution given by considering ΔW as a CV[6]. There exist

[6]If not true, there exists a $\eta_{\Delta W}$ at the resolution of ΔW with less error, which implies that the pullback of $\eta_{\Delta W}$ would also outperform η at the model resolution, which is a contradiction if using

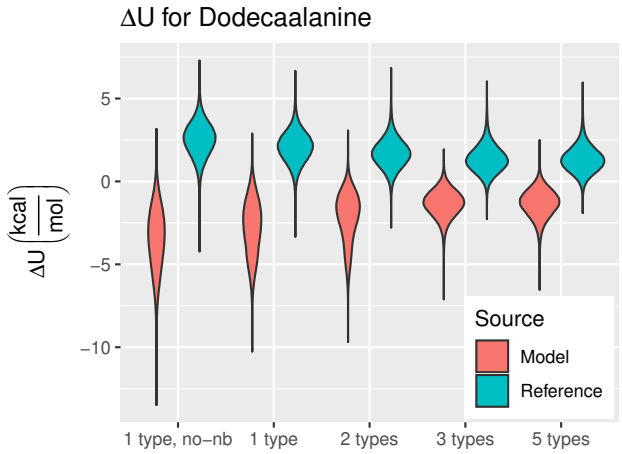

Figure 3.4: Violin plots of ΔW for multiple models of dodecaalanine. Each model distribution is divided along the reference and model ensembles. Note that varying models modify ΔW, changing the shape of the projected reference distribution.

other CVs that equivalently recapitulate ΔW, including the distribution in the full SHAP feature space. The dimensionally reduced coordinates are not guaranteed to share this property, but seem to recapitulate the overlap shown by ΔW alone.

In contrast to the lack of change present in Fig. 3.3, increasingly complex force-field bases produce notable changes in the distribution of ΔW. More significant drops in the magnitude of error can be seen after addition of nonbonded interactions and the introduction of a third site type. Interpreting ΔW as a typical CV, it is also apparent that all of the models considered have relatively poor overlap with the reference data set.

3.5.1.1 UNIFORM NO-NONBONDED MODEL

SHAP variables were constructed for the 1 type DDA model lacking nonbonded interactions using the per-feature output of the shap library (TreeExplainer) as input for UMAP based dimensional reduction (Fig. 3.5). The lack of overlap seen in Fig. 3.4 is preserved, along with additional organization representing the various sources of error. While UMAP lowers dimensionality to allow visualization, it does not provide explicit formulas for the resulting CVs. However, examination

a proper loss function. This is implicity used when featurizing classification problems. Equality of η implies equality of the relative entropy at the model and ΔW resolutions.

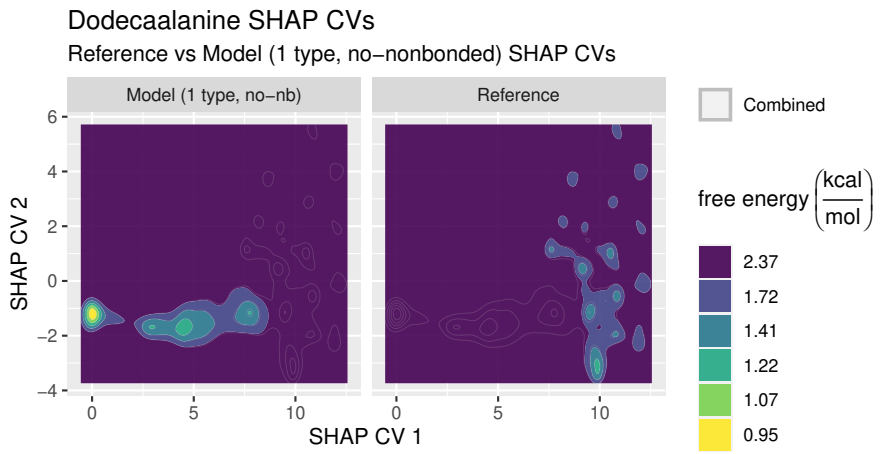

Figure 3.5: Free energy surfaces produced along the SHAP CVs generated from the 1 type no-nonbonded dodecaalanine model. Light grey lines characterize the combined density of the model and reference data and serve as a visual guide.

of the individual aggregated SHAP values (the numerical output of TreeExplainer) themselves provides immediate insight into the distances characterizing ΔW. For example, the feature with the largest mean absolute SHAP value by a factor of 3.5 is the N-terminal CG bond; the next set of maximal SHAP values spans distances between sites separated by a single site (e.g. backbone index 3 and backbone index 5). Plotting the feature with the largest mean absolute SHAP value, the N-terminal bond distance, with the SHAP CVs (Fig. 3.6) demonstrates that the elongated error present from $(0, -1.7)$ to $(8, -1.7)$ is primarily due to this bond distance. Repeating this visualization with the remaining dominant features implies that larger scale N-terminal effects (combined with minor distortion at the C-terminus) dictate the width of this feature. Comparison of existing CVs against the generated SHAP CVs provides information on their interdependence. For example, comparison of the values of radius of gyration and Q-helicity (not shown) with the generated SHAP coordinates implies that the variety of small islands present in the right of the reference ensemble primarily represent both partially helical states and minor bond distance errors in the middle of the protein, while the fully extended configurations are diffusely present around $(8, -1.5)$. Although the helical errors are spread across multiple features, they correspond to single islands in the SHAP based projection, insight that is primarily gained due to the nonlinear dimensional reduction.

Both nonbonded interactions and a single additional type for the termini (the 2 type model)

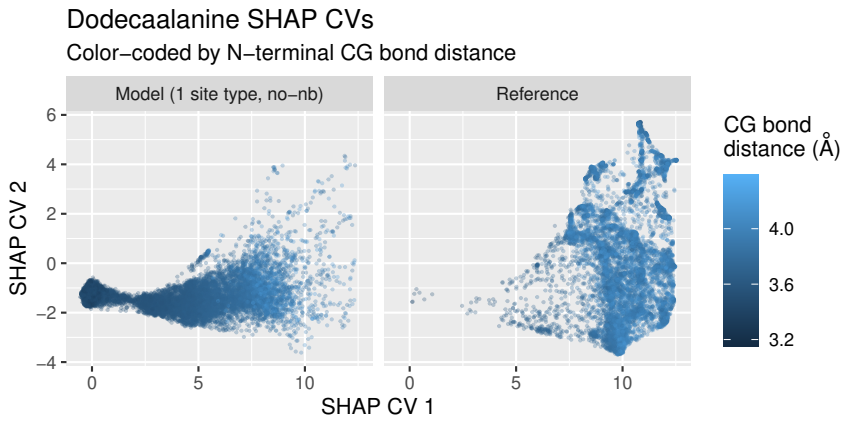

Figure 3.6: Samples from the 1 type no-nonbonded dodecaalanine and reference ensembles projected onto the SHAP CVs in Fig. 3.5. Samples are colored by the distance between the N-terminal CG site and its nearest bonded neighbor.

each reduce the elongated error feature dominating the model lacking nonbonded interactions; however incorporating a distinct bead for each each termini (the 3 type model) fully removes the elongated error present in Fig. 3.5, as can be seen by projecting the 3 type model ensemble onto the coordinates present in Fig. 3.5 (Fig. 3.7). The additional site types also slightly increase the density present in various partially helical states, but not sufficiently to register in Fig. 3.7.

3.5.1.2 THREE-SITE MODEL

In order to better describe the remaining error in the more complex models, new SHAP CVs were generated based on the comparison of the 3 type ensemble to the reference data (Fig. 3.8). The global geometry described by the resulting SHAP CVs differs significantly from that present in Fig. 3.5: as expected, the bond errors dominating the 1 type model without nonbonded interactions are no longer present. Inspection of the individual mean absolute SHAP values no longer isolates a particular portion of the protein as broadly responsible for ΔW. The large diffuse area present (as well as the protrusion at $(10,0)$) in the model ensemble is due to erroneous helix-like formation at the C-terminus, a phenomena spread over 4 sites. The subbasins near $(8,2)$ are due to differences in the inner walls present in ΔW. Sharper features around $(4,6.5)$ are due to the remaining helix formation (primarily present though 1-4 distances), while features around $(10,7.5)$ are due to a

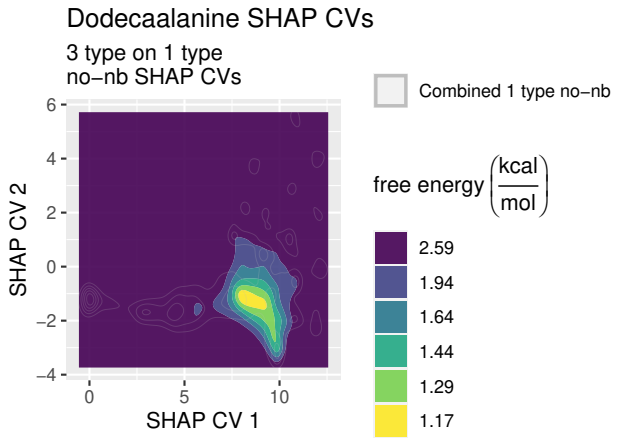

Figure 3.7: 3 type free energy surface projected along the SHAP CVs generated from the 1 type no-nonbonded dodecaalanine model. Filled surfaces represent the 3 type model ensemble, while grey lines represent the combined 1 type no-nonbonded and reference ensembles presented in Fig. 3.5.

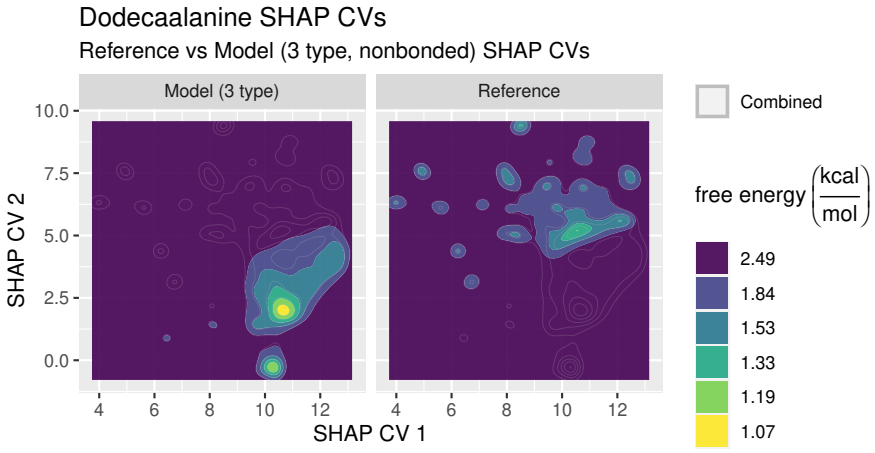

Figure 3.8: The free energy surface projected along the SHAP CVs generated from the 3 type dodecaalanine model. Grey lines characterize the combined density of the model and reference data and serve as a visual guide.

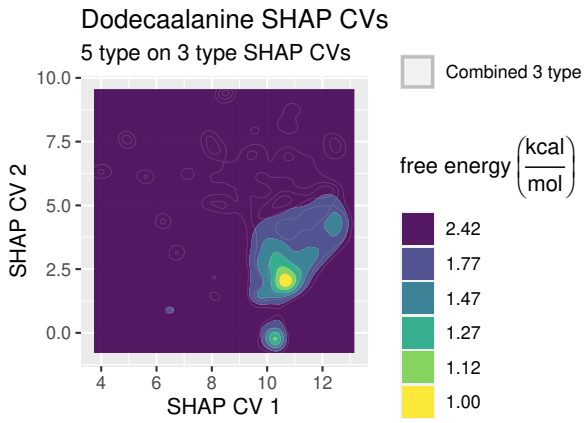

Figure 3.9: 5 type free energy surface projected along the SHAP CVs generated from the 3 type dodecaalanine model. Filled surfaces represent the 5 type model ensemble, while grey lines represent the combined 3 type and reference ensembles presented in Fig. 3.8.

higher propensity for shorter bond distances and angles throughout the middle of the protein. These types of errors are degenerate along the protein backbone, resulting in many minima. The complex nature of the presented error makes it difficult to isolate small adjustments to the force-field basis to improve the model. Attempting to improve the quality of the model by adding additional site detail to each termini results in little improvement, consistent with said error analysis. This is seen in both Fig. 3.4 and 3.9, where Fig. 3.9 visualizes the ensemble produced by the 5 type model projected into the coordinates created from the 3 site model. This lack of improvement contrasts strongly the improvement seen in Fig. 3.7, supporting the view that additional parameters, if not targeted towards known areas of error, do not necessarily improve model performance.

3.5.2 ACTIN

ADP and ATP actin were first analyzed using two different elastic network models created using the heteroENM methodology[149]: one set of models was created at a 12 site resolution, while the other was created using a 4 site subset of the 12 site model (sites indexed 1 through 4). The results for ADP actin are primarily shown in the main text while the results for ATP actin are found in appendix B. Second, atomistic simulations of ADP and ATP were directly compared to one another at said 4 and 12 site resolutions without the use of external CG models. Third, an atomistic

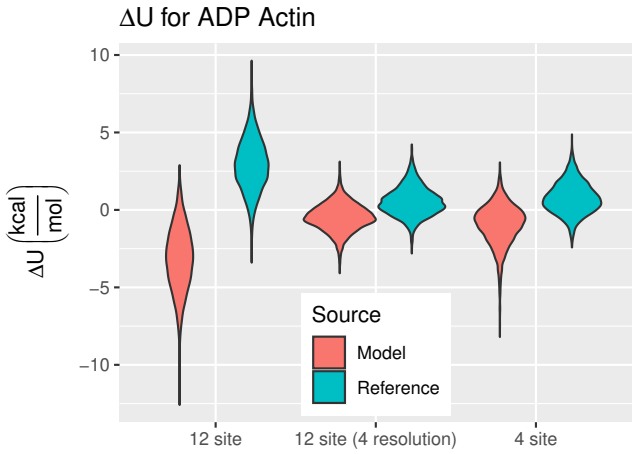

Figure 3.10: Violin plots of ΔW for multiple models and resolutions of ADP actin, divided along the reference and model ensembles. The "12 site (4 resolution)" and "4 site" models are compared to the reference at the 4 site resolution, while "12 site" is compared at the 12 site resolution. Note that varying models/resolutions change the form of ΔW, which changes the shape of the projected reference distribution.

simulation of a filament of ADP-Pi actin was analyzed to understand its internal heterogeneity. Additional visualizations related to content in the main text are found in appendix A.

3.5.2.1 ADP ACTIN

The first measure of error discussed is the distribution of ΔW found for various models of ADP actin (Fig. 3.10). In order to better understand model error across resolutions, the samples produced by the 12 site ADP model were additionally mapped to the 4 site resolution for comparison to the 4 site model. The distribution of ΔW for the 12 site model at both resolutions as well as the errors in the 4 site model are visualized in Fig. 3.10. At the 12 site resolution, the 12 site model exhibits a large spread of ΔW. When mapped to the 4 site resolution, however, it performs marginally better than the 4 site model when compared to the reference ensemble. SHAP variables were generated between the 12 site elastic network and the reference ensemble (Fig. A.1) at the 12 site resolution; a diffuse central basin is present, along with a variety of smaller subbasins. Collectively, the majority of the error present is associated with sites 5 and 9, which represent the D-loop (a transiently structured region located on the edge of the protein) and the nucleotide ADP (situated in the center

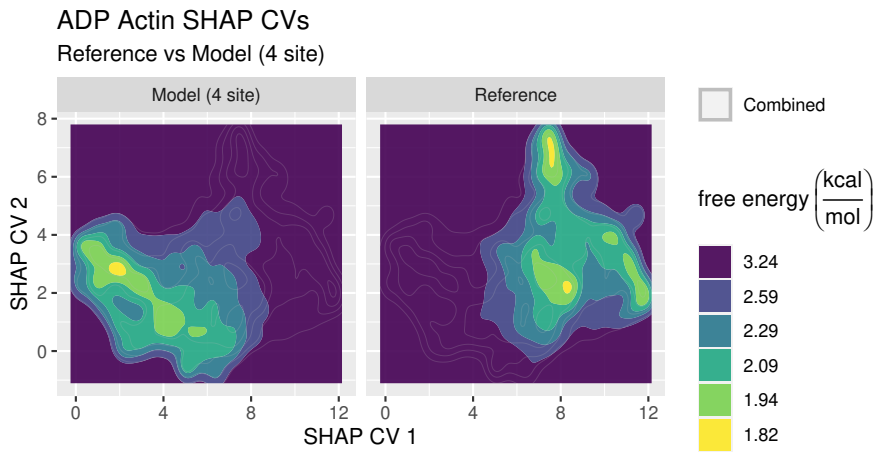

Figure 3.11: The free energy surface given by the SHAP CVs calculated by comparing the 4 site elastic network to the reference ensemble at a 4 site resolution.

of the protein). This suggests that a model built at the 4 site resolution may result in less error. The harmonic network model built at the 4 site resolution was used to test this hypothesis; the resulting SHAP CV based comparison is shown in Fig. 3.10 with the corresponding error also present in Fig. 3.11. As previously mentioned, the spread of ΔW is indeed reduced relative to the 12 site model when compared at the 12 site resolution. Inspection of plotted SHAP values indicates that the horizontal spread (along CV 1) is due to error associated with the 2-4 and 3-4 distances, while the vertical spread is attributable to error in the 1-4 distance.

However, ΔW for the 4 site model is naturally determined at a different resolution than that of the 12 site model at the 12 site resolution. As where the 4 site model is only judged on the behavior of sites 1-4 (which represent the core of each of the 4 main subdomains), the 12 site model is compared using all 12 sites— and as a result must capture more complex behavior. This likely does not align with error validation in practice: if both a 4 site and 12 site model are considered, it is likely that only sites 1-4 are critical (if not, the 4 site model would be an invalid choice regardless of its accuracy). In other words, the pertinent behavior of the 12 site model may very well be concentrated in sites 1-4. As alluded to when discussing Fig. 3.10, the current analysis allows us to analyze the 12 site model at the resolution of sites 1-4 by mapping the 12 site model to the 4 site resolution. The resulting mapped 12 site ensemble is then projected onto the variables generated by

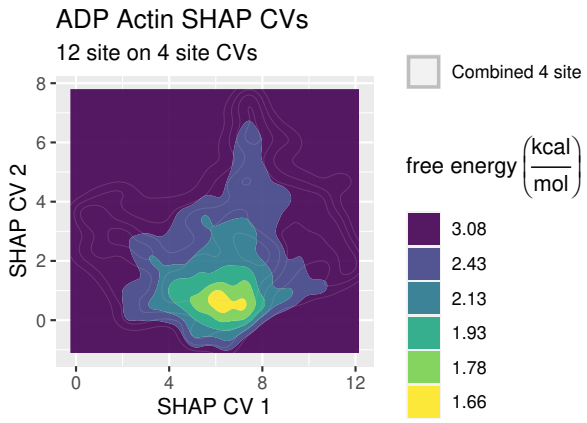

Figure 3.12: 12 site free energy surface produced along the SHAP variables generated by comparing a 4 site elastic network model to mapped reference data. Filled surfaces represent the 12 site model ensemble, while grey lines represent the combined 4 site model and reference ensembles presented in Fig. 3.11.

the SHAP variable 4 site model (Fig. 3.12). The offset of the 12 site ensemble in CV space (along with Fig. 3.10) implies that errors related to the 2-4 and 2-3 are resolved in the 12 site model; this is reinforced by considering the free energy surface given by said distances (Fig. 3.13).

3.5.2.2 ADP AND ATP ACTIN

The previous subsection used ΔW and SHAP based analysis to understand the difference between a model of actin and the mapped atomistic data it approximated. In this subsection we compare two mapped atomistic actin ensembles that differ in the nucleotide present in site 9. Specifically, we compare ATP and ADP actin at the resolution of the 12 and 4 site maps. The hydrolysis of ATP to ADP in actin is known to have direct relevance to the regulation of the cytoskeleton and provides insight into the of effect actin polymerization on rate of hydrolysis[150, 152, 153]. The mean absolute SHAP (MAS) values for the comparison at the 12 site resolution are given in table A.1; high prevalence of the nucleotide and D-loop are again present. The corresponding SHAP coordinates fully separate the ensembles and do not provide strong interpretive value (Fig. A.2). Horizontal spread is roughly due to nucleotide behavior combined with D-loop positions, while distances involving site 1 dominate the vertical spread. Comparison of the individual free energy surfaces indicated by table A.1 shows little overlap between the two ensembles; see, for example,

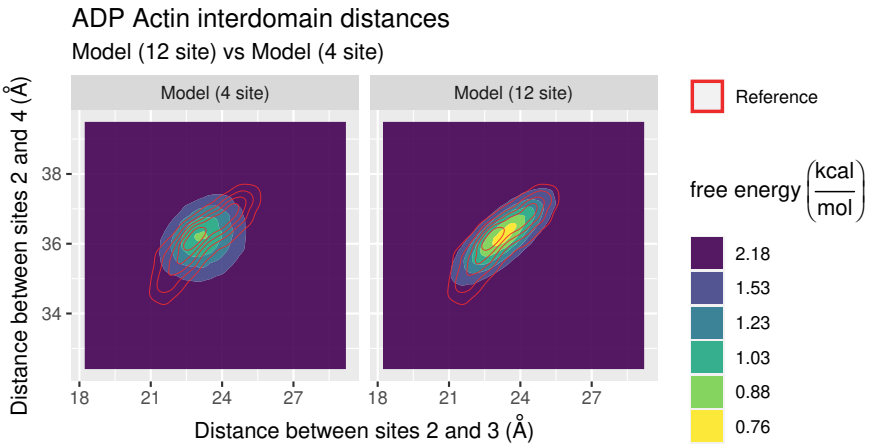

Figure 3.13: Free energy surface produced along the distance between sites 2 and 4 and the distance between sites 2 and 3. Filled contour panels represent the two model ensembles, while the red line contour overlay correspond to statistics from the reference ensemble.

Fig. 3.14, A.3, and A.4. This reinforces the known[152] dependence of the nucleotide binding pocket and D-loop behavior on the nucleotide state.[7] It is perhaps more insightful to ask whether

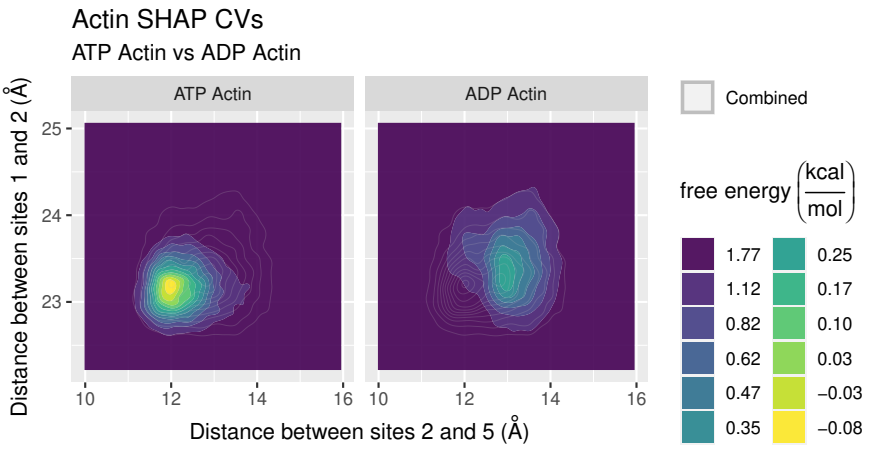

Figure 3.14: Free energy surfaces produced along the 2,5 and 1,2 distances for the ATP and ADP actin ensembles.

the four main subdomains of actin (sites 1-4) register significant configurational differences upon the conversion of ATP to ADP. This is probed by performing SHAP analysis at said resolution, the leading MAS values of which are shown in table 3.1. The corresponding SHAP based coordinates are seen in Fig. 3.15. Variation in SHAP CV 2 is due to the 1,4 distance while variation in

[7]We additionally note that the differing chemical composition of site 9 also likely affects its behavior in the current analysis.

Distance	1,4	2,4	2,3	1,2	3,4	1,3
MAS	2.11	1.29	1.12	0.88	0.69	0.50

Table 3.1: The mean absolute SHAP (MAS) values found when comparing ATP actin to ADP actin at the 4 site resolution. Distance refers to the indices of the two sites between which the distance is defined. Note that the MAS are here presented in units of $k_b T$.

SHAP CV 1 is due to the 2,4 and 2,3 distances; this is reinforced by noting the lack of overlap present in the free energy surface given by the 1,4 and 2,4 distances (Fig. 3.16). Performing this

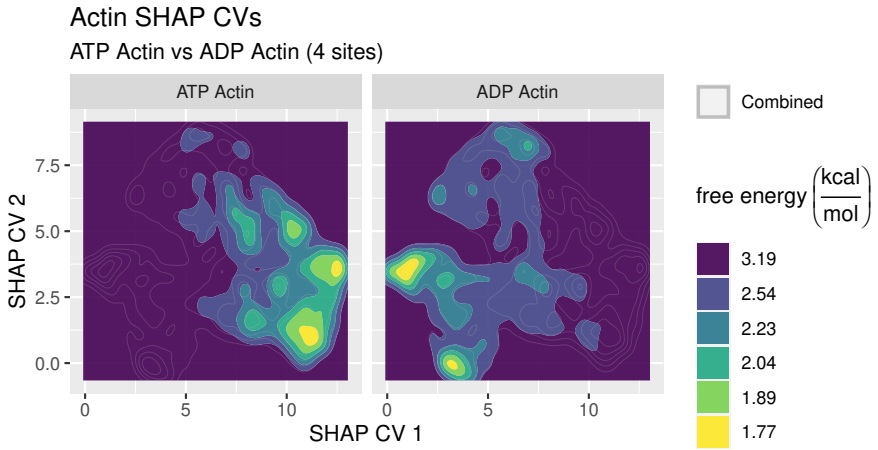

Figure 3.15: Free energy surfaces produced along the SHAP variables generated by comparing ATP to ADP actin at the 4 site resolution.

analysis at the 4-site resolution indicates that while significant overlap is present at the resolution of these major subdomains, the two ensembles are still relatively distinct. A similar analysis may be useful in determining the effects of polymerization or ion binding events upon major subdomain organization[144].

3.5.2.3 ACTIN FILAMENTS

The proposed classification approach can be scaled to understand heterogeneity in large protein complexes composed of identical units. While in previous subsections we studied individual pairs of ensembles in detail, we here provide preliminary results found when studying the overall structural variation found in a single 13-unit actin filament. Each unit corresponds to an individual actin monomer in the ADP-Pi state, although identical analysis may be performed on filaments in

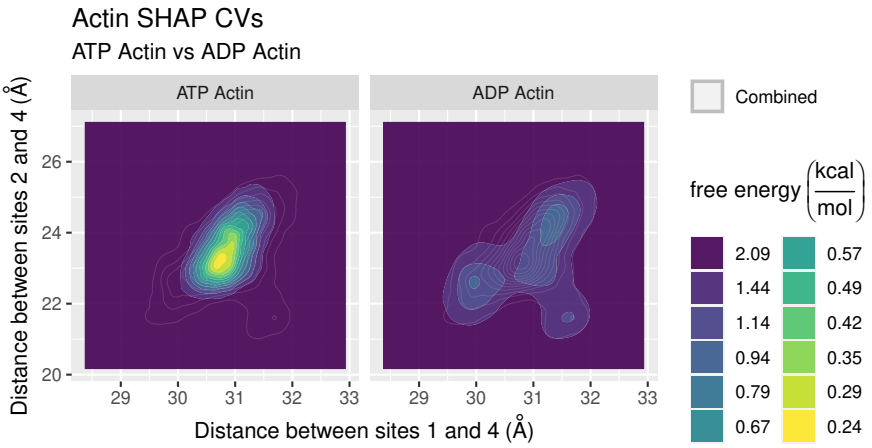

Figure 3.16: Free energy surfaces produced along the 1,4 and 2,4 distances for the ATP and ADP actin ensembles.

either the ATP or ADP state. Said actin units are organized in a helix consisting of two strands, each of which is composed of a chain of actin monomers. Actin filaments are known to be heterogeneous[150, 154, 155]: in other words, the behavior of monomers along the filament varies. However, as discussed in the previous sections, the comparison of protein ensembles at even modest resolutions is hampered by their high dimensionality.

Actin filaments are dynamic structures which perform a variety of biological functions in the cell. These roles include the hydrolysis of their contained nucleotide, ATP, we all as their spontaneous polymerization and depolymerization in response to external stimuli. The impact the configuration of each monomer has on these various functions, especially the cooperative effect various monomers have on their neighbors, is of high interest (see, for example, references [150, 153, 156]). The phenomena pertinent to our current investigation is the addition and elimination of individual monomers to an existing actin filament. Actin filaments are known not to be functionally symmetric: The rate of monomer association is higher at one end of the filament (termed the barbed end) than the other end (termed the pointed end). The structural source of this discrepancy was studied in reference [150], which found that structural heterogeneity was gradually found at the ends of the filament, while interior actin units seemed relatively homogeneous. We here extend this analysis using our proposed classification based approach.

In order to extend our method to the resolution of molecular assemblies we first use classification to derive the average accuracy (i.e., the average of the 0-1 loss, see section 2.2) found during classification with any two constituents of the complex under study. In the case of the actin filament under consideration, the assembly was composed of 13 units, resulting in 78 pairwise classification comparisons between the various units present in the filament. Each classification problem only considered two monomers at the 4 site resolution[8] considered previously; information related to manner in which the monomer was arranged in the filament was not used. These pairwise accuracy values are visualized in Fig. 3.17. Note that the diagonal of these heatmaps is minimized at 50% as the comparison of a monomer to itself results in the lowest possible accuracy. Darker colors are generally found near the pointed end of the filament, but are sporadically seen throughout. Unit 7 is a noted outlier relative to its neighbors. Upon clustering (Fig. 3.17) it becomes clear that units 1,2, and 7 are relatively distinct from the remainder of the filament, while overall similarity is present in the barbed end (with the exception of unit 11). The general level of distinguishability, even at this coarse resolution of 4 sites, is high throughout the filament.

This high degree of heterogeneity is seemingly at odds with existing analysis of actin[150]. Previous studies have focused on the behavior of structural variables that are known to be strong indicators of configurational differences between polymerized and monomeric actin. These analyses, primarily based on the dihedral angle connecting connecting the four subdomains of actin, suggest (Fig. A.5) that the barbed and pointed ends differ while the internal actin units are relatively homogeneous. The patterns observed Fig. 3.17 show, however, that these patterns do not hold when more collective collective are considered, resulting in interesting preliminary conclusions. First, the heterogeneity present at the barbed end, including that found between units 1 and 2, is greater than the corresponding behavior at the pointed end; furthermore, unit 7 is relatively similar to the barbed end. Second, strong similarity of pairs of monomers, such as 12 and 13, 9 and 10, and 6 and 11, is present, while general similarity among internal units is not. The molecular source of these heterogeneities is currently being investigated.

[8]Classification was also attempted at the 12 site resolution; however, all monomers were found to effectively distinct ($> 97\%$ accuracy), impeding drawing conclusions about filament trends.

The SHAP based analysis used in previous sections can be applied to the comparison between each unit; however, doing so easily produces an overwhelming amount of information. Comparison of the MAE values for each pair of units (not shown) suggests that source of heterogeneity present in the filament is not uniform; in other words, while it is easy to describe the difference between two monomers using a small number of variables, the suitable variables change for different monomers, complicating the description of the filament as a whole. For example, while the distance between sites 3-4 and is central to distinguishing unit 1 from the rest of the filament, the distances between sites 1-2 provides more information for comparisons involving units in the center of the filament. It is not clear if the complexities detected, however, have biological relevance; current analysis at the atomistic resolution will prove to be essential in determining what type of heterogeneity may affect physical behavior. Nevertheless, these results draw the view that any portion of the filament is self similar is into question, even at the 4 site resolution.

3.6 DISCUSSION AND CONCLUSIONS

The results presented demonstrate three uses of SHAP CVs. First, the error of an existing model can be described and used as a guide to adjust the force-field basis. These CVs can similarly be used to quantify the effect of a proposed change to said basis. Second, the effect of resolution on the accuracy of a CG model can be quantified: in this case, an increase in resolution improved the behavior of a CG model quantified at a coarser resolution. Third, SHAP CVs can be used to quantify the large scale effect of molecular changes; this final approach was additionally extended to investigate the heterogeneities in molecular assemblies. While the conclusions and model adjustments are not necessarily counterintuitive, they were produced with minimal human intervention.

In the case of DDA, the largest reduction in error was seen when accounting for bond disagreements. This is supported by the fact that the large free energies found in effective bonded interactions can easily result in areas of phase space that are effectively not traversed by either the model or reference data, as a lack of overlap in any dimension implies a lack of overlap at the full phase space resolution. The conclusions drawn from the SHAP CVs contrast strongly with

conclusions drawn from pre-selected CVs: while Fig. 3.3 suggests that the CG model occupies a subset of the phase space favorable to the mapped atomistic data, Fig. 3.4, along with Figs 3.5 and 3.8, illustrate that the two ensembles are close to disjoint, while the latter two figures still represent variation in Q-helicity (and to a lesser extent, radius of gyration). This does not imply that pre-selected CVs cannot correctly discover issues with overlap— ΔW is a CV in itself and may be selected/approximated using external knowledge.

In the case of actin, it is perhaps surprising that at a 12 site resolution the distribution of ΔW is of a similar magnitude as that found in DDA: actin has a persistent tertiary structure while DDA is relatively disordered. However, it is important to note that actin was modeled using harmonic potentials as where DDA was modeled using spline interactions. Based on SHAP based analysis, this error is primarily due to difficulty modeling the D-loop and nucleotide, a conclusion supported by the reduction of error when considered at the 4 site resolution. Notably, however, the 12 site model performs better than one constructed without these problematic sites due to improved reproduction of cross correlations. This is attributable to the limited basis of the force-field, and suggests that approaches which select CG mappings without considering the limited force-field basis that is eventually applied will not produce mapping operators that are necessarily useful in practice. Similarly, if the proposed approach is combined with a series of maps at varying resolutions, the effect of model resolution can be validated in a novel and rigorous way. The interplay between ΔW and resolution supports the idea that the correctness of an approximate CG model may be difficult to consider without an implied resolution for analysis. The study of the effect of ATP hydrolysis in turn suggests that similar techniques may be used to understand the multiresolution effects of microscopic changes such as hydrolysis, which may help enhanced sampling studies select pertinent collective variables, similar to work in Sultan & Pande [151]. Similarly, the presented preliminary results on actin filaments show that classification may useful for understanding the heterogeneity driving function in molecular assemblies, and more specifically draws into question views of relative homogeneity in actin filaments overall, even when viewed at a coarse resolution. The biological implications of these ideas are under current investigation.

3.6.1 INVARIANCE AND ADVERSARIAL LEARNING

The relationship between η and ΔW (Eq. (3.2)) provides important insight to classification in the current context. First, if the force-fields generating both ensembles are known, the information provided by classification can alternatively be gleaned by calculating the overall difference in free energies using, for example, the Bennet Acceptance Ratio[157]. Similarly, if performing classification between a CG model and a mapped reference ensemble, estimating the ideal classifier is analogous to estimating the true manybody-PMF along with the corresponding free energy difference. Additionally, Eq. (3.2) directly implies that any symmetry or locality shared by W_{FF} and W_{PMF} is shared by η. This has important consequences when considering applying the proposed XAI approach to novel chemical systems: the corresponding classification problem will contain physical symmetries and locality, and approaches which do not take this into account will likely provide poor estimates of η. A straightforward example is a homogeneous liquid where an asymmetric classifier is trained on the Cartesian coordinates: sampling sufficient to converge various local correlations (e.g., radial distribution functions) may be insufficient to parameterize such a classifier. On the other hand, functions which are formulated to obey permutational and rototranslational symmetries, while widely investigated as techniques for atomistic and CG force-field development[13, 158, 159], will require appropriate explanation methods.

Systematic coarse-graining methodologies typically define a numerical measure of error and then return the force-field which minimizes said error. Classification suggests similar ideas of global error based on the accuracy achievable when performing classification between the ensemble implied by the force-field and the reference ensemble: a lower level of mean accuracy implies better emulation of the reference statistics (the accuracy is minimized if the reference and model are indistinguishable; this results in a constant η of 0.5). A natural question is then to consider force-fields which are optimized using this particular measure of quality. These force-field optimization approaches lead to adversarial learning, an approach firmly established by generative adversarial networks[82], which when applied to CG force-field development is termed Adversarial Residual Coarse-Graining (ARCG)[67], which we fully characterize in chapter 4. The properties

of η described in the previous paragraph additionally often apply to adversarial learning[9]. The error estimation present in ARCG[67] can resultingly be viewed as simultaneously providing an estimate of W_{PMF} and the difference in configurational free energy. Conversely, the variational error in ARCG can be calculated without performing any classification: if a higher order force-field is used to approximate W_{PMF} and supplemented with a free energy difference method, the derivatives updating the parameters are similarly calculable through Eq. (3.2). Additionally, as η is central to adversarial residuals[67, 81], the SHAP based explanations proposed in this chapter are fundamentally related to global residuals such as the relative entropy and the Hellinger distance. The value of these various divergences provide a quantification of the overlap of ΔW described Figs. 3.4 and 3.10; however, the numerical values of these divergences are difficult to interpret without context.

CG models are often created to study specific phenomena, and it may not be necessary to perfectly produce all the behavior of the mapped atomistic system. In this cases the proposed methodology can be adapted by customizing the resolution at which it is performed, such as in the example of actin. However, more broadly, the concept of bottom-up error analysis as presented here may not be appropriate for these models. The modeler must decide whether to view the model as a way to reproduce specific phenomena or whether to view the model as a drop in quantitative replacement for atomistic simulation. Certain coarse-graining strategies, such as ARCG, can parameterize a force-field to reproduce the manybody behavior of a subset of the particles present in the CG system. However, doing so incorporates additional human influence into the creation of said CG model: as the resolution becomes coarser, the approach begins to resemble top-down parameterization strategies. We note that machine learned atomistic force-fields are often quantified including values similar to ΔW[106, 160]; if CG models are to be eventually considered as accurate as their fine-grained counterparts, utilizing similar measures of quality would seem to be critical.

[9]More precisely, they apply when the variational process corresponds to classification. The adversarial approach has since been expanded to encompass additional divergences unrelated to classification.

3.6.2 XAI AND FUTURE DIRECTIONS

The analysis in this chapter focused on using SHAP values to describe the behavior produced by CG potentials, and more generally demonstrated a method to describe the differences present in high dimensional free energy surfaces. The approach trains a classifier to estimate ΔW, and then used techniques from XAI to explain said estimate. Interpretable models and explanations intrinsically provide a way to understand the high dimensional differences characterizing the quality of a proposed CG force-field, and we fully expect that other methods from the rapidly developing field of XAI will find similar utility. Furthermore, the study of explanations and interpetability is fundamentally relevant to the creation of CG models: CG force-fields are rarely created solely to reproduce the manybody-PMF of the training ensemble. They are instead often created to either extract knowledge from the system under study or to investigate new physical settings, tasks that intrinsically require human understanding of the limitations and workings of the utilized CG model. Any technique for bottom-up CG model creation which uses external human validation is a candidate for using explainable techniques.

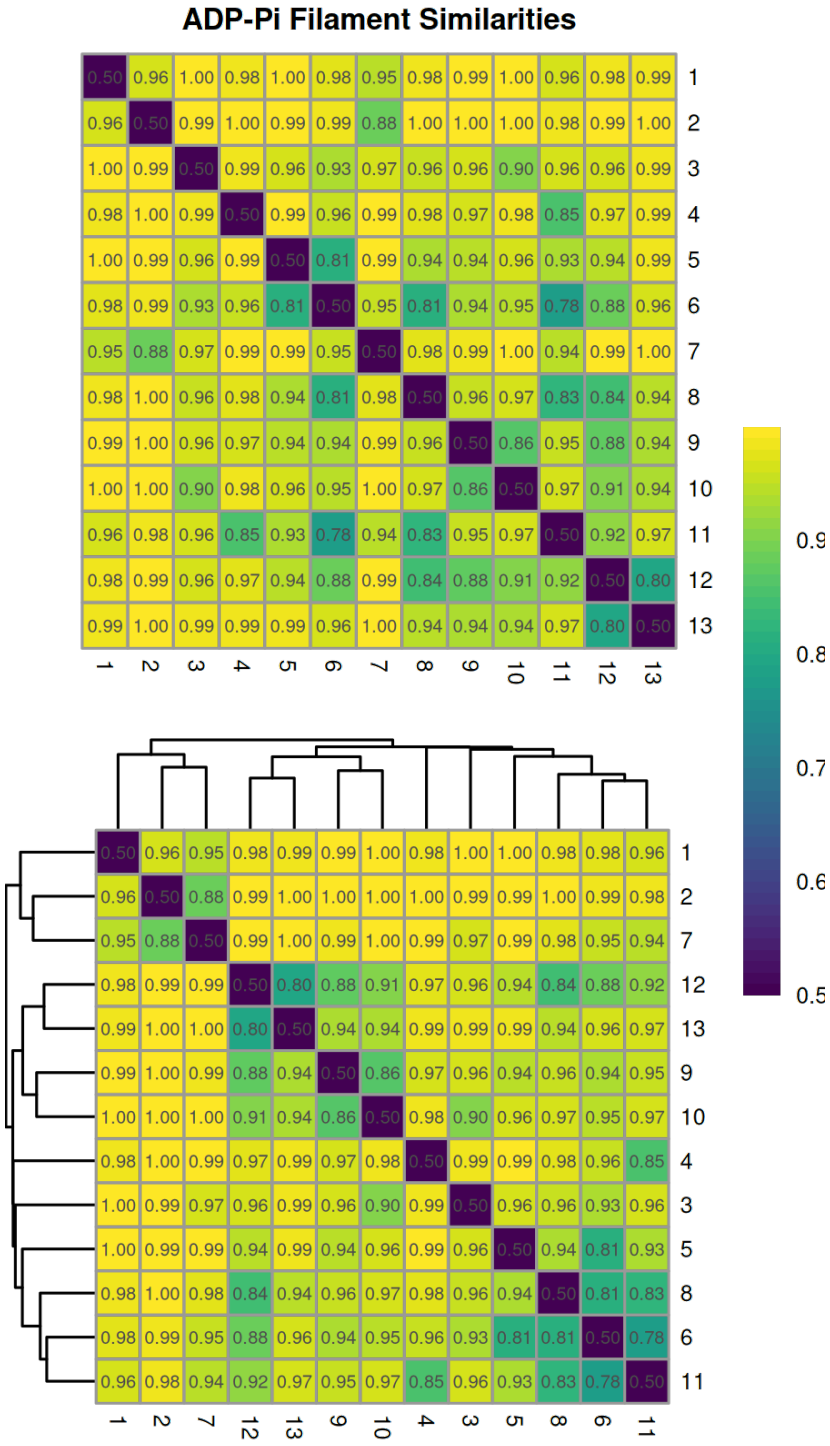

Figure 3.17: Classification based similarities of units present in the actin filament. Color, along with the number internal to each square, specifies the average accuracy of a classifier trained to distinguish configurations belonging to the index specified on the horizontal and vertical axis. The top heatmap is organized by the index unit in the filament relative to the barbed end; the lower graph is organized via complete hierarchical clustering. Note that 50% accuracy corresponds to identical ensembles, as where 100% accuracy corresponds to ensembles which are effectively distinct.

CHAPTER 4

CREATING NEW COARSE-GRAINED MODELS

4.1 MOTIVATION

As discussed at length in chapter 3, many coarse-grained (CG) models are approximate. The ways in which various CG parameterization algorithms adapt to this situation varies[69]. Simultaneously, generative adversarial networks[82] (GANs) have proven to be incredibly accurate at generating samples such as images. The quantification of error suggested in chapter 3 is closely related the training process used when parameterizing GANs. Together, this suggests the following question: what if we parameterized CG models using our estimate of ΔW?

Doing so opens the door to many interesting connections. First, the measures of error used by various types of GANs[1] are closely linked to existing approaches for creating CG models. Second, the approaches presented allow for one to control the resolution at which models are parameterized. While seemingly esoteric, this implies that the "CG" model can adopt any representation that can be mapped to the primary CG resolution; this, in turn, enables the systematic optimization of CG models with general virtual sites, for which we provide two numerical examples. Virtual sites are CG sites which have complex or unknown relationships with the primary CG resolution and have recently showed value in references [98, 162]. Sometimes no relationship is present; other times, the relationship is nonlinear. These sites are primarily useful as scaffolds for particular types of force-fields. The multi-resolution analysis performed additionally provides a novel parsimonious

[1] As of September 2018, there were at least 500 named variants of GANs[161], and this number has continued to grow.

algorithm unrelated to GANs to use when optimizing systems with virtual particles. The work presented here is primarily drawn from Durumeric & Voth [67].

4.2 INTRODUCTION

As discussed in chapter 1, CG molecular dynamics (MD) can be effective for studying systems where the motions of nearby atoms are highly interdependent[11, 30, 33, 37, 163]. By simulating at the resolution of CG sites or "beads", each associated with multiple correlated atoms, biomolecular processes at the second timescale and beyond can be accurately probed. High-fidelity CG MD models often depend on flexible parameterizations; as a result, the design of systematic parameterization strategies is an active area of study (e.g., methods and applications in references [38, 40–65]). As alluded to in section 2.1, there are two nonexclusive classes of parameterization strategies for CG models of interest to this chapter: top-down and bottom-up approaches[11, 30, 33]. Top-down approaches aim to parameterize CG models to recapitulate specific macroscopic properties, such as pressure and partition coefficients,[2] while bottom-up methods attempt to parameterize CG models to reproduce the multidimensional distribution given by explicitly mapping each atomistic configuration (produced by a suitable reference simulation) to a specific CG configuration[43–47]. The distribution of this mapped system is produced via a Boltzmann distribution with respect to an effective CG Hamiltonian referred to as the manybody potential of mean force (manybody-PMF).

Certain scientific inferences could be informally drawn from the fit CG force-field itself, assuming that the force-field is constrained to interpretable low dimensional contributions (e.g., pairwise forces, such as in reference [164]); this is akin to the parameter inspection of our stove model discussed in chapter 1. For example, one could attempt to infer the effect of an amino acid mutation on protein behavior by considering how the approximated manybody-PMF differs when fit on reference wild type and mutant proteins simulations, similar to the analysis of low dimensional free energy surfaces. However, the primary use of CG models is typically based on their ability

[2] We note that recent work[72] has reinforced how macroscopic observable matching has unexpected challenges when trying to establish a firm microscopic connection to atomistic systems if care is not taken in choosing which observable form to optimize with.

to generate CG configurations of a system of interest using their approximate force-field[51, 58, 165, 166]. The computational similarity of CG models with their atomistic counterparts allows CG models to be simulated using the same high performance software packages as those used in atomistic simulation[19, 167–172]. As a result, the computational profile of CG models is often controlled by the same dominating factor as atomistic models: the calculation of the force-field at each timestep[19, 23, 173]. This cost similarly provides motivation for specific low dimensional force-field contributions. However, there is no guarantee that a force-field characterized solely by traditional bonded and pairwise nonbonded terms either describes the true manybody-PMF of the CG variables or can accurately reproduce all observables of interest to the parameterization[11, 30, 33]. In the case of bottom-up methods, while typical approaches will produce the manybody-PMF in the infinite sampling limit when they are capable of representing any CG force-field, in practice each method creates a characteristic approximation (e.g., reproducing two-body at the expense of higher order correlations).

The compromises invoked by various bottom-up CG methods in realistic applications are critical to the utility of the resulting models. Certain methods focus on reproducing correlations dual to the potential form used[40–42, 69, 174]; for example, when using a pairwise nonbonded potential these methods recapitulate the radial distribution function of the target system. Other particular methods are characterized by attempting to reproduce both these dual correlations along with certain higher order correlations intrinsically connected to the CG potential[43–45, 48, 69]. The nature of the distributions approximated suggests three natural approaches for improving an inaccurate model: improve the CG force-field basis used, modify the CG representation, or select a different procedure to generate the CG force-field. The first two options are often a central part of the design of a systematic CG model; however, realistic systems, such as proteins, may not be well described by correlations that are typically connected to computationally efficient CG potentials coupled with appropriate CG representations[33]. More generally, the specific correlations critical to a reasonably accurate CG model may depend on the study at hand, and may be representable by simple force-fields—but only at the expense of other correlations connected to that potential form as dictated

by a particular bottom-up method. As a result, the diversity of possible applications motivates the creation of additional strategies for bottom-up CG modeling, each of which has different biases in the approximations it produces.

The task of generating examples (such as images) similar to a known empirical sample is of significant interest to the Machine Learning (ML) community[83, 175–177]. The creation of an artificial process that can produce realistic samples often entails encoding an understanding of the true mechanism underlying the real world distribution; internal representations of an accurately parameterized generative model, such as neural network parameters, can be transferred for use in secondary tasks such as classification[178] or image retrieval[179]. The artificial samples produced by the models themselves have additionally shown value by providing novel molecular targets for synthesis [180, 181] or as labeled images for training in classification or regression[182, 183]. A substantial number of these complex applications utilize implicit generative models [82, 83, 184, 185]. Implicit generative models, such as Generative Adversarial Networks (GANs)[82], are characterized by their lack of an explicit probability distribution, or an associated Hamiltonian, at the resolution they produce examples[83]. For example, a GAN may be trained to generate pictures containing human faces[82]. Each picture that could be generated has a parameterization specific probability of being a reasonable picture of a human face (admittedly, this probability is often very close to one or zero); however, the GAN itself does not have explicit knowledge of this probability. Instead, the GAN is simply characterized as a procedure that transforms random numbers from a simple noise distribution to images that follow the probability distribution of plausible pictures. The methods used to train GANs therefore focus on the ability to critique a model distribution against reference samples without knowledge of the probability density function characterizing the model. This is in strong contrast to typical molecular simulation[7, 10, 99], which traditionally requires a known Hamiltonian to produce samples through MD or Markov Chain Monte Carlo—and whose systematic parameterization techniques often naturally explicitly involve evaluation of the corresponding model Hamiltonian surface [38, 40, 43–48, 50–62, 64]. However, both methods are focused on accurately producing samples, or configurations, as their primary goal.

This chapter focuses on making this intuitive connection between GANs and MD models explicit, allowing us to apply established insight from the adversarial community to bottom-up CG modeling, giving rise to new strategies for CG parameterization we term Adversarial-Residual-Coarse-Graining (ARCG). By doing so we facilitate the use of additional classes of CG model quality measures that may show promise in modifying the approximations characterizing the optimal CG model when using a constrained set of candidate potentials to represent the CG force-field. We additionally find that it is possible to decouple the resolution at which one critiques the behavior of the CG model and the resolution at which a CG force-field is required: as an example we describe a novel rigorous avenue to increase the expressiveness of bottom-up CG models through the use of virtual sites. Critically, we do not utilize a full GAN architecture to generate CG samples; rather, we utilize the supporting theory [81, 82, 186–190] to optimize traditional CG force-fields. Section 4.3 provides both an informal and a formal summary of the theoretical underpinnings, while section 4.4 provides details on a particular instance of ARCG and a computational implementation. Section 4.5 then provides results on three simple toy systems and three molecular systems, and section 4.6 outlines the consequences of the results and possible future studies. Section 4.7 provides concluding remarks.

4.3 THEORY

The purpose of this section is to both informally describe and formally define ARCG, and to summarize connections between ARCG and previous CG parameterization methods. We begin by presenting an intuitive understanding of a specific form of ARCG to provide clarity for the subsequent mathematical description. We then follow by defining notation and the fine-grained/CG systems to which ARCG applies. We define ARCG and describe its estimation and optimization. We then move to decouple the resolution at which one critiques the CG model from the resolution native to the CG Hamiltonian, thereby generalizing our application to systems containing virtual CG sites. We continue by discussing the corresponding challenges with momentum consistency, and we finish by summarizing ARCG's relationship to previous CG methods.

4.3.1 INFORMAL DESCRIPTION OF ARCG

Bottom-up CG models are parameterized to approximate the free energy surface implied by mapping fine-grained (FG) configurations to the CG resolution[11, 33], similar to the analysis in chapter 3. Generally, this entails considering many different possible CG models (each, for example, characterized by a different pair potential) and their relationship to the reference FG data. Often, this is operationalized by creating a variational statement and searching for the CG model that minimizes it (for example, minimizing the relative entropy between the CG model and FG data[47]). After such a procedure is complete the modeler is well advised to visually inspect and compare the configurations produced by the selected CG model to those produced by the reference FG model. If the configurations are dissimilar, then the CG model is likely not adequate, and aspects of the variational statement or set of initial models considered must be modified and the parameterization process repeated.

It is natural to ask whether the final inspection of configurations produced by the FG and CG models can be intrinsically linked to the variational statement parameterizing the CG model. It is intuitive that for systematic CG parameterization methods derived from configurational consistency [38, 40, 43–48, 50–55] that when an indefinite amount of samples are used and all possible CG models are considered that the optimized CG model will perfectly reproduce the mapped FG statistics, and as a result, the configurations produced by the FG and CG models will be indistinguishable. However, in cases where perfectly reproducing the FG statistics is infeasible it seems natural to ask if a model could be trained using this criteria of distinguishability directly.

While it could be possible in simple situations to use a human observer to intuitively rank CG models by considering the configurations they produce, this procedure quickly becomes subjective and untenable for complex models; this, after all, was the problem analyzed by chapter 3. Following the analysis of chapter 3, the implied procedure for CG parameterization is to optimize the CG model such that it is intrinsically difficult to perform classification: as a result, the computer will inevitably make many mistakes on average when attempting to isolate configurations characteristic to only the FG and CG data. One possible intuitive manifestation of ARCG theory concretely

implements this classification procedure while maintaining clear connection to CG methods such as relative entropy minimization (REM)[47]. Previous CG parameterization methods have used similar, but not identical, motivations to produce parameterization strategies[47, 55, 59]. ARCG theory serves to connect, clarify, and reframe these methods where possible while extending beyond the classification metaphor.

We reiterate that the task of classification is a variational procedure itself[81, 191]: the ideal estimate of the true sources of a set of molecular configurations has a lower error than all other estimates. The optimization in classification searches over these various possible hypotheses. As a result, at each step of force-field optimization ARCG must perform this variational search over possible hypotheses, resulting in two nested variational statements in the full model optimization procedure: one required for classification, and the other for choosing the resulting CG model. Importantly, the error rate of the optimal classifier can be explicitly linked to various f-divergences (e.g., relative entropy) evaluated between the mapped FG and CG distributions[81]. This suggests an equivalent formalism with which to view ARCG: the variational estimation of divergences. This alternate interpretation additionally illustrates how additional divergences, such as the Wasserstein distance[190], can be estimated, even without a clear connection to classification. As a result, ARCG theory is primarily treated through the lens of variational divergence estimation in the following sections.

The variational estimation intrinsic to ARCG affords an interesting extension to traditional parameterization strategies: the resolution at which the CG Hamiltonian acts may be finer than the resolution at which the model is compared to the reference data. Equivalently, CG samples can be mapped before being compared to the mapped reference FG samples. For example, additional particles may be introduced to facilitate complex effective interactions between the CG particles, and then may be mapped out before comparing to the mapped reference FG samples. Applying such a mapping creates issues with many other parameterization strategies as discussed in section 4.3.6.

4.3.2 ADVERSARIAL-RESIDUAL-COARSE-GRAINED MODELS

ARCG models are characterized by a set of possible $\mathcal{F}$ that are defined variationally as the difference in ensemble averages of a pair of coupled scalar functions. The functions, [3] f and g, are found as producing the maximum of the following variational definition

$$\mathcal{F}[p_{\text{mod},\mathbf{R},\theta}, p_{\text{ref},\mathbf{R}}] := \max_{(f,g)\in\mathcal{Q}} \left\{ \langle f \rangle_{p_{\text{mod},\mathbf{R},\theta}} - \langle g \rangle_{p_{\text{ref},\mathbf{R}}} \right\}, \tag{4.1}$$

leading to a minimax variational statement for the fit model itself

$$\theta^{\dagger} = \operatorname*{argmin}_{\theta} \left[\max_{(f,g)\in\mathcal{Q}} \left\{ \langle f \rangle_{p_{\text{mod},\mathbf{R},\theta}} - \langle g \rangle_{p_{\text{ref},\mathbf{R}}} \right\} \right]. \tag{4.2}$$

In other words, for a specific choice of p_{mod} and p_{ref} the numerical value of our residual is determined by a specific (f,g) pair; all other choices of pairs of observables in $\mathcal{Q}$ produce a more optimistic estimate of the quality of our model. These observables are evaluated via their configurational average at the CG resolution. As we update θ, the optimal choice of (f,g) will change.

The definition of $\mathcal{Q}$ depends on the particular $\mathcal{F}$ specified. For example, if $f = g$ for all pairs in $\mathcal{Q}$, this expression defines the class of integral probability metrics (IPM) or Maximum Mean Discrepancy (MMD) distances[195, 196], with each MMD distance or IPM then being defined by further constraints on $\mathcal{Q}$. Typically, the function space in MMD is restricted to the unit ball in a reproducing kernel Hilbert space, a choice which allows the maximization to be resolved via a closed expression. The examples in this document will estimate f-divergences: in this case,

$$\mathcal{Q} := \left\{ \left(-\frac{1}{2} l_{\text{mod}} \circ \hat{\eta}, \frac{1}{2} l_{\text{ref}} \circ \hat{\eta} \right) : \hat{\eta} \in [0,1]^{\chi_{\mathbf{R}}} \right\} \tag{4.3}$$

[3] These functions, as well as other functions throughout the document, must be integrable with respect to the measures defining the model or reference distributions. We do not assume such functions are continuous unless noted. We refer the reader to Reid & Williamson [81], Bottou *et al.* [189], L'Ecuyer [192], Kleijnen & Rubinstein [193], and Milgrom & Segal [194] for more information on necessary constraints; these constraints do not affect the proposed implementation.

where we have used $\circ$ to denote function composition, e.g., $f \circ g(x) := f(g(x))$, $[0,1]^{\mathcal{X}_{\mathbf{R}}}$ denotes the set of functions from $\mathcal{X}_{\mathbf{R}}$ to $[0,1]$, and l_{ref} and l_{mod} are functions determined by the specific f-divergence estimated and whose closed form is given in the next section.

Force-field selection requires an optimization over θ to satisfy the external minimization in Eq. (4.2). The strategies available for doing so depend on the structure of Q. For low dimensional parameterizations it may be feasible to do a grid search over possible models and to use Eq. (4.1) to select the ideal model. However, for higher dimensional parameter spaces an attractive option is to use methods utilizing the gradient with respect to θ. If the maximized estimate over Q is differentiable at a particular point with respect to θ, then (due to the envelope theorem[194], see appendix C) the derivatives with respect to θ at that point only include terms related to the ensemble average over the model distribution, $p_{\text{mod},\mathbf{R}}$

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathcal{F}\left[p_{\text{mod},\mathbf{R},\theta}, p_{\text{ref},\mathbf{R}}\right] &= \frac{\mathrm{d}}{\mathrm{d}\theta_i}\max_{(f,g)\in Q}\left\{\langle f\rangle_{p_{\text{mod},\mathbf{R},\theta}} - \langle g\rangle_{p_{\text{ref},\mathbf{R}}}\right\} && (4.4)\\ &= \frac{\partial}{\partial\theta_i}\langle f^\dagger\rangle_{p_{\text{mod},\mathbf{R},\theta}} && (4.5)\end{aligned}$$

where $f^\dagger$ represents one of the optimal observables found at the internal maximum. When the maximized inner estimate is expressible in closed form (which is true in the case of the f-divergences estimated in this document), we can directly confirm the existence of this derivative. Assuming that the integral and derivative operators may be exchanged, simple substitution provides a covariance expression for estimation:

$$\frac{\partial}{\partial\theta_i}\left\langle f^\dagger\right\rangle_{p_{\text{mod},\mathbf{R},\theta}} = \beta\left\langle f^\dagger\right\rangle_{p_{\text{mod},\mathbf{R},\theta}}\left\langle\frac{\partial U_\theta}{\partial\theta_i}\right\rangle_{p_{\text{mod},\mathbf{R},\theta}} - \beta\left\langle f^\dagger\frac{\partial U_\theta}{\partial\theta_i}\right\rangle_{p_{\text{mod},\mathbf{R},\theta}}. \quad (4.6)$$

These results suggest a straightforward numerical optimization of Eq. (4.2) using sample based averages and gradient descent (and related gradient based methods, e.g., RMSprop[140]). We typically represent Q by indexing with a finite dimensional vector ψ. At each iteration of optimization, holding θ constant, we maximize over ψ using samples from the model and reference distributions

to estimate our expected values; then, holding ψ constant, we take a single step on the gradient of θ estimated by the sample average of the covariance expression. This two step process is iterated until convergence of θ.

Not all definitions of Q produce meaningful procedures for creating CG models. Generally, particular forms of $\mathcal{F}$ are derived individually, each of which is amenable to the procedures outlined here. We continue by investigating an informative subset of possible $\mathcal{F}$, characterized via f-divergences, that will provide functionality directly encompassing REM CG[47] and the approach of Vlcek & Chialvo [55], and is additionally related to the approach of Stillinger [38].

4.3.3 F-DIVERGENCES

The f-divergences are a category of functions characterizing the difference between two distributions[81, 197, 198]. When probability density functions are available we can express this family of divergences as

$$\mathbb{I}_f(p_{\text{ref},\mathbf{R}}, p_{\text{mod},\mathbf{R}}) := \int_{\mathcal{X}_\mathbf{R}} p_{\text{mod},\mathbf{R}}(x) f\left(\frac{p_{\text{ref},\mathbf{R}}(x)}{p_{\text{mod},\mathbf{R}}(x)}\right) \mathrm{d}x \tag{4.7}$$

where each member of the family is indexed by a convex function f that satisfies $f(1) = 0$. Relative entropy, the divergence central to REM CG, can be obtained by defining $f(x) := x \log x$,[4] and the Hellinger distance, central to previous methods by Stillinger [38] and Vlcek & Chialvo [55] can be obtained by via $f(x) := (\sqrt{x} - 1)^2$.

The f-divergence between p_{mod} and p_{ref} can be expressed in multiple variational statements[81, 186, 187, 189]. We here utilize its duality with the training difficulty of classification which is mathematically expressed in the following formulation

$$\mathbb{I}_f(p_{\text{mod},\mathbf{R}}, p_{\text{ref},\mathbf{R}}) = \max_{\hat{\eta} \in [0,1]^{\mathcal{X}_\mathbf{R}}} \left[-\frac{1}{2} \langle l_{\text{mod}} \circ \hat{\eta} \rangle_{p_{\text{mod},\mathbf{R}}} - \frac{1}{2} \langle l_{\text{ref}} \circ \hat{\eta} \rangle_{p_{\text{ref},\mathbf{R}}} \right] \tag{4.8}$$

[4] We note that the x proceeding the log here effectively changes the distribution over which log is averaged; relative entropy traditionally averages $\log\left(\frac{p_{\text{ref}}(x)}{p_{\text{mod}}(x)}\right)$ over p_{ref}.

where

$$\begin{aligned} \underline{L}(x) &:= -2(1-x)f\left(\frac{x}{1-x}\right) \\ l_{\text{mod}}(h) &:= \underline{L}(h) - h\left.\frac{\partial \underline{L}}{\partial x}\right|_h \\ l_{\text{ref}}(h) &:= \underline{L}(h) + (1-h)\left.\frac{\partial \underline{L}}{\partial x}\right|_h . \end{aligned}$$

Note the similarity between this representation and Eq. (2.18). The function $\hat{\eta}$ is a function of a CG configuration, mapping each configuration to a real number in $[0,1]$.[5] Note that substitution into Eq. (4.2) (along with the removal of prefactors) provides us with our training residual

$$\theta^{\dagger} = \underset{\theta}{\text{argmin}} \left[\max_{\hat{\eta} \in [0,1]^{\mathcal{X}_{\mathbf{R}}}} \left\{ -\langle l_{\text{mod}} \circ \hat{\eta} \rangle_{p_{\text{mod},\mathbf{R},\theta}} - \langle l_{\text{ref}} \circ \hat{\eta} \rangle_{p_{\text{ref},\mathbf{R}}} \right\} \right] \tag{4.9}$$

where the optimal $\hat{\eta}$ producing the corresponding f-divergence, denoted η, is known to be[81]

$$\eta(x) = \frac{p_{\text{ref},\mathbf{R}}(x)}{p_{\text{mod},\mathbf{R}}(x) + p_{\text{ref},\mathbf{R}}(x)}. \tag{4.10}$$

While here we have denoted our inner variational statement as optimizing over a space of functions instead of pairs of functions, this is equivalent to Eq. (4.2) when defining Q via Eq. (4.3).

The optimization present in Eq. (4.8) considers a diverse set of functions over a high dimensional domain. However, as expanded on in Sec. 4.4, numerically tractable strategies typically restrict the set of functions considered by only including functions which are expressible through selected features of the molecular systems studied (e.g., functions of the distance matrix). When this restricted set still contains η, however, the presented equality with f-divergences still applies. We

[5] We note that certain proper losses may only be defined on $(0,1]$, $[0,1)$, or $(0,1)$. The is partially reflective of the fact that certain f-divergences, such as relative entropy, are not always defined when the support of the corresponding densities is not the same. There exist cases such that the limiting behavior of such losses is still valid for distributions with differing support, such as the log loss. The optimization performed in this document only compares distributions for which the KL divergence is defined and for which $\eta \in (0,1)$. The variational statements hold more generally; see Reid & Williamson [81] for more details.

introduce the following term to describe such suitably expressive Q: in the context of f-divergences, when Q contains $(-\frac{1}{2}l_{\text{mod}} \circ \eta, \frac{1}{2}l_{\text{ref}} \circ \eta)$, i.e. when optimization with the population averages would return the corresponding f-divergence, we will refer to Q as being complete. Additionally, when Q is expressive enough such that it is complete for each step in an optimization process, we will additionally refer to it as complete, with the distinction evident from context. For example, if for any two configurational distributions produced by Lennard-Jones potentials Q contained η, then Q would be complete when optimizing a model Lennard-Jones system to fit data produced by a reference Lennard-Jones system with initially differing parameters. The implications of optimizing via an incomplete Q are discussed in section 4.4.

As described in section 2.2, Eq. (4.9) has a notable intuitive description, which will be useful when considering implementation and connections to similar methods. Consider an external observer that has access to a mixture of molecular configurational samples, some of which are produced by our mapped reference simulation and others from our CG model (termed our reference and model samples, respectively). The observer is faced with the following task: they must distinguish which examples came from which source based solely on configurational details. We represent the observer's guess by the function $\hat{\eta}$, which maps each molecular configuration to a number in the interval $[0,1]$. We associate the label 0 with configurations from our model and the label 1 with configurations from our reference set (note that the labels are discrete, but our estimate is a real number between 0 and 1 inclusive). We decide in this case to use the square loss, giving us the following definitions for our loss functions:

$$l^{\text{sq}}_{\text{mod}} \circ \hat{\eta}(x) := \hat{\eta}(x)^2 \tag{4.11}$$

$$l^{\text{sq}}_{\text{ref}} \circ \hat{\eta}(x) := (1 - \hat{\eta}(x))^2 \tag{4.12}$$

where x is a particular molecular configuration. For example, if the observer guesses a probability of 0.68 for a configuration that was drawn from the reference set, they are penalized $(1-0.68)^2 = 0.1024$. If the configuration instead came from the model data set, they are penalized $0.68^2 = 0.4624$.

The observer wishes to minimize their penalty, and if they are able to guess 1 for all configurations drawn from the reference set and 0 for all the configurations drawn from the model set, then their loss will be minimized at 0. This is the task of classification described from a human perspective.

If the model is very poor, achieving a average loss of 0 will be easy—the configurations from the model will be distinct from the reference configurations. However, for higher quality models many of their configurations will plausibly come from either the model or the reference simulation. Even with the perfect $\hat{\eta}$, a configuration which has a 50% probability of coming from the reference and model sets will entail a minimum loss of 0.25 (this minimum is entailed when the estimated probability is also 50%); this loss cannot be reduced further. We refer to this loss as the irreducible loss. This is analogous to the least squares residual present in linear regression with Gaussian noise. The ideal line minimizes the least squares residual, but the least squares residual is nonzero as the line cannot perfectly fit the data.

This inability to perfectly distinguish samples is directly related to ΔW in chapter 3 and our f-divergences (e.g., relative entropy)[81]. Modifying the manner in which we penalize incorrect predictions (via l_{mod} and l_{ref}) specifies which divergence is produced. In this example, we have decided on the form of our losses directly; when estimating a particular f-divergence the expressions defining the losses are given by Eq. (4.8). This loss function is asymmetric depending on the true origin of the sample: l_{mod} penalizes a prediction on a sample gained from the model, while l_{ref} penalizes a prediction on a reference sample. Notably, while there are constraints on what functions l_{mod} and l_{ref} can be defined as in order for η (the optimal $\hat{\eta}$) to obey Eq. (4.10), these constraints are already taken into account by Eq. (4.8): a valid f-divergence will always yield losses whose optimal estimate is given by Eq. (4.10).

As a result, we simply need to train a classifier with a loss on our samples and consider the average loss implied by its probabilistic predictions. An extended formal description of this task and the corresponding duality is presented in Reid & Williamson [81]. This interpretation is central to the term adversary in the name of Generative Adversarial Networks[82]: the adversary attempts identify the source of each sample and we wish to make its task as difficult as possible.

4.3.4 VIRTUAL SITES

The ARCG framework can be lightly generalized to decouple the resolution at which the CG potential acts and the resolution at which we compare our CG and reference systems. More specifically, we see that we can apply a distinct mapping operator to our CG system before it is compared to the mapped FG samples. To better illustrate the practical use of this extension we begin by providing a motivating example.

As previously discussed, many bottom-up CG methods are shown to produce the ideal manybody-PMF when they are allowed to adopt any force-field in the ideal sampling limit. However, CG models are often limited to molecular mechanics type potentials (e.g., pairwise nonbonded potentials), which often do not contain the ideal manybody-PMF as a possible parameterization. For example, one might use Multiscale Coarse-Graining[43–46] (MS-CG) to parameterize a CG lipid bilayer in which all of the solvent and some of the lipid degrees of freedom have been removed. Upon generating samples using the CG model we may find that certain properties of the membrane, such as its thermodynamic force of bilayer assembly, are poor. However, the MS-CG method has likely provided one with its correct characteristic approximation; in order to improve the model with the same parameterization method one must either increase the complexity of the CG force-field via higher order terms or change the FG details retained via modification of $\mathcal{M}$, the CG mapping operator seen in Eqs. (2.4) and (2.5). Here, we discuss a third option: augmenting the CG representation directly without modifying $\mathcal{M}$. As a simple example consider modeling the interaction of two benzene molecules using a CG pairwise potential. Here, the CG representation is given by three sites per benzene ring. It may be difficult to capture the π-stacking effect using this type of potential at this CG resolution. As a remedy one could add particles normal to the plane containing the benzene molecule, as shown in Fig. 4.1, without associating these additional CG sites to FG sites via $\mathcal{M}$. Importantly, however, we will only critique the behavior of our CG model after these virtual sites have been integrated out: the CG model is optimized to minimize the relative entropy between the mapped FG reference and CG model after the integration over the possible positions of these virtual CG sites.

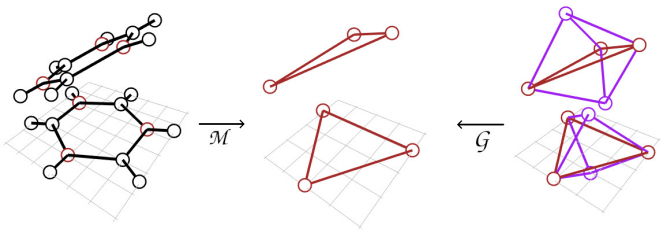

Figure 4.1: An example of virtual particle usage. The atomistic representation of benzene (left) is mapped via $\mathcal{M}$ to a CG representation (center) only preserving three carbons (red). The full CG representation (right) of the same configuration has these three carbons and two additional virtual sites (purple) to help a pairwise potential capture the correct manybody-PMF. These sites are removed upon application of the virtual particle map $\mathcal{G}$. These virtual sites have no atomistic counterpart.

Description of the formalism encompassing these situations requires us to suitably expand our notation. We still consider all distributions described previously but use the following modifications: first, samples from p_{mod} are no longer generated by a simulation using the approximated manybody-PMF as its Hamiltonian. Instead, these samples are produced via a new mapping operator $\mathcal{G}$ and simulation of a new finer grained representation characterized by $p^{\text{pre}}_{\text{mod}}$ via its own Hamiltonian $\left(\sum_{i=1}^{3\nu} p_i^2/2m_i + U^{\text{pre}}_{\text{mod}}(r^{3\nu})\right)$ where m_i are the masses at the pre-CG resolution. As a result, p_{mod} is redefined with the following relations.

$$p_{\text{mod},\mathbf{R}}(\mathbf{R}^{3N}) := \int_{\mathcal{X}_r^{\text{pre}}} p^{\text{pre}}_{\text{mod},r}(r^{3\nu})\delta(\mathcal{G}_r(r^{3\nu}) - \mathbf{R}^{3N})\mathrm{d}r^{3\nu} \tag{4.13}$$

$$p_{\text{mod},\mathbf{P}}(\mathbf{P}^{3N}) := \int_{\mathcal{X}_p^{\text{pre}}} p^{\text{pre}}_{\text{mod},p}(p^{3\nu})\delta(\mathcal{G}_p(p^{3\nu}) - \mathbf{P}^{3N})\mathrm{d}p^{3\nu} \tag{4.14}$$

The resulting relations between resolutions are summarized in Fig. 4.2.

Importantly, our training procedure needs two minor modifications. First, the variational estimation of divergences presented in Eq. (4.1) is composed solely of ensemble averages, which are approximated via sample averages; these averages can be evaluated by generating empirical samples from p_{mod} via samples drawn from $p^{\text{pre}}_{\text{mod}}$ and application of $\mathcal{G}$. This is a consequence of Eq. (4.15).

$$\langle f \rangle_{p_{\text{mod}}} = \langle f \circ \mathcal{G} \rangle_{p^{\text{pre}}_{\text{mod}}} \tag{4.15}$$

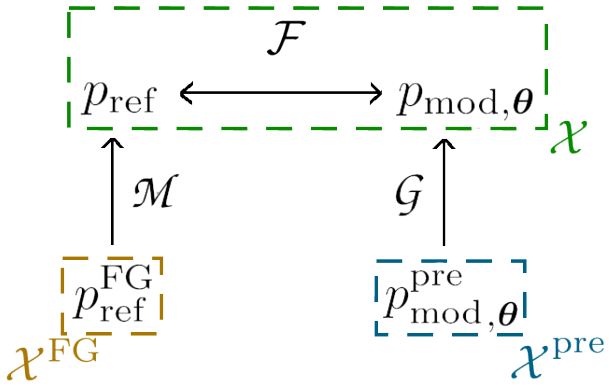

Figure 4.2: The relationship between resolutions when comparing FG and CG systems at a custom resolution, such as the case of virtual sites. Samples from the pre-CG domain $\mathcal{X}^{\mathrm{pre}}$ (e.g., a CG configuration including virtual sites) are mapped to the CG domain $\mathcal{X}$ (e.g., a CG configuration without virtual sites) via $\mathcal{G}$; samples from the FG domain $\mathcal{X}^{\mathrm{FG}}$ (e.g., atomistic) are mapped to the same CG domain $\mathcal{X}$ via $\mathcal{M}$. The mapped samples are then compared via $\mathcal{F}$.

Second, the gradients required for optimization of the parameters of the variational search (θ) are calculable again through Eq. (4.15), allowing us to utilize our previous expression Eq. (4.6) at the resolution native to our new pre-CG Hamiltonian by minimizing the variationally optimized observable composed with $\mathcal{G}$.

Importantly, while our examples in this section have primarily concerned situations in which fictional particles are added to the CG representation and then completely integrated over before calculating divergences, $\mathcal{G}$ can easily be generalized. Fundamentally, it has the full flexibility of $\mathcal{M}$; similarly, additional constraints are born from maintaining momentum consistency via methods described in the next subsection. However, if one discards momentum consistency, it is possible to maintain an intuitive pre-CG representation while nonlinearly modifying $\mathcal{M}$ and $\mathcal{G}$ to represent custom high-dimensional observables. In this case these mapped distributions are used for determining the quality of the pre-CG model. We reserve the bulk of our discussion and investigation of these more complex options to future work.

4.3.5 MOMENTUM CONSISTENCY

Previous sections have discussed the configurational variational statement central to ARCG; here, we discuss how to ensure momentum consistency. In the case that no pre-CG resolution is considered, momentum consistency in ARCG may be achieved through identical methods as stated in previous approaches, such as MS-CG[45]. However, when considering three distinct resolutions momentum consistency takes on a slightly modified form. We provide suitable constraints for a common case below, although extensions are straightforward.

Momentum consistency is characterized by the following equation:

$$p_{\mathrm{ref},\mathbf{P}}(\mathbf{P}^{3N}) = p_{\mathrm{mod},\mathbf{P}}(\mathbf{P}^{3N}). \tag{4.16}$$

We here consider the specific case where both $\mathcal{M}_{\mathbf{r}}$ and $\mathcal{G}_r$ are linear functions that satisfy the constraints defined in the MS-CG work[45]: $\mathcal{G}_r$ is limited to associate each CG site in $\mathcal{X}^{\mathrm{pre}}$ unambiguously to at most a single site in $\mathcal{X}$ and has imposed translational and positivity constraints, and analogous constraints are placed on $\mathcal{M}_{\mathbf{r}}$ (see appendix for more details). The momentum map $\mathcal{M}_{\mathbf{p}}$ (and $\mathcal{G}_p$ with appropriate modifications) is assumed to take the following form as in reference [45]:

$$\mathcal{M}_{\mathbf{p}I}(\mathbf{p}^{3n}) := M_I^{\mathcal{M}} \sum_{i \in I_I^{\mathcal{M}}} \frac{{c_{Ii}^{\mathcal{M}}}^2 \mathbf{p}_i}{m_i}, \tag{4.17}$$

In this case, previous work [45] has shown that the constants defining $\mathcal{M}_{\mathbf{p}}$ (and similarly $\mathcal{G}_p$) can be combined with the masses of the sites contributing to a mapped site to provide a definition of the mapped masses (Eq. (4.18)) that define a Boltzmann distribution equal to the mapped momentum distribution

$$\left(M_I^{\mathcal{M}}\right)^{-1} := \sum_{i \in I_I^{\mathcal{M}}} \frac{{c_{Ii}^{\mathcal{M}}}^2}{m_i}, \tag{4.18}$$

where $M_I^{\mathcal{M}}$ is the mass of CG particle I as implied by map $\mathcal{M}$, $I_I^{\mathcal{M}}$ is the set of all atoms that map to CG site I according to map $\mathcal{M}$, and $c_{Ii}^{\mathcal{M}}$ is the coefficient describing how the positions

of FG particle i contribute to CG particle I according to map $\mathcal{M}$. More generally, this implies that we can explicitly characterize the mapped momentum distributions for both the mapped FG and mapped CG systems, which when combined with Eq. (4.16) provides the following relation implying momentum consistency in a system with virtual particles

$$\exp\left(-\beta\sum_{I=1}^{N}\frac{\mathbf{P}_I^2}{2M_I^{\mathcal{G}}}\right) \propto \exp\left(-\beta\sum_{I=1}^{N}\frac{\mathbf{P}_I^2}{2M_I^{\mathcal{M}}}\right) \tag{4.19}$$

$$\left(M_I^{\mathcal{G}}\right)^{-1} := \sum_{i\in I_I^{\mathcal{G}}}\frac{{c_{Ii}^{\mathcal{G}}}^2}{m_i}. \tag{4.20}$$

The only solution to this equation is to set $M_I^{\mathcal{G}} = M_I^{\mathcal{M}}$ for each CG site I; in this case we find a set of equations implying consistency (Eq. (4.21)).

$$\left[0 = \sum_{i\in I_I^{\mathcal{M}}}\frac{{c_{Ii}^{\mathcal{M}}}^2}{m_i} - \sum_{i\in I_I^{\mathcal{G}}}\frac{{c_{Ii}^{\mathcal{G}}}^2}{m_i}\right] \forall \text{ CG sites } I \tag{4.21}$$

Note that these equations are positively constrained with respect to masses and mapping constants (along with the previously stated constraints). This provides a simple condition connecting our FG masses, pre-CG masses, $\mathcal{M}$, and $\mathcal{G}$, and allows one to check for momentum consistency if all the relevant variables are defined. It is important to note that I indexes the CG sites at the resolution of p_{ref} and p_{mod}—that is, without the virtual particles. As such, in the case of $\mathcal{G}$ simply dropping virtual particles consistency is trivially satisfied by simply matching the masses of the non dropped particles to those implied by the FG system with $\mathcal{M}$. Additional details may be found in the appendix.

4.3.6 RELATED METHODS

Despite differences in representation, ARCG can be formulated to elucidate connections to a variety of previous CG parameterization strategies, some of which have been mentioned in previous sections. This is performed via the appropriate design of the characteristic function space $\mathcal{Q}$ in Eq.

(4.1). Additionally, ARCG bears resemblance to a recent CG method based on distinguishability and classification[59]. In this section we make explicit connections between the f-divergence implementation presented in this chapter and such external methods. The applications of the f-divergence duality presented here are in the infinite sampling limit, with a complete Q, and with converged optimization; in practice, significant differences in seemingly equivalent methods may arise.

Classification has been recently used to train a CG model by using the resulting decision function $\hat{\eta}$ to directly update the CG configurational free energy[59]. This is motivated by noticing that the η that satisfies the variational bound in Eq. (4.8) can be related to the pointwise free energy difference as described in Eq. (3.2), suggesting a procedure where ΔW is scaled and used as an additive update to the CG potential. This procedure is similarly valid using any of the f-divergence losses discussed in this chapter[81]. However, beyond the differing update rules, the variational divergence approach presented in this chapter is differentiated by a subtle but important difference in characteristic assumptions. The divergence interpretations of ARCG rely on the completeness of Q, but place no constraint on $\{p_{\text{mod},\theta}\}_{\theta\in\mathcal{T}}$, where $\mathcal{T}$ denotes the set of all model parameterizations considered. In contrast, the interpretation of the method of Lemke & Peter [59] also requires an fully expressive Q; however, as the update to p_{mod} inherently utilizes members of Q, the method naturally also forces $\{p_{\text{mod},\theta}\}_{\theta\in\mathcal{T}}$ to be fully expressive, i.e. $p_{\text{ref}} \in \{p_{\text{mod},\theta}\}_{\theta\in\mathcal{T}}$. In other words, Q and $\{p_{\text{mod},\theta}\}_{\theta\in\mathcal{T}}$ are directly coupled. As a result, in the case that the classifier used in the additive update method similarly has a relation to a specific f-divergence, an ideal model would always be chosen, rendering the specific choice of f-divergence inconsequential. Beyond this it is unclear how to expand the update rule of Lemke & Peter [59] to apply to virtual sites as the classifier is only directly present at the resolution of p_{mod} and extension of the update to the resolution of $p^{\text{pre}}_{\text{mod}}$ is unclear.

REM CG proposes that approximate CG models should be parameterized by minimizing the relative entropy,[47] or KL-divergence, between the distributions produced at the FG resolution:

$$\int_{\chi^{\text{FG}}_{\mathbf{r}}} p^{\text{FG}}_{\text{ref},\mathbf{r}}(x) \log\left(\frac{p^{\text{FG}}_{\text{ref},\mathbf{r}}(x)}{p^{\text{FG}}_{\text{mod},\mathbf{r}}(x)}\right) \mathrm{d}x \tag{4.22}$$

where we have introduced a new quantity, $p^{\mathrm{FG}}_{\mathrm{mod},\mathbf{r}}$, defined to be the probability density implied by the CG model over FG space (which is not used in ARCG theory); the exact form implied over the FG space depends on the interpretation of REM CG considered[69]. This at most differs by a constant (when considering CG force-field optimization) from the relative entropy considered at resolution of the CG model, given by

$$\int_{\chi_{\mathbf{R}}} p_{\mathrm{ref},\mathbf{R}}(x) \log\left(\frac{p_{\mathrm{ref},\mathbf{R}}(x)}{p_{\mathrm{mod},\mathbf{R}}(x)}\right) \mathrm{d}x. \tag{4.23}$$

KL-divergence is an f-divergence (generated by $f(x) := x\log x$) and in the case of Eq. (4.23) can resultingly be formulated and solved for in the current framework, providing the following losses through Eq. (4.8)

$$l^{\mathrm{RE}}_{\mathrm{ref}}(h) = 2\left[\log\left(\frac{1-h}{h}\right) - 1\right] \tag{4.24}$$

$$l^{\mathrm{RE}}_{\mathrm{mod}}(h) = 2\frac{h}{1-h}. \tag{4.25}$$

We utilize this method for the computational examples presented in Sec. 4.4. We note that the full specification of REM CG considers comparing a coarser CG model to a finer FG model at the FG resolution by defining a new model density at the FG resolution, as where we have used many-to-one functions to reduce the resolution of the FG and pre-CG model in our theoretical approach. However, calculation of the relative entropy at CG resolution produces the same force-field selection rule as the FG relative entropy when considering the CG force-field. Optimization of systems with virtual particles is not straightforward via REM CG as most refinement schemes require $\langle \partial_\theta U_{\mathrm{mod}\theta} \rangle$ which is difficult to directly calculate as the explicit form of $U_{\mathrm{mod}\theta}$ is unknown in the case of virtual particles. We do, however, show that this derivative can be variationally approximated in an alternative way distinct from the classification approach central to this chapter in section 4.4.

Schöberl *et al.* [63] extended REM CG by framing coarse-graining as a generative process where the FG statistics are non-deterministically produced by the CG variables by means of a backmapping

operator, a method termed Predictive Coarse-Graining (PCG). This approach allows optimization of the backmapping operator itself and additionally allows more flexibility in describing the connection between the FG and CG systems. This allows PCG to describe CG models with virtual particles. Additionally, PCG is trained using expectation-maximization, which can be framed as a two part process with a variational search providing the information for a gradient update of the parameters. PCG differs from ARCG in multiple ways. First, PCG aims to optimize an iteratively tightened lower bound on the relative entropy of the CG model, whereas ARCG encompasses the optimization of a larger variety of possible metrics, including relative entropy. Additionally, the variational estimation in PCG is solved via a closed form expression and generates a gradient update which optimizes said lower bound, as where the variational optimization in ARCG is solved iteratively in practice and provides the exact gradient of relative entropy. Finally, ARCG is not formulated as generating statistics at the FG resolution and instead is formulated on the CG resolution. Despite these differences, the overall similarity between PCG and ARCG suggests that the two methods could be used to extend each other. We reserve a detailed analysis of these connections for a future work.

Alternatively, recent work by Vlcek & Chialvo [55] (as well as previous work by Stillinger [38]) suggests that the Bhattacharyya distance (*BD*) Eq. (4.27) is a natural metric to judge approximate models.

$$BC(p_{\mathrm{mod},\mathbf{R}}, p_{\mathrm{ref},\mathbf{R}}) := \int_{\chi_{\mathbf{R}}} \sqrt{p_{\mathrm{mod},\mathbf{R}}(x) p_{\mathrm{ref},\mathbf{R}}(x)}\mathrm{d}x \tag{4.26}$$

$$BD(p_{\mathrm{mod},\mathbf{R}}, p_{\mathrm{ref},\mathbf{R}}) := -\log BC(p_{\mathrm{mod},\mathbf{R}}, p_{\mathrm{ref},\mathbf{R}}) \tag{4.27}$$

While the Bhattacharyya distance is not an f-divergence, it is related to one via a monotonic transformation: the Hellinger distance (*H*)

$$H(p_{\mathrm{mod},\mathbf{R}}, p_{\mathrm{ref},\mathbf{R}}) := \sqrt{1 - BC(p_{\mathrm{mod},\mathbf{R}}, p_{\mathrm{ref},\mathbf{R}})} \tag{4.28}$$

$$= \mathbb{I}_{(\sqrt{t}-1)^2}(p_{\mathrm{mod},\mathbf{R}}, p_{\mathrm{ref},\mathbf{R}}). \tag{4.29}$$

This can be variationally approximated in the same framework as REM CG, resulting[81] in the following losses:

$$l^H_{\text{mod}}(h) = 2\sqrt{\frac{h}{1-h}} \tag{4.30}$$

$$l^H_{\text{ref}}(h) = 2\sqrt{\frac{1-h}{h}}. \tag{4.31}$$

Justification of the Bhattacharyya distance may be grounded in information geometry and the distinguishability of samples produced by the FG and CG models. Despite the apparent similarity to the fictional game described earlier, said justification of Vlcek & Chialvo [55] is grounded in distinguishing populations (and not individual configurations) via their collective empirical samples, while our game focuses on distinguishing individual configurations. The stated connection simply occurs through our duality with f-divergences.

Inverse Monte Carlo (IMC)[40], also known as Newton Inversion (NI), optimizes the expected value of a vector-valued observable that characterizes the difference between the mapped FG and CG systems (often through their radial distribution functions) and can be adapted for use on systems with virtual particles. The covariance derivative term presented in Eq. (4.6) may be used to create a set of equations analogous to those used in traditional IMC/NI in order to couple an arbitrary potential to a vector observable quantifying the quality of the model.[6] Importantly, observables (e.g., the radial distribution function) defined on non-virtual sites may still be calculated with virtual particles present, which allows the optimization to be performed at the pre-CG resolution using reference data defined only at the CG resolution. ARCG may be viewed similarly as optimizing the expected value of observables. However, in ARCG the observable minimized at each step is scalar, must be variationally found at each step of optimization, and the corresponding derivatives

[6] In this form the observables considered no longer must be dual to the potential form optimized. Additionally, if the number of force-field parameters does not equal the number of observables considered, the linear system typically used in IMC/NI optimization may be over or underdetermined, which implies that the Newton method used for optimization cannot be applied without modification. However, first order methods such as those used in this document (RMSprop) remain valid options to minimize a scalar residual crafted from the vector observable.

additionally equal those of a formal divergence. However, due to the envelope theorem, the derivatives calculated for both ARCG and IMC/NI share a similar covariance form as shown in Eq. (4.6). Additionally, the vector observables considered in IMC/NI are typically connected to the form of potential being optimized. As an example, when optimizing pairwise spline potentials the observable is closely related to the empirical radial distribution function. The calculation of observables of this type generally follows a histogram based approach which can limit the generalization to higher dimensional potentials. We note that while ARCG does in practice use similar statistics, no histogramming is performed.

There exist additional CG methods that are difficult to directly compare to ARCG (e.g., references [40, 43–46, 48]). However, in general, most methods considered make assumptions that strongly inhibit virtual site application. Specifically, methods often assume that the CG potential (or its derivatives) can be applied at the resolution of the reference CG samples acquired (either through calculation of the residual or the update strategy facilitating optimization), or that samples with virtual particles can be obtained, although extensions are sometimes feasible. For example, traditional MS-CG force-matching optimizes the CG force-field to optimally match mapped forces; with a general linear $\mathcal{G}$ and $U_{\text{mod}}^{\text{pre}}$ this would likely require an iterative procedure to determine the mean force implied at the CG resolution by $\mathcal{G}$ and $U_{\text{mod}}^{\text{pre}}$. Alternatively, gYBG (in the case of pairwise potentials) inverts two- and three-body CG correlations to produce a force-field at the corresponding resolution of the observed correlations; similarly, Iterative Boltzmann Inversion requires a map to define the iterations that connect modifications in the potential to changes in the observed correlations (which is nonintuitive when considering parameters associated with general virtual sites). These limitations often do not appear to be fundamental ones, but rather one of implementation; extensions to these methods that circumvent this limitation are likely possible. One such extension is given for REM in section 4.4. Beyond the extension described, there are three straightforward strategies to remove this limitation, the first two of which I know are in current use. First, methods such as binning or kernel density estimation are used to approximate the probability density at a resolution differing from the CG configurational Hamiltonian (e.g., the radial distribu-

tion approach in reference [55]). This approach is often limited to lower dimensional spaces when comparing models. Second, constraints are placed on virtual sites such that $U_{\text{mod}}^{\text{pre}}$ may be related via closed expression to U_{mod}[162]. This approach inherently requires limiting the type of virtual site considered. Third, methods that allow the observed mapped FG sample to be backmapped to the pre-CG domain are applied and then traditional approaches are used on the backmapped sample (although this does seem possible without modification in ref [63] due to its generative formulation). This approach requires that the reference virtual particles are assigned a distribution using a set rule. We note that this final strategy does not necessarily directly optimize U_{mod}; rather, it optimizes $U_{\text{mod}}^{\text{pre}}$ to match the backmapped distribution (which would still result in configurational consistency at the resolution of $\mathcal{X}_{\mathbf{R}}$). These extensions may additionally be used with various methods discussed in the previous paragraphs; for example, traditional REM CG is applicable without modification if a suitable backmapping strategy exists. In contrast, ARCG is well suited to higher dimensions, imposes no constraint on the virtual sites, and does not require backmapping; however, it incurs increased training complexity.

Finally, we note that while there is significant overlap between ARCG and GANs with respect to the residual calculation and optimization, the method by which samples are produced in the models is conceptually distinct. GANs are characterized by transforming noise to a fit a desired distribution; the optimization of the model parameters modifies the nature of this transformation. In contrast, the transformation present in ARCG is held constant, while the underlying sample generating process is modified.

4.4 IMPLEMENTATION

Previous sections have provided abstract descriptions of the ARCG method, including the specific form with connection to f-divergences. In this section we provide the corresponding concrete expressions for optimizing models using relative entropy by implementing the classification based approach described in Sec. 4.3.3. The corresponding expressions are compared to previous expressions used for REM, and additionally used to derive an alternative method to the one pursued.

Additional practical points on implementation, relaxations of the method for stability, and the specification of Q are also discussed.

As previously noted, the relative entropy between p_{ref} and p_{mod} is an f-divergence and is obtained by setting $f(x) := x \log x$. This implies equivalence with a classification task with the aforementioned losses in Eq. (4.24), from which we derive the model optimization statement using Eq. (4.9) and associated gradients using Eq. (4.6), such that

$$\mathcal{F}^{\text{RE}}\left[p^{\text{pre}}_{\text{mod},r,\theta}, p_{\text{ref},\mathbf{R}}; \mathcal{G}\right] = \max_{\hat{\eta}} \left\{ -\left\langle \log\left(\frac{1-\hat{\eta}}{\hat{\eta}}\right)\right\rangle_{p_{\text{ref},\mathbf{R}}} - \left\langle \frac{\hat{\eta}\circ\mathcal{G}}{1-\hat{\eta}\circ\mathcal{G}}\right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} \right\} \tag{4.32}$$

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathcal{F}^{\text{RE}}\left[p^{\text{pre}}_{\text{mod},r,\theta}, p_{\text{ref},\mathbf{R}}; \mathcal{G}\right] = &-\beta\left\langle \frac{\eta\circ\mathcal{G}}{1-\eta\circ\mathcal{G}}\right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} \left\langle \frac{\partial U^{\text{pre}}_{\text{mod}\theta}}{\partial\theta_i}\right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} \\ &+\beta\left\langle \left(\frac{\eta\circ\mathcal{G}}{1-\eta\circ\mathcal{G}}\right)\frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial\theta_i}\right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}}\end{aligned} \tag{4.33}$$

where we have discarded irrelevant constants. This comprises a full residual and associated gradient for optimization. However, in practice, the loss functions are poorly behaved: pointwise values of $\hat{\eta} = 1$ easily create a divergent residual value (identical to the corresponding situation with the traditional relative entropy estimation methods). Fortunately, the optimal η is shared among all proper losses[81]. As a result, η can be similarly discovered with the corresponding statement using the log-loss[81, 191]

$$\eta = \operatorname*{argmin}_{\hat{\eta}} \left\{ \langle \log\hat{\eta}\rangle_{p_{\text{ref},\mathbf{R}}} + \langle \log(1-\hat{\eta}\circ\mathcal{G})\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} \right\} \tag{4.34}$$

while the gradient estimation remains unchanged. To summarize, the models trained in this chapter indirectly minimize Eq. (4.32) by producing derivatives over θ via Eq. (4.34) and Eq. (4.33), where η retains the same meaning across equations. This equality only rigorously holds when assuming that the variational search is considered using population integrals and finds the global minimum.

While the envelope theorem, along with the work of Reid & Williamson [81], imply the validity of Eqs. (4.32) and (4.33), it is helpful to analyze these equations external to these theorems. This analysis provides an alternative view to the optimization performed in this document. First, the statement given in Eq. (4.32) can be seen to be maximized at the relative entropy as follows. Assuming that no virtual particles are present, note that by substituting Eq. (4.10) one finds that $\frac{\eta}{1-\eta} = \frac{p_{\text{ref},\mathbf{R}}(x)}{p_{\text{mod},\mathbf{R}}(x)}$. In this case substituting η we find

$$\mathcal{F}^{\text{RE}}\left[p^{\text{pre}}_{\text{mod},r,\theta}, p_{\text{ref},\mathbf{R}}\right] = -\left\langle \log\left(\frac{p_{\text{mod},\mathbf{R}}(x)}{p_{\text{ref},\mathbf{R}}(x)}\right)\right\rangle_{p_{\text{ref},\mathbf{R}}} - \left\langle \frac{p_{\text{ref},\mathbf{R}}(x)}{p_{\text{mod},\mathbf{R}}(x)}\right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} \tag{4.35}$$

The second term reduces to the integral of a normalized probability density and is therefore 1, resulting in

$$= \int_{\chi^{\text{FG}}_{\mathbf{r}}} p_{\text{ref},\mathbf{R}}(x) \log\left(\frac{p_{\text{ref},\mathbf{R}}(x)}{p_{\text{mod},\mathbf{R}}(x)}\right) \mathrm{d}x - 1, \tag{4.36}$$

which corresponds to the relative entropy with a known offset. Second, to see that this is in fact the global maximum, index the space of functions $([0,1]^{\chi_{\mathbf{R}}})$ using the following sequential transformations:

1. $f(x) \mapsto \frac{f(x)}{1+f(x)}$
2. $f(x) \mapsto f(x) p_{\text{mod},\mathbf{R}}(x)$
3. $f(x) \mapsto (Z_f, Z_f^{-1} f(x))$ where $Z_f := \int f(x) \mathrm{d}x$

Note that each step is invertible when $p_{\text{mod},\mathbf{R}}(x) \neq 0$, which only happens for typical molecular potentials on a set of measure 0. Denoting $F_f(x) = Z_f^{-1} f(x)$ Eq. (4.32) becomes

$$\max_{(Z_f, F_f)} \left\{ -\left\langle \log\left(F_f(x)\right)\right\rangle_{p_{\text{ref},\mathbf{R}}} + \log Z_f - Z_f \right\} \tag{4.37}$$

Gibb's inequality (alternatively, the positivity of the relative entropy) implies that $F_f = p_{\text{ref},\mathbf{R}}$ while $Z_f - \log Z_f$ is minimized at 1, implying $Z_f = 1$. Continuing to Eq. (4.33), substituting the η results

in

$$\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathcal{F}^{\mathrm{RE}}\left[p^{\mathrm{pre}}_{\mathrm{mod},r,\theta}, p_{\mathrm{ref},\mathbf{R}}\right] = \beta\left\langle \frac{\partial U^{\mathrm{pre}}_{\mathrm{mod}\theta}}{\partial\theta_i}\right\rangle_{p_{\mathrm{ref},\mathbf{R}}} - \beta\left\langle \frac{\partial U^{\mathrm{pre}}_{\mathrm{mod}\theta}}{\partial\theta_i}\right\rangle_{p_{\mathrm{mod},\mathbf{R}}}, \tag{4.38}$$

which is identical to that calculated for relative entropy using traditional methods[174].

While reproducing established formula for residuals and derivatives improves confidence in our expressions, the corresponding formulas can additionally be applied at a coarser resolution. While doing so to Eq. (4.32) does not produce additional insight, substitution with Eq. (4.33) results in the following formula (see appendix F)

$$\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathcal{F}^{\mathrm{RE}}\left[p^{\mathrm{pre}}_{\mathrm{mod},r,\theta}, p_{\mathrm{ref},\mathbf{R}}; \mathcal{G}\right] = \beta\left\langle \overline{\frac{\partial U^{\mathrm{pre}}_{\mathrm{mod}\theta}}{\partial\theta_i}}\right\rangle_{p_{\mathrm{ref},\mathbf{R}}} - \beta\left\langle \overline{\frac{\partial U^{\mathrm{pre}}_{\mathrm{mod}\theta}}{\partial\theta_i}}\right\rangle_{p^{\mathrm{pre}}_{\mathrm{mod},r,\theta}}, \tag{4.39}$$

where $\overline{\frac{\partial U^{\mathrm{pre}}_{\mathrm{mod}\theta}}{\partial\theta_i}}$ corresponds to to the partial derivative averaged over the virtual particle positions for a given CG configuration. Critically, while in Eq. (4.38) the partials are evaluated without issue on configurations from both the reference and model ensembles, this is not possible in the case of virtual particles (the configurations from the reference ensembles do not have virtual particles). Eq. (4.39) states that one can do least squares regression to predict the value of $\frac{\partial U^{\mathrm{pre}}_{\mathrm{mod}\theta}}{\partial\theta_i}$ using configurations without virtual particles from the CG model. The resulting function (which does not require virtual particles for its evaluation) can be used as a drop in replacement for a first order REM update. These expressions have strong similarities to those found in expectation maximization methods. We leave implementation of Eq. (4.39) to future work; later numerical optimization results in this chapter instead use the previously derived adversarial approach.

The numerical examples section 4.5 are computed in the following way. First, the CG (or pre-CG) model is represented using a force-field and samples are generated using standard molecular dynamics software. These samples are mapped if necessary using $\mathcal{G}$. Reference examples are similarly generated and mapped using $\mathcal{M}$ if needed. The variational estimator is represented using either a neural network, logistic regression, or gradient boosted decision tress which implement Eq. (4.34). The estimator then is trained on the reference and model samples. Finally, the gradient

is calculated using the output of the variational estimator, Eq. (4.33), and the model samples; this gradient is then used to update the model parameters. This process is iterated, although the reference sample is not regenerated. The variational estimators are not fed the Cartesian coordinates of the input system directly; instead, various features are calculated for each frame, and these features are given as input to the variational estimator. This has the effect of constraining that $\hat{\eta}$ be a function of these features. Additional points on each of these details is discussed for each example or may be found in the appendix.

In some cases of ARCG, including the case of f-divergence estimation, the functions achieving the inner maximum with a complete $\mathcal{Q}$ can be expressed as a pointwise function of the mapped distributions. Specifically, as noted in Eq. (4.10), in the case of relative entropy the optimal witness function η is expressible as a function of the conditional class densities (p_{mod} and p_{ref}, where we have dropped the configurational subscripts). This can guide how elements of a tractable $\mathcal{Q}$ are parameterized. When the algebraic forms of p_{ref} and p_{mod} are known to be functions of summary statistics of their respective systems (e.g., the inverse 6 and 12 moments in a traditional Lennard-Jones potential[199]), we can often express a complete $\mathcal{Q}$ exactly with a manageable number of terms. However, this is not true of practical bottom-up CG application: the form of the mapping operator does not provide us with an algebraic understanding the implied mapped free energy surfaces. The resulting η does maintain invariances with the free energy surfaces it is composed of (e.g., rotational and translational invariances).

When considering the creation of CG models for realistic chemical systems it is therefore reasonable to examine how an incomplete $\mathcal{Q}$ would affect the approximate model produced by ARCG. This situation may arise when providing an incomplete feature set as input to a parametric representation of a function or when the parametric representation itself is limited in its capacity (e.g., when logistic regression is used). The implications of general restrictions on $\mathcal{Q}$ are complex and reserved for future studies; however, in the case of f-divergences certain conclusions are clear without substantial analysis:

1. When considering identical p_{ref} and p_{mod}, η is a constant function and is typically included

in Q. In other words when p_{mod} completely satisfies configurational consistency Q is typically complete. Additionally, this η provides a lower bound on the divergence estimate given by ARCG for any p_{mod} considered. Assuming Q contains this function (but not assuming completeness), if the residual estimated for a candidate p_{mod} is not zero, the distribution considered does not satisfy configurational consistency and the Q considered is always expressive enough to determine that the model satisfying configurational consistency is preferable. Alternatively, if a divergence of zero is returned for a given p_{mod}, this is motivation to expand the expressiveness of Q.

2. If the parametric estimator acting on the given set of features is assumed to be fully expressive, then the divergence returned by ARCG is simply the divergence calculated at the resolution of the features themselves. The model optimization process can be recast as one with a complete Q at the resolution given by composing $\mathcal{M}$ and $\mathcal{G}$ with the featurization function. For example, suppose one is using ARCG to create a CG lipid bilayer with a solvent composed of virtual particles. However, additionally suppose there exists solvent in the FG system that could be connected to the behavior of the CG solvent. One could reclassify the virtual particles present instead as typical CG sites, but then limit the variational search to functions which do not take the solvent behavior into account, producing an identical optimization principle as that of the virtual solvent approach described previously.

3. An incomplete Q can change which approximate model ARCG returns. For example, consider selecting one of two models which reduces the relative entropy with respect to a reference distribution. In this example our model and reference distributions each contain two degrees of freedom and their probability distributions are independent with respect to these dimensions. Suppose that the first candidate model has a different probability distribution along the first degree of freedom as the reference data but the same distribution along the second degree of freedom. Additionally, suppose that second candidate model is similar except that the roles of the first and second degrees of freedom are reversed: the first degree

of freedom has the same distribution as the reference data but the second degree of freedom differs. If we consider variational estimators which use a $\mathcal{Q}$ limited to functions of only the first or second degrees of freedom, it is clear the optimal model chosen is a function of the particular incomplete $\mathcal{Q}$ selected.

The effect of these properties on the parameterization of complex models will be investigated in a future work.

MS-CG and REM CG are known to be convex when paired with potentials which vary linearly with respect to their parameterization[45, 46, 200]. ARCG requires two types of optimization: the solution of the variational statement defining the divergence and the optimization of the CG model parameters.[7] The convexity of inner variational statement defining the divergence depends on the particular divergence chosen and the representation of the function space used (e.g., logistic regression or neural network). The optimization estimating the divergence for the examples presented in this document is convex when using logistic regression and nonconvex when using a neural network or gradient boosted decision trees. The outer optimization modifying the CG model is characterized by the parameterization of the CG force-field, the divergence emulated, and the completeness of $\mathcal{Q}$. When $\mathcal{Q}$ is complete the characteristics of the divergence emulated can be used to check for convexity; for example, the optimization of the CG parameters for the simple Lennard-Jones liquid example presented is convex. However, when $\mathcal{Q}$ is incomplete or the potential is nonlinear convexity is unclear, as is true for the remainder of the examples presented.[8]

The integrals characterizing the variational residual are computationally approximated as sample averages. Optimizing a function using a sample average introduces the possibility that the function which maximizes the sample average is a poor approximation of the function which maximizes population average. In the context of classification this error is captured by considering whether

[7] Convergence results exist for Generative Adversarial Networks in the case where it is not assumed that the variational divergence estimation converges at each iteration; as an example, see Heusel *et al.* [201]. However, such analysis moves beyond the divergence based interpretation presented and we reserve further discussion of this topic to future work.

[8] It is important to note that virtual particles potentials which use linear parameterizations at the pre-CG resolution are not necessarily linear at the CG resolution.

the classifier is overfitting the data sample. There are multiple strategies to overcome this[191]; in the current study we use *L*2 regularization for neural networks and only allow the variational estimator to update a limited number of times at each iteration. More complex regularization is used with gradient boosted decision trees and is described in section 4.5.6. When optimizing examples with flexible potentials and large feature sets (for example, the water, methanol, and dodecalanine models presented in the next section), we have found that using a neural network quickly overfits the data provided, even with relatively strong regularization. However, reducing the number of iterations allowed at each step of variational optimization causes the neural network to exhibit considerable hysteresis between iterations, causing the force-field being optimized to orbit around an ideal solution. To ameliorate this we use logistic regression in the case of liquids, where the solution to the logistic regression is optimized using a limited number of iterations of l-BFGS, and gradient boosted decision trees were used in the case of dodecaalanine. Note that the output of logistic regression and gradient boosted decision trees readily afford estimates of the class conditional probabilities, which in turn directly connects their optimal solution to η.

The issues associated with overfitting and hysteresis are intrinsically connected the size of the finite samples used to approximate the integrals in Eq. (4.1). Overfitting would be reduced by increasing the sample size, which in turn would allow additional optimization at each variational iteration and would therefore reduce hysteresis. In practice, we have found that increasing the sample size to the point that the hysteresis is removed slows down the rate of force-field optimization considerably due to the time needed to both generate molecular samples and evaluate high dimensional gradients. This difficulty naturally suggests the use of modified sampling strategies to reduce the discrepancy between the sample and population averages. As Eq. (4.1) only involves expectation values any modified sampling scheme which allows for the calculation of an unbiased ensemble average is a candidate for this strategy. It is worth noting that ARCG selects an optimal observable function based on the ensemble averages of a large set of candidate observables, and that the error for each observable may be different, complicating the use of variance reduction techniques which are designed for a single observable. It would also be possible to improve the

sampling of the feature space on top of which Q is represented; for example, if Q is represented by a neural network acting on two statistics calculated for each sample, free energy estimation may be used to better resolve the joint distribution of these two statistics themselves. This approach could be extended to produce a parameterization method which improves the observable estimates on the fly, such as that in Abrams & Vanden-Eijnden [202]. While we have not pursued these strategies here, future applications to more complex systems will likely need to consider these options.

The variational search over possible $\hat{\eta}$ was either performed via a neural network outputting class probability predictions penalized via the log-loss or, logistic regression, or gradient boosted decision trees penalizing the log-loss. Logistic regression was used in the cases of water and methanol, gradient boosted decision trees were used in the case of dodecaalanine, and neural networks were used in all other cases. All neural networks used in examples in this document utilized a simple feed-forward architecture with at least two layers (not including the input and output layer). The results were found to be insensitive to the architecture chosen, and the specific architectures used may be found in appendix G. All internal nodes used rectified linear activation functions with the output normalized via softmax. The duality with classification underpins the utility such traditional choices have in our variational search.

In practice, we have noticed that ARCG optimization may suffer from instability, especially when optimizing the parameters of a model that produces a distribution significantly different than its optimization target. This issue can be noted by observing that the classifier achieves 100% accuracy during parameterization, producing uninformative gradients. In these cases we find that an effective strategy is to introduce standard Gaussian noise into both the model and reference samples; the variance of this noise is gradually reduced to zero as the optimization progresses. It is likely that a correct local minima is achieved in this case as the optimization appears stationary at the end of minimization, but it is unclear if the selection of a specific local minima is biased using this strategy.

A proof-of-concept python/Lammps based implementation is available at upon contact with the Voth Group at the University of Chicago. This code base makes extensive use of the theano,

theanets, pyLammps, numpy, scikit, and dill libraries. All computational examples presented in this document may be found in the test portion of this code, which includes the complete settings used to generate the data used. Visualizations and analysis were performed with the matplotlib and seaborn libraries, as well as the base plotting system, ggplot2, rgl, and data.table packages in R. Extensions providing scalability for more complex systems and potentials are a topic for future work.

4.5 RESULTS

The adversarial relative entropy approach described in section 4.4 in Eqs. (4.33) and (4.34) was applied to six test systems. First, a simple single component 12-6 Lennard-Jones (LJ) system was optimized to approximate a reference LJ system at the same resolution (no virtual particles were present, and no coarse-graining of either the reference or model was performed). Second, a system representing two bonded real particles where force is partially mediated by a single harmonically bonded virtual particle was optimized to approximate a reference system of the same type. Third, a binary LJ liquid undergoing phase separation was simulated and optimized after particles of a single type had been integrated out; this distribution was fit to match a similarly integrated binary LJ system. Fourth, a CG model using pairwise *b*-spline interactions and a single site per molecule was used to approximate liquid methanol. Fifth, a CG model using pairwise *b*-spline interactions and a single site per molecule was used to approximate liquid water. Sixth, dodecaalanine was modeled using pairwise potentials similar to those described in section 3.5.1.In the fourth and fifth cases we observed good convergence of suitable correlation functions; however, in cases with virtual particles we found that numerically recovering the known parameters of the reference system is difficult; in other words, it seems likely that the parameter space is either redundant or sloppy[203], with similar correlation functions arising from distinct parameter sets. We note that while the potentials considered here are relatively simple, ARCG is fully applicable to more complex potentials such as those in Zhang *et al.* [64].

The first three examples considered here are theoretically able to capture the reference distributions used for fitting (i.e., the model optimized is not misspecified). This is ensured by generating

reference data using a force-field that is directly representable by the CG force-field family. For example, the LJ CG model in the first example was optimized to reproduce the statistics generated by a particular LJ reference potential. Additionally, when either the reference or model are modified using a mapping function, this mapping operator is forced to be the same between the two systems, and the reference data is again produced using a force-field which is expressible by the CG model. For example, in the case of the virtual solvent LJ system, a distinct system of binary LJ particles was simulated for both the model and reference data samples, each with differing parameter sets. Both systems then had the particles of a specific shared type integrated out. The resulting integrated distributions were then the basis of comparison used to train the model parameters (with new statistics being created for the CG model at each iteration). This is not true for the examples approximating water, methanol, and dodecaalanine: here, the CG model is approximating the mapped distributions using a pairwise potential and is unable to capture the true free energy surface.

4.5.1 LENNARD-JONES FLUID

A single component 12-6 LJ fluid was simulated with 864 particles at 300K (the potential form is given in Eq. (4.40) with r_{ij} denoting the Euclidean distance between particles i and j).

$$U(\mathbf{R}^{3N}) = 4\varepsilon \sum_{i>j} \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right] \tag{4.40}$$

The system was simulated at constant NVT conditions using a Langevin thermostat with coupling parameter set to 100.0 fs and a timestep of 1.0 fs. No virtual particles were present; i.e., $\mathcal{G}$ and $\mathcal{M}$ are set to be the identity function. Inverse sixth and twelfth moments were used as input to the variational estimator (in this case, this set of features is known to be complete, see appendix G for details). System A_{initial} with $\varepsilon_{A_{\text{initial}}} = 0.6\text{kcal/mol}$ and $\sigma_{A_{\text{initial}}} = 3.5\text{Å}$ was optimized to match the statistics of system B characterized by $\varepsilon_B = 0.75\text{kcal/mol}$ and $\sigma_B = 3.0\text{Å}$. Upon optimization, the parameters of A were seen to quantitative converge to those of B: $\varepsilon_{A_{\text{opt}}} = 0.746\text{kcal/mol}$ and $\sigma_{A_{\text{opt}}} = 3.00\text{Å}$. Additionally, convergence of the pairwise correlation functions (Fig. 4.3) was observed. The

initial parameters of A resulted in a homogeneous liquid, while those of system B (and system A upon optimization) resulted in liquid-vapor coexistence. During training Gaussian noise was used to smooth out initial gradients to resolve initial soft wall differences; this noise is reduced to zero by the end of optimization. Optimization was performed using RMSprop[140] with individual rates for each parameter. These results demonstrate good convergence properties with small parameter sets when no virtual particles are considered in the pre-CG resolution.

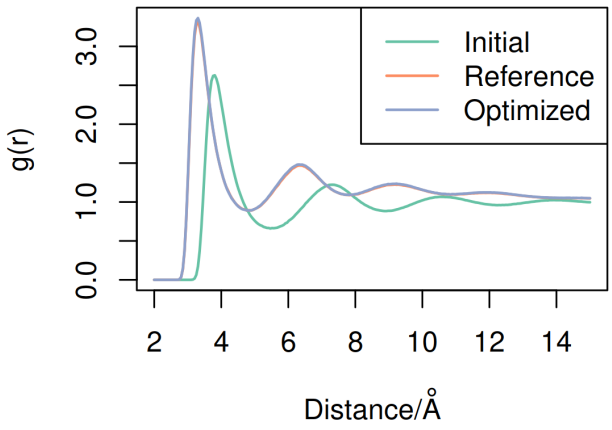

Figure 4.3: Radial distribution functions calculated for the unoptimized system A_{initial}, the reference system B, and the optimized system A_{opt}. Note that the reference and optimized RDFs are within line thickness.

4.5.2 VIRTUAL BOND SITE

A system of three particles completely connected via harmonic bonds was simulated at 300 K. The system was propagated in constant NVT conditions using a Langevin thermostat with coupling parameter set to 100.0 fs and a timestep of 1.0 fs. Two types of particles were present; we denote the types of the particles X, Y, X. Upon application of $\mathcal{M}$ and $\mathcal{G}$ the Y particle is removed, resulting in a system composed of two particles of type X (i.e., the Y particle is a virtual site). This mapped system is optimized using the distance between the two X particles as input to the discriminator; in this case, this feature set is complete. Initial, optimized, and reference parameters are seen in

table 4.1. Optimization was performed using RMSprop. Convergence to a specific parameter set

System	$x_{XY}/\text{Å}$	$k_{XY}/\frac{\text{kcal}}{\text{mol}}\text{Å}^{-2}$	$x_{XX}/\text{Å}$	$k_{XX}/\frac{\text{kcal}}{\text{mol}}\text{Å}^{-2}$
B	2	2.7	2.3	0.4
A_{initial}	0.65	2.2	1.4	0.15
A_{opt}	1.70	2.06	2.66	0.224

Table 4.1: Parameters for systems with virtual bonded sites. x denotes the zero energy point of the bond while k denotes bond strength. Subscripts specify the particle types between which the bond acts. System A_{initial} was optimized to match system B, resulting in A_{opt}.

that reproduces observed correlations (Fig. 4.4) is fast; however these parameters differ from the parameters of the reference system. Additional simulations were run where the CG model was initialized with parameters set to those of the reference system (results not shown); in this case, we observed local diffusion around a small set of parameters including the true set. This suggests that virtual particles may create degeneracy in model specification in practice (i.e., even if the model parameters are identifiable, the specification is sloppy). This case represents an application

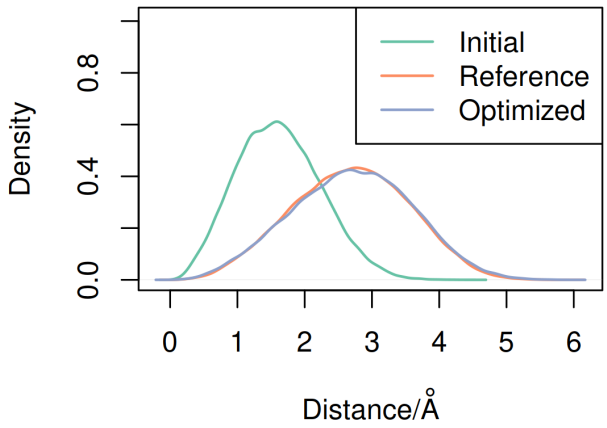

Figure 4.4: Bond distance distribution functions calculated for the unoptimized system A_{initial}, the reference system B, and the optimized system A_{opt}.

where a pairwise force-field may be augmented via bonded virtual particles to create modified

correlations.[9] For example, a heterogeneous elastic network[149] may be augmented by introducing virtual particles to facilitate higher order correlations.

4.5.3 VIRTUAL SOLVENT LENNARD-JONES FLUID

A binary system composed of 864 LJ particles of types X and Y was simulated at 300 K. The system was simulated at constant NVT conditions using a Langevin thermostat with coupling parameter set to 100.0 fs and a timestep of 1.0 fs. Equal numbers of X and Y particles were present prior to the application of mapping operators; upon application all particles of type Y were removed (i.e., the Y particles are virtual sites). The target system was parameterized to undergo phase coexistence, while the unoptimized CG model was well mixed. Parameters are found in table 4.2. Optimization was performed using RMSprop with rates adjusted for each parameter. Gaussian noise was used to stabilize initial training. Visual inspection of representative molecular configurations showed greatly improved similarity for the optimized parameter set (Fig. 4.5). Again, while convergence of correlation functions is readily observed (Fig. 4.6), parameters do not converge to those of the reference system, likely due to sloppiness in specification.

System	$\sigma_{XX}/\text{Å}$	$\varepsilon_{XX}/\frac{\text{kcal}}{\text{mol}}$	$\sigma_{YY}/\text{Å}$	$\varepsilon_{YY}/\frac{\text{kcal}}{\text{mol}}$	$\sigma_{XY}/\text{Å}$	$\varepsilon_{XY}/\frac{\text{kcal}}{\text{mol}}$
B	0.7	3.6	0.7	3.6	0.35	3.5
A_{initial}	0.6	3.5	0.6	3.2	0.5	3.1
A_{opt}	0.713	3.600	0.722	3.594	0.349	3.494

Table 4.2: Parameters for the integrated LJ systems. Subscripts specify the particle types between which the potential acts. System A_{initial} was optimized to match system B, resulting in A_{opt}.

This case is representative of the situation where higher order correlations may be captured by the addition of virtual solvent particles. For example, the hydrophobic driving force underlying a CG lipid bilayer could be facilitated by a virtual solvent. This is distinct from using traditional explicit solvent where each solvent molecule is directly connected to the FG reference system: there,

[9] Systems using harmonically bonded virtual particles have been considered in the past, and under certain constraints can be shown to not affect the stationary distribution of the CG particles (see Zwanzig [204, 205] and Davtyan *et al.* [206]). The particles considered here do not satisfy these constraints.

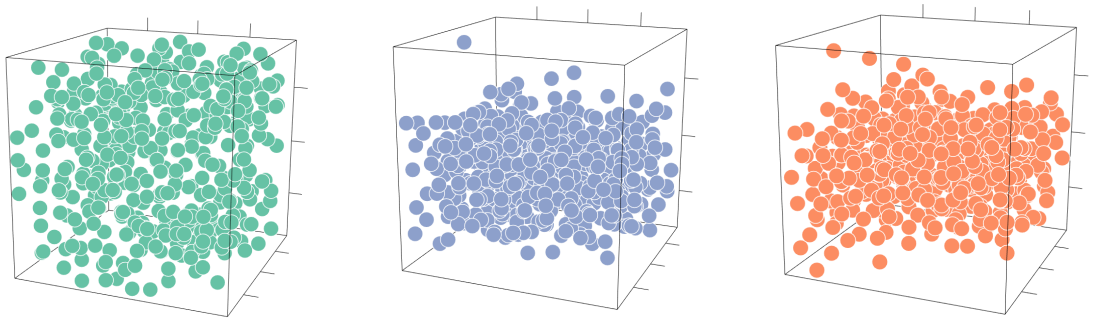

Figure 4.5: Sample configurations of the unoptimized model (green), the optimized model (blue) and the reference data (orange). Configurations are shown at the resolution of comparison, i.e., after the application of $\mathcal{M}$ and $\mathcal{G}$. Slab type formation, similar to that present in the optimized model, is seen after parameter optimization.

the behavior of the solvent is incorporated into the quality of the model, as where the approach of ARCG ignores the direct solvent behavior.

4.5.4 SINGLE SITE METHANOL

Methanol was modeled using a single site CG liquid. The reference atomistic (FG) trajectory of 512 molecules was simulated in the NVT ensemble at 300K with a Nose-Hoover damping time of 1 ps after NPT equilibration at 1 atm. The OPLS-AA[207–209] force-field was used in the atomistic system. The FG system was mapped to the to the CG resolution by retaining only the central carbon; no virtual sites were present in the CG system. The CG potential was described using a pairwise *b*-spline potential using 15 equally spaced knots and a 10 Å cutoff (the last three control points were set to zero to enforce a smooth decay at the cutoff). The CG system was run at the same temperature and volume as the FG system using a Langevin coupling parameter of 100 fs and a timestep of 1 fs. The starting potential used for the simulation was a WCA potential (Fig. 4.8). Quantitative reproduction of the radial distribution function was observed (Fig. 4.7). Convergence was smoothed using Gaussian noise whose standard deviation decayed to zero by the end of the optimization.

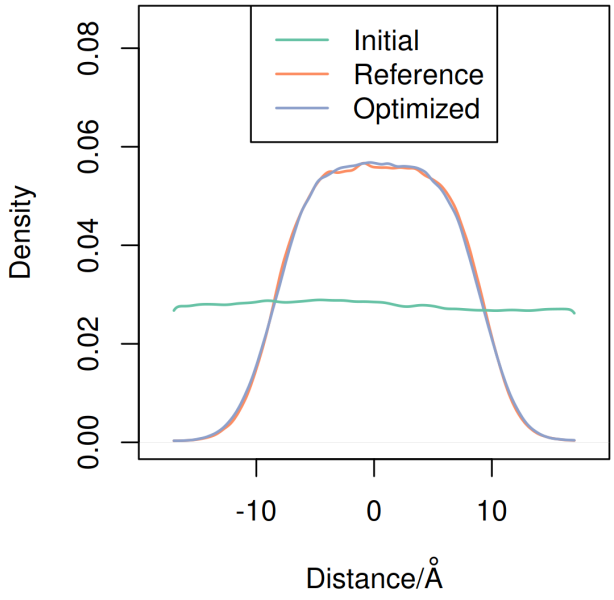

Figure 4.6: Probability densities across the slab type formations present in the integrated binary LJ systems (along the z axis of the simulation box). No slab structure is present in the initial model.

4.5.5 SINGLE SITE WATER

A single site model of water was trained using 512 molecules of SPC/E water simulated at 300K. The molecular (FG) system was equilibrated at 1 atm and production NVT samples were produced with the Nose-Hoover thermostat with a 1 ps damping time. The mapping connecting the FG system to the CG system was the center of mass mapping; no virtual sites were present in the CG system. The CG system potential was limited to pairwise interactions with a 7 Å cutoff and was parameterized using b-splines with 37 knots (see appendix G for knot locations and further details). The last three spline control points were set to zero to enforce continuity at the cutoff. The starting potential used for the simulation was a WCA potential (Fig 4.9). CG simulations were run at the same volume as the FG system with a Langevin thermostat, whose coupling parameter was set to 100 fs, and a 1 fs timestep. Quantitative reproduction of the radial distribution function was obtained (Fig 4.10). Gaussian noise was used initially to smooth convergence and was tapered to a

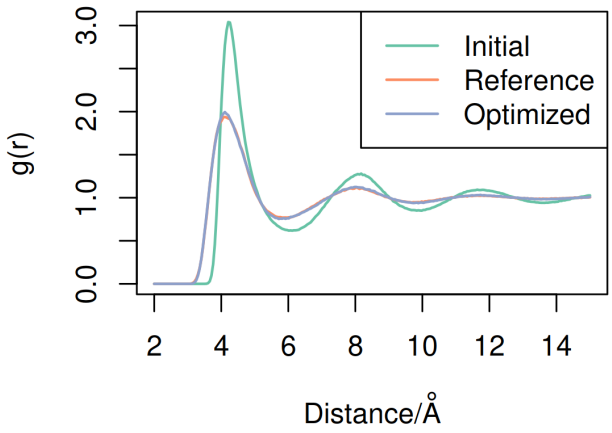

Figure 4.7: Radial distribution functions for the reference, initial, and optimized methanol systems. Note that the optimized and reference RDFs are nearly within line thickness.

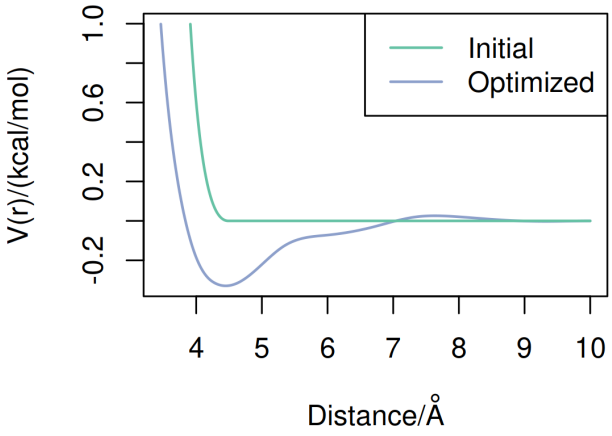

Figure 4.8: Pairwise potential functions characterizing the initial and optimized methanol systems.

standard deviation of zero by the end of the optimization.

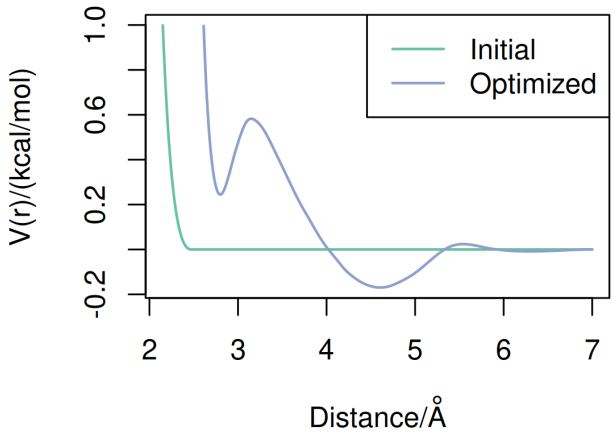

Figure 4.9: Pairwise potential functions characterizing the initial and optimized water systems.

4.5.6 DODECAALANINE

DDA was modeled at the 12 site resolution. Modeling details such as the data used as reference, the force-field basis, and the CG mapping operator used are the same as those described in section 3.4.1. Only models with two site types were considered. DDA is difficult to model at the amino acid resolution using pairwise potentials: as discussed in section 3.5.1, models at this resolution have difficulty capturing the observed configurational statistics. However, as seen in this section, ARCG can successfully create reasonable models using distance matrices and gradient boosted decision trees. While Eq. (3.2) demonstrates the utility of specialized molecular machine learning approaches (e.g. those found in references [60, 64, 68, 159]) when designing adversaries, the results presented in this section use no such methods. These more sophisticated methods will likely be critical to future applications.

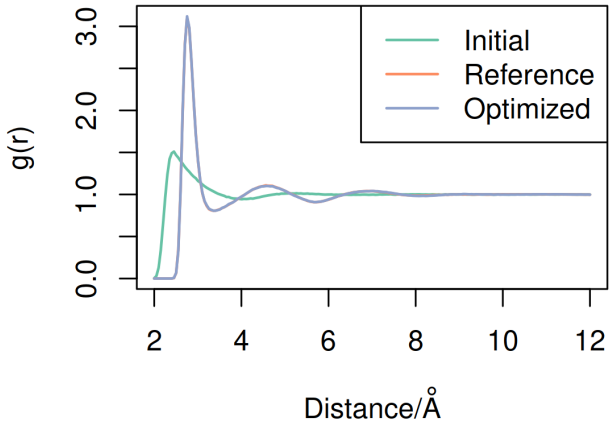

Figure 4.10: Radial distribution functions for the reference, initial, and optimized water systems. Note that the optimized and reference RDFs are within line thickness.

4.5.6.1 ADVERSARIES

Adversary design for DDA was found to be more difficult than for previously modeled systems. A large variety of possible adversaries were tested, including logistic regression, neural networks, Gaussian processes, and tree ensembles, each of which was studied using a variety of hyperparameters. Each method was tested using the full distance matrix of the protein as input[10]. The only methods found to produce stable results were random forests and gradient boosted decision trees (which, due to their explainability, were previously used in section 3.4.1). While unsuccessful adversaries were generally able to distinguish samples from the model and reference distributions, various issues arose during optimization:

1. The resulting gradients were unstable.

2. The resulting potentials did not produce accurate correlations.

3. The classifier optimization was too slow.

Feedforward neural networks were found to suffer from issue 1, Gaussian processes suffered from

[10]Various per-distance transformations, such as $x \mapsto 1/x^2$ were additionally tested.

issue 2, and logistic regression suffered from issue 3.[11] Tree ensemble methods, when used with appropriate hyperparameters, were found to produce accurate correlations at reasonable computational cost.

However, the tree based results presented in this section are preliminary. The models discussed have been optimized extensively and are converged; however, the hyperparameters discovered were empirically selected from hundreds of hyperparameter combinations. The individual impact of each hyperparameter has not been discerned, nor have the proposed hyperparameters been used when studying a additional proteins. Furthermore, optimization of the proposed force-fields is slow, requiring approximately two node-months of computer time. These difficulties, however, do not seem insurmountable. First, while initial efforts in understanding successful adversaries was inhibited due to the lack of a successful example, the presented results provide a coherent starting point for future adversary design. Second, training time can be reduced by using accurate starting potentials such as those derived from Multiscale Coarse-Graining[43–46] or Hetero-elastic network models[149]. The cost of training is largely due to calculation of gradients and not the proposed variational search or featurization, and as a result may be accelerated without delving into highly optimized classification libraries.

Two ARCG models are presented and compared to the corresponding REM model and reference data in the current section. The first ARCG model is designed using the losses and approach used in previous sections (Eqs. (4.33) and (4.34)), resulting in a variational estimation of the relative entropy. The second model optimizes the Hellinger distance (Eq. (4.28)) using the losses presented in Eq. (4.30); these losses are used to derive derivatives analogous to Eq. (4.33) and are similarly optimized using Eq. (4.34). The first model is referred to as ARCG-REM and the second as ARCG-Hel. Both models were optimized using the gradient boosted decision trees with dropout[125]; however, the hyperparameters used differ between the two models.

ARCG-REM was optimized using ensembles of 1000 trees. Trees were limited to have 5

[11] Logistic regression was applied to a binned version of the distance matrix; this binning was expensive and did not produce better results than tree based methods. Furthermore, the completeness of the resulting function search seemed unlikely.

terminal leaves and 1023 bins were used to discretize each feature. 180K molecular sample were generated by sampling every 100 timesteps. At each iteration 40% of these samples were used for classification while the remaining 60% are used to observe overfitting trends as a validation set. 45K samples were used each iteration to calculate the gradient using Eq. (4.33); these samples were drawn from both the validation and train sets. Additional experiments limiting gradients samples to the validation set did not meaningfully change results. RMSprop was used to translate gradients into parameter updates using a rate of 0.009 and a update clip of 0.07. Remaining parameters (including those specifying the CG force-field) are identical to the REM settings described in section 3.4.1. ARCG-Hel was parameterized using the settings described for ARCG-REM with the several modifications: first, trees were limited to 100 leaves; second, 90K samples were used to calculate the gradient at each iteration; and third, 80 spline knots were used to characterize bonds and angle bonds. Attempts to parameterize ARCG-REM using the tree ensembles conditions present for ARCG-Hel resulted in unstable and oscillatory potentials. Attempts to parameterize ARCG-Hel using the conditions described for ARCG-REM are ongoing.

The pair potentials for ARCG-REM, ARCG-Hel, and REM may be found in appendix G.4. The potentials are qualitatively similar; however, minor variations in peak height and shape are present. It is notable that ARCG-Hel is even qualitatively close to the relative entropy based models as it optimizes a different divergence; it is, however, difficult to attribute the existing quantitative differences to the choice of divergence or the optimization and force-field settings.

4.5.6.2 VALIDATION

As discussed in chapter 3, validation of the configurational distribution for DDA is complex; as a result, the presented validation for models of DDA is more extensive than that given for water or methanol in previous sections. Pairwise statistics for the resulting DDA models are shown in Figs. 4.11, 4.12, 4.14, 4.14, and 4.13. Fig. 4.11 visualizes the distance between the terminal beads of the CG protein. Notably, the cutoff for nonbonded potentials is set to 20 Å, and a error in density is present at this point for all models considered. This appears to be due to the fact that the potential

does not naturally taper off at 20 Å (appendix G.4). ARCG-REM qualitatively matches the behavior of REM at this point, but exaggerates the cutoff related distortion. The overall density profile of the ARCG-REM model, however, closely matches that of the REM model. The ARCG-Hel model provides a qualitative match; however, striation in density is present.

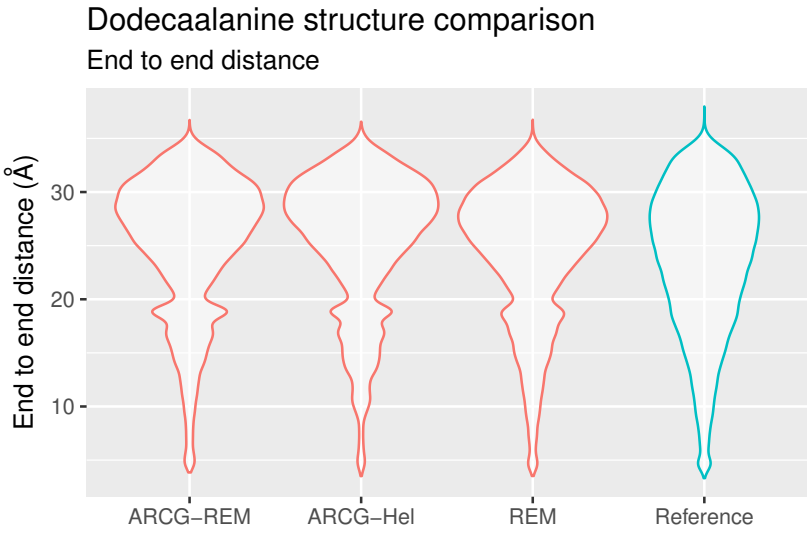

Figure 4.11: Violin plot of end to end distances for dodecaalanine models. REM-ARCG corresponds to using ARCG to estimate a relative entropy based model, REM-Hel corresponds to ARCG minimizing the Hellinger distance, and REM corresponds to direct REM; red violins are approximating the blue violin. Note that the ARCG-REM and ARCG-Hel models use differing parameters making comparison infeasible.

Fig. 4.12 visualizes all distances present in the protein, providing a quick qualitative understanding of the spacing present throughout DDA. Strong agreement is present across all models, with small differences found in the peak second from the bottom in the reference violin. This primarily corresponds to the distances present between every other CG bead (e.g., the distance between beads 4 and 6), and is shown directly in Fig. 4.13. Small differences are present in the ARCG-Hel distribution; despite these discrepancies strong overall agreement is again observed. Similar close agreement is seen for the distribution of distances between adjacent beads (Fig. 4.14). Small errors can be seen, however, in the bond distributions for the termini, as seen in Fig. 4.15. The source of the ARCG-REM model's deviation from the REM model's potential (appendix G.4) and correlation is unclear.

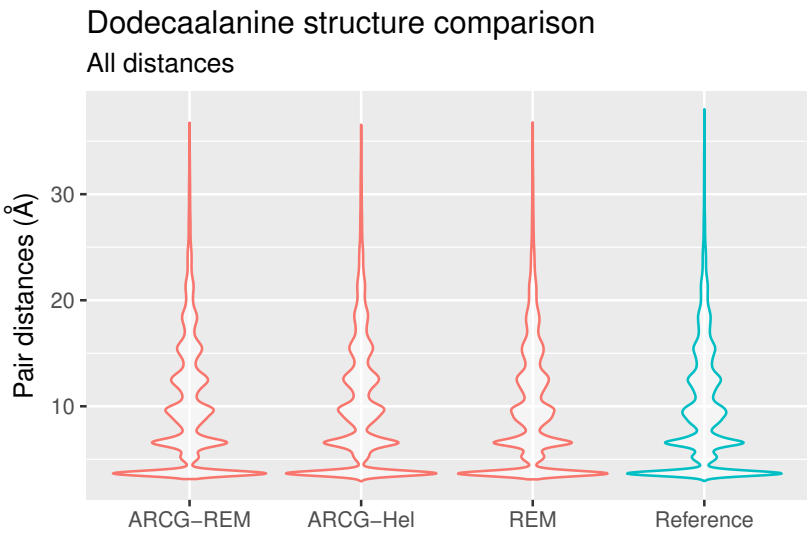

Figure 4.12: Violin plot of all pairwise distances for various dodecaalanine models. See Fig. 4.11 for abbreviation definitions.

The reported pairwise distributions provides some measure of accuracy of the proposed ARCG models. Unfortunately, as demonstrated in chapter 3, these low dimensional observables can belie issues in phase space overlap. Following the methodology (and hyperparameters) proposed in chapter 3, distributions of ΔW are presented in Fig. 4.16. Small differences between the ARCG and REM models are present, with the REM model possessing a longer tail in the projected model distribution. This longer tail corresponds to a larger mismatch as quantified by ΔW. Mean absolute SHAP (MAS) values are presented in tables 4.3 4.4, and 4.5. As discussed in section 3.5.1, error is primarily concentrated in the 1,2 and 11,12 distances. This trend is seen across all three models, although the balance between these two bonds differs, reflecting the variation seen in Fig. 4.15. SHAP CVs generated between the reference data and each model show similar patterns as implied in section 3.5.1, with heavy distortion primarily due to terminal bond errors (Figs. G.1, G.2, and G.3).

Collectively, these correlations and CVs imply that the ARCG parameterized models are highly similar to the traditionally parameterized REM model. Directly comparing the ARCG-REM and ARCG-Hel models to the REM model further quantifies these similarities; corresponding distributions of ΔW are found in Fig. 4.17. Strong overlap is present for both ARCG models, especially

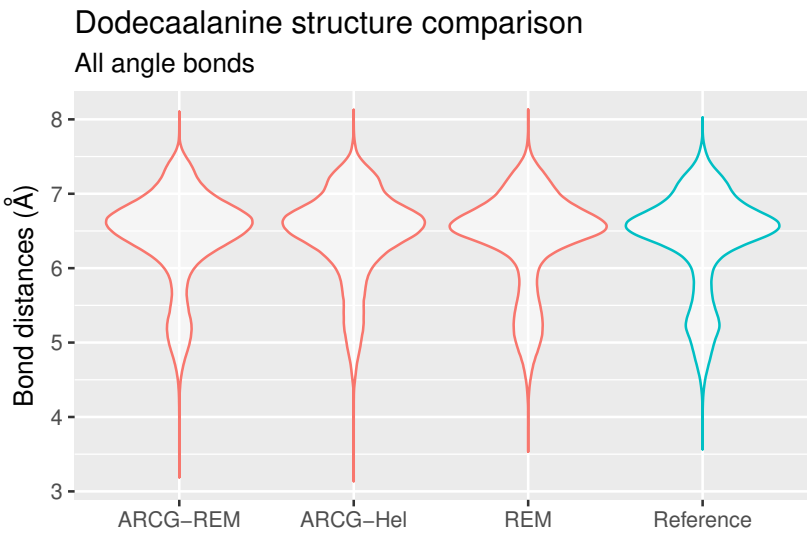

Figure 4.13: Violin plot of all angle bond distances for various dodecaalanine models. See Fig. 4.11 for abbreviation definitions.

Distance	11,12	1,2	1,3	1,5	8,12	10,12
MAS	1.06	0.58	0.30	0.24	0.23	0.21

Table 4.3: Top 6 Mean absolute SHAP (MAS) values for classification performed between the dodecaalanine REM model and the reference data. Note that the MAS are here presented in units of $k_b T$.

when compared to the magnitude of ΔW found in Fig 4.16. Examination of the corresponding MAS values (tables 4.6 and 4.7) provides additional insight. The deviations present in Fig. 4.15 are evident, with the ARCG-Hel model reporting somewhat high MAS values for the corresponding bonds, while the ARCG-REM model does not. As reinforced by Fig. 4.16, the models are not identical; unfortunately, the uncorrelated nature of the resulting errors is not clarified by the use of SHAP based CVs. Together, the results of this section imply high similarity between ARCG based models and those parameterized via traditional REM. This similarity persists, perhaps surprisingly, even when considering ARCG models minimizing the Hellinger distance. The variation between these various models is demonstrably lower than their deviation from the reference data. The exact manybody differences, however, remain difficult to summarize. Analogous results were found when using the Jenson-Shannon divergence as well multiscale coarse-graining (see appendix G.4): while small differences are present, the qualitative error made by all models is essentially the same.

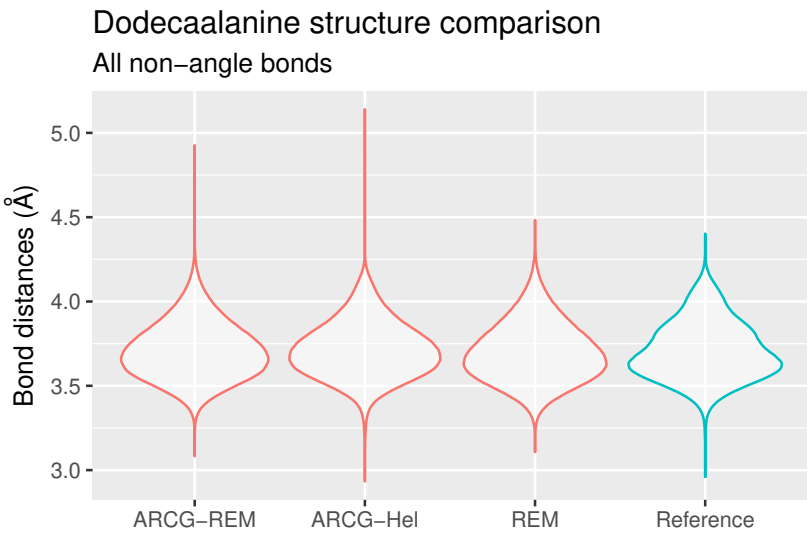

Figure 4.14: Violin plot of all non-angle bond distances for various dodecaalanine models. See Fig. 4.11 for abbreviation definitions.

Distance	11,12	1,2	1,3	7,10	10,12	8,11
MAS	0.73	0.70	0.30	0.21	0.21	0.20

Table 4.4: Top 6 mean absolute SHAP (MAS) values for classification performed between the dodecaalanine ARCG-REM model and the reference data. Note that the MAS are here presented in units of $k_b T$.

4.6 DISCUSSION

In previous sections we have described a broad new class of variational statements for optimizing CG models and derived methods for their optimization by utilizing the theory underpinning adversarial models in ML. Subsequently we have shown that it is possible to parameterize a CG model via ARCG at a coarser resolution than that native to the CG Hamiltonian. A clear application of ARCG is the parameterization of models that contain virtual sites; however, the CG distribution may be critiqued at any coarser resolution, providing the intriguing ability to control what aspects of a CG model are visible for optimization purposes. In the process of doing so we showed that gradients needed at each step of divergence minimization can be reformulated as modifying the system Hamiltonian to minimize the value of a specific observable, but that this observable depends on the distributions being considered at that step of optimization. We note that more generally the

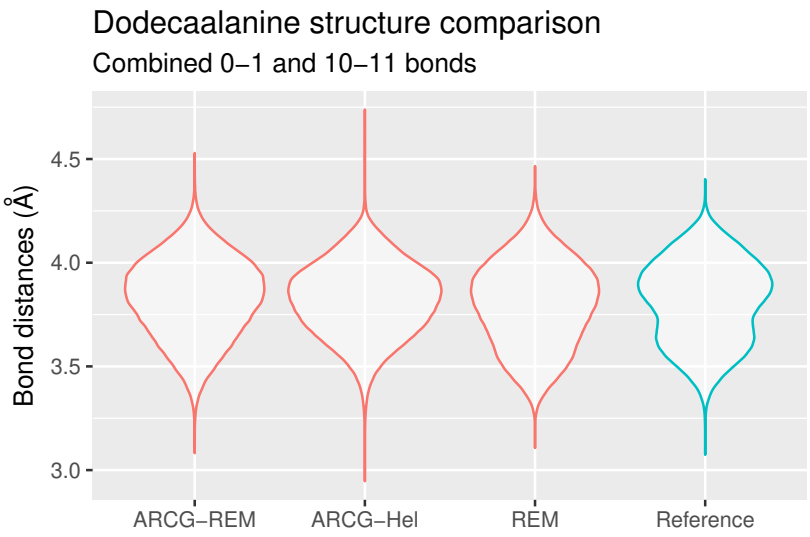

Figure 4.15: Violin plot of distances corresponding to the 1,2 and 11,12 (termini bonds) distances for various dodecaalanine models. See Fig. 4.11 for abbreviation definitions.

Distance	11,12	1,2	1,3	4,7	1,5	8,12
MAS	0.72	0.70	0.26	0.25	0.23	0.22

Table 4.5: Top 6 mean absolute SHAP (MAS) values for classification performed between the dodecaalanine ARCG-Hel model and the reference data. Note that the MAS are here presented in units of $k_b T$.

method presented can be used to calculate the KL divergence (and any of the other divergences discussed) between distributions for which no probability density/mass is known and for which one cannot be approximated via kernel density approximation or binning.

Beyond our central results we have provided work and discussion on two supporting topics.

1. We have provided comparisons to multiple contemporary methods for CG parameterization. In certain cases we have shown that divergences characterizing their configurational variational principles can be used in ARCG modeling. In one case we showed that a classifier based approach bears striking but not complete similarity to the presented approach. In the remaining cases we have discussed how decoupling the resolution at which we critique a model from the resolution of the CG Hamiltonian creates difficulties in said approaches.

2. We have provided a set of sufficient conditions for momentum consistency in the case of

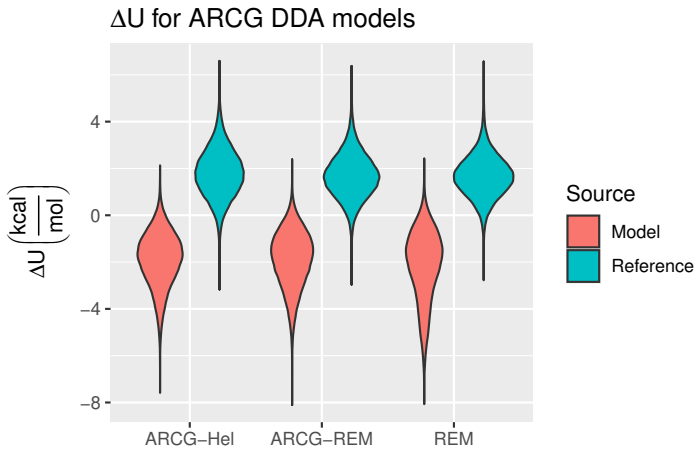

Figure 4.16: Violin plots of ΔW for various models of dodecaalanine divided along the reference and model ensembles. See Fig. 4.11 for abbreviation definitions.

virtual sites and described how these conditions may be extended. These are closely related to consistency requirements for traditional bottom-up CG models.

Additionally, we have provided numerical examples (and a computational implementation) for which we have optimized CG potentials to match specific distributions, some of which utilize CG virtual particles. Simple test models show quantitative agreement for calculated correlations. Virtual particle based models showed visual agreement and qualitative agreement in matching exact coefficients. Complex single molecule models (DDA) were analyzed using ΔW and showed a high degree of similarity relative to established REM models. Difficulties in convergence when virtual particles are present appeared to be either due to instability in the parameterization process or sloppiness in the model specifications. The manner in which this will affect multi-molecular systems is yet to be seen, but may present a significant challenge. It is clear that in the most general case parameter uniqueness with virtual particles is not guaranteed: if CG consistency can be obtained without virtual particles, then a model that can both decouple the virtual particle interaction from the real particles and modify the behavior of the virtual particles independently of said coupling will inherently be nonidentifiable. Additionally, it is likely that in the case of f-divergence based ARCG optimization that a relatively good initial hypothesis for the CG potential may be necessary, or significant amounts of noise must be added initially during optimization.

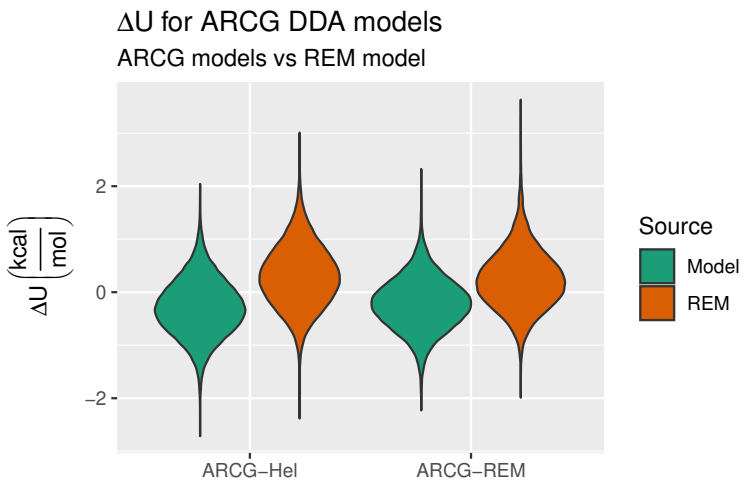

Figure 4.17: Violin plot of distances corresponding to the 1,2 and 11,12 (termini bonds) distances for various dodecaalanine models. See Fig. 4.11 for abbreviation definitions.

Distance	8,11	4,6	4,5	7,12	5,7	9,10
MAS	0.12	0.12	0.11	0.11	0.11	0.10

Table 4.6: Top 6 mean absolute SHAP (MAS) values for classification performed between the ARCG-REM and REM dodecaalanine models. Note that the MAS are here presented in units of $k_b T$.

Distance	11,12	1,2	8,10	4,6	5,7	7,9
MAS	0.19	0.19	0.15	0.15	0.14	0.14

Table 4.7: Top 6 mean absolute SHAP (MAS) values for classification performed between the ARCG-Hel and REM dodecaalanine models. Note that the MAS are here presented in units of $k_b T$.

There are multiple additional studies that could naturally expand and clarify the results presented.

1. The methods provided can be applied to approximate multi-molecule systems without virtual particles. This will require multiple steps: first, the proof-of-concept software framework presented will have to be expanded for larger system sizes. Second, the training method used will have to be developed such that it remains stable, whether through the systematic addition of noise or the use of enhanced sampling techniques. Third, the feature-space used to index Q will likely have to be correctly engineered based on knowledge of the FG and CG Hamiltonians. All three of these are tractable challenges.

2. The effect of using virtual particles should be investigated both computationally and theoretically, as previous analysis on incomplete basis sets (e.g., that on relative entropy and MS-CG [69]) does not apply transparently. In the process of doing so a better theoretical understanding of how to utilize these methods to capture specific higher order correlations in the training data should additionally be investigated, possibly leading to new ways in which bottom-up CG parameterization may be tuned to reproduce specific novel correlation functions.

3. The effect of various divergences on training approximate CG models should be further investigated theoretically and through simulation. This will facilitate the design of CG parameterization methods that have different biases in the approximations they produce when coupled with realistic CG potentials. This applies to not only to various f-divergences but also the wider set of divergences not heavily discussed in this article, such as the Wasserstein[190], Sobolev[210], Energy[188], and MMD[196] distances. The Wasserstein and Energy distances share the interesting property of taking into account the spatial organization of the domain of the probability distributions considered through a separate spatial metric. Combined with kinetically informed coordinate transforms such as TICA[211] and variants[212, 213], it may be possible to parameterize models to have stationary distributions that are kinetically close to one another.

4. The effect of an incomplete Q should be investigated. In this case the presented divergence based interpretation is not trivially accurate[214]. Understanding of how imperfect classifiers affect the parameterization of approximate models may have large implications on the optimization of complex multicomponent systems; overly expressive Q will likely impede model parameterization as more sampling of the CG and FG system may be required.

4.7 CONCLUDING REMARKS

In this chapter we discussed a new class of methods for the systematic bottom-up parameterization of a CG model. In doing so we illustrated concrete connections between CG models and algorithms such as generative adversarial networks. Utilizing these connections we both decoupled the resolution at which we critique our CG model from the CG potential itself and enabled the use of a variety of novel measures of quality for CG model parameterization. We provided a proof of concept implementation and several numerical examples. Additionally, we illustrated precise connections to several previous methods for CG model parameterization. Finally, we noted multiple future branches of studies that can now be pursued. Together, these results open a new conceptual basis for future systematic CG parameterization strategies.

CHAPTER 5

LOOKING FORWARD

This document provides connections between generative adversarial networks and coarse-grained molecular dynamics (CG MD) force-field parameterization. These connections establish that classification is a viable tool for the sample based comparison of two high dimensional free energy surfaces. This development facilitated the creation of new error characterization algorithms as well as new strategies for parameterizing CG force-fields. In retrospect, these connections are simple: the ergodic nature of MD, along with the explicit energy function, allows for straightforward application of the probabilistic tools of machine learning. This connection implies that generative adversarial networks perform similar tasks as MD and Markov Chain Monte Carlo and that adversarial training can be applied to force-field development. There do, however, exist differences, such as the availability of derivatives and objective function used.

Unfortunately, CG force-fields are challenging to validate. It is difficult to quantify the difference of adversarially trained CG models relative to their traditional counterparts: the intrinsically manybody distributions present in most CG models resist complete analysis. More generally this lack of validation seems to be at odds with the promise of quantitatively accurate CG models. As reference quantum mechanical calculations become more feasible, the validation of high fidelity machine-learning based atomistic force-fields has continued to increase significantly[106, 160], making clear the distance between quantitative CG models and their atomistic counterparts. Improving error quantification is critical for both improving confidence in the predictive ability of existing algorithms as well as justifying the application of more complex CG force-field strategies such as adversarial learning. We note that in certain cases qualitative errors allow creative

approaches (e.g., virtual sites in the case of of lipid bilayers[98, 215]) to be applied in the absence of clear high dimensional validation metrics, but that the ease of quantifying improvement is directly related to the poor quality of the models. Adversarial learning has not yet been applied to these cases but may result in models with more substantial differences than those observed in chapter 4.

Chapter 3 provides promising initial insight into this problem of force-field validation: the approaches central to force-field development can be used for estimating quantitative error through the ΔW. Indeed, it was variationally shown through this lens that the current selection of adversarially trained models do not significantly differ in distribution from those parameterized using traditional methods, reinforcing that force-field bases (and not training strategy) remain the dominating factor in model behavior. When combined with modern force-field representations (such as those in references [106, 160]) ΔW therefore seems to be promising option for quantitative validation. The more aggressive proposal of using existing methods from explainable artificial intelligence remains to be fully explored, but provides a novel direction for future research. The utility of these error characterization techniques will only be evident if CG models are reasonably accurate: classification will provide little insight to a CG lipid model if it is unable to form a coherent bilayer. As CG force-fields become more sophisticated, however, the observed errors will likely become less qualitative.

With improved error quantification in hand many future studies can designed. Due in part to the rising popularity of GANs, the number of different training strategies that can be implemented using the variational residual central to ARCG is large and rapidly increasing, providing a virtually unbounded set of possible training metrics (Q). When combined with improved error quantification, these various metrics, along with the possibilities provided by virtual particles and higher order potentials, provide the basis for a multitude of studies. For example:

1. CG force-fields can be optimized to minimize various metrics found to be effective in the GAN community. For example, a CG force-field representing a lipid bilayer could be optimized to minimize a variational estimate of the Wasserstein metric as proposed in [190]. While the novel metrics described in this work (Jensen-Shannon divergence and Hellinger

distance) did not result in notably different models, this does not preclude that other metrics may have more notable affects on CG potentials. This direction can be expanded to include other energy based model strategies; see reference [216].

2. Virtual particles provide a novel way to increase the flexibility of CG models. This approach has been explored in a preliminary manner in studies such as [98]; however, the work presented allows these studies to be connected to rigorous variational principles through both adversarial training and the modified relative entropy gradient approach described in chapter 4. This new approach provides a strong foundation for future studies exploring the design of systems with virtual particles.

3. The capability to parameterize systems with virtual particles arises from the flexibility afforded by $\mathcal{G}$. This same flexibility can also be used to tune systematic coarse-graining to focus on specific scales. For example, a filament of actin monomers (each of which is here described at the twelve site resolution) can be directly optimized to reproduce filament-scale behavior (one site per monomer). Optimization at this reduced resolution can additionally be linearly combined with optimization at the full twelve site resolution to fine tune the behavior produced by specific bottom up algorithms.

4. The classification based approach in chapter 3 can be applied to understand the impact of CG mapping resolution. For example, if maps are constructed for globular actin at various resolutions but mapped to the same resolution for comparison, the sites which sites are critical to configurational accuracy can be determined in a systematic manner. This approach can be used to reframe the issue of CG resolution, although the combinatorial complexity of map space remains a challenge.

5. The use of techniques from explainable artificial intelligence provides an interesting avenue for further development. If the pertinent characterization of error is provided by ΔW, any explainable or interpretable classification algorithm represents a candidate for understanding the high dimensional behavior intrinsic to CG models. For example, the variational estimates

provided by a non-parametric model can be compared to those obtained through feature engineering and an intrinsically interpretable model such as a rule list or logistic regression, each of which would provide novel ways to describe the structure of ΔW.

Additional studies which expand on and combine these proposals are also possible. The primary question is which of these approaches is mostly likely to provide immediate improvement relative to existing methods. This, unfortunately, is less clear, and likely depends on design choices not touched upon in this document. For example, which of the many possible GAN metrics will be valuable for modeling a CG bilayer? Which representation of virtual particle will improve model results? These choices have yet to be determined.

The parallels provided with generative models and MD suggest the generation of atomistic (and CG) configurations using methods such as GANs. Indeed, work along these lines is already underway. A particularly promising direction is that of Boltzmann generators[89], as their training procedure is accelerated relative to simple likelihood based methods. Approaches such as these are a promising alternative to costly simulations such as MD. However, it remains unclear to me how these generative machines can be constructed such that their transferability based errors will be intuitive. CG models, on the other hand, have the advantage of being the product of relatively transparent potentials, and furthermore seem to retain more interpetability. The way in which these methods will complement one another remains to be seen.

In summary, the number of approaches available for bottom-up CG is significant and growing. However, the increase in modeling possibilities, along with the growth in available data, only adds to the need for effective and shared validation metrics. Together, complex models and effective validation strategies suggest a promising future to bottom-up CG modeling. However, much work remains to be done before CG models achieve the reliability of their atomistic counterparts. Hopefully, the analysis provided here can stimulate novel work in these areas.

APPENDIX A

SUPPLEMENTARY ACTIN VISUALIZATIONS

This appendix contains supplemental visualizations from section 3.5.2. Fig. A.1 refers to monomeric 12 site ADP actin, whereas Figs. A.2, A.3, and A.4 as well as table A.1 refer to the comparison between ATP and ATP actin at the 12 site resolution. Fig. A.5 refers to the actin filament.

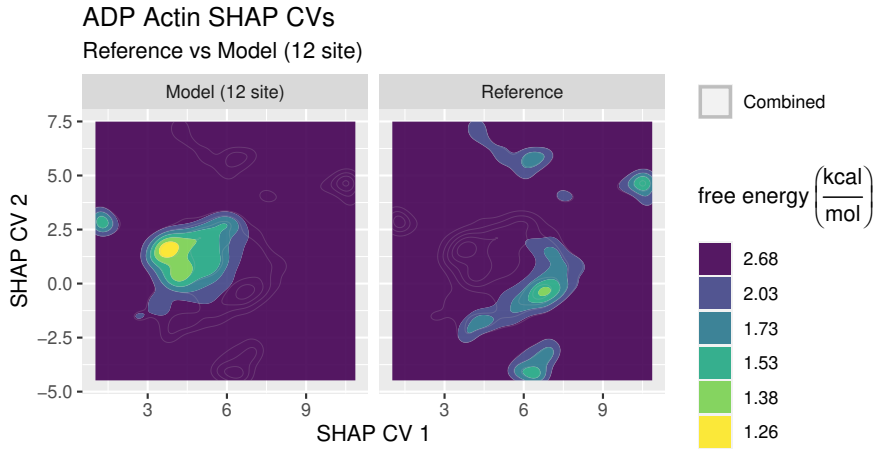

Figure A.1: The free energy surface given by the SHAP CVs calculated by comparing the 12 site elastic network to the reference ensemble at a 12 site resolution.

SHAP CVs for ADP actin at the 12 site resolution are found in Fig. A.1. The subbasins present at $(1.5, 2.5)$ and $(11, 5)$ are due to the distance between sites 1 and 9. The subbasin at $(6, 5)$ is attributable to the distance between sites 2 and 4. The subbasin at $(6, -3)$, along with the broadness of the central basin, correspond to relatively uncorrelated heterogeneity occurring between site 5 and the rest of the protein, along with variation in the position of site 9.

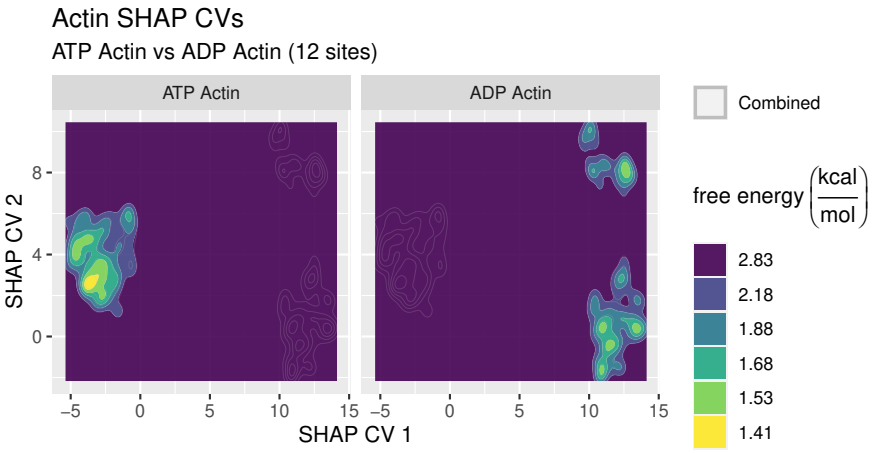

Figure A.2: Free energy surfaces produced along the SHAP variables generated by comparing ATP to ADP actin at the 12 site resolution.

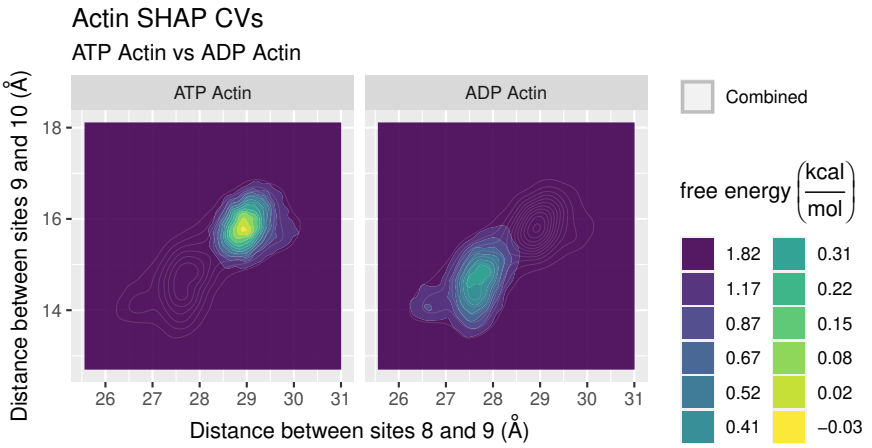

Figure A.3: Free energy surfaces produced along the 8,9 and 9,11 distances for the ATP and ADP actin ensembles.

Distance	8,9	9,11	2,5	1,9	1,2	7,12	3,9
MAS	5.99	3.78	2.3	2.21	2.02	1.66	1.63

Table A.1: The top 7 mean absolute SHAP (MAS) values found when comparing ATP actin to ADP actin at the 12 site resolution. Distance refers to the indices of the two sites between which the distance is defined.Note that that the top 4 values contain either the nucleotide (site 9) or the D-loop (site 8). Note that the MAS are here presented in units of $k_b T$.

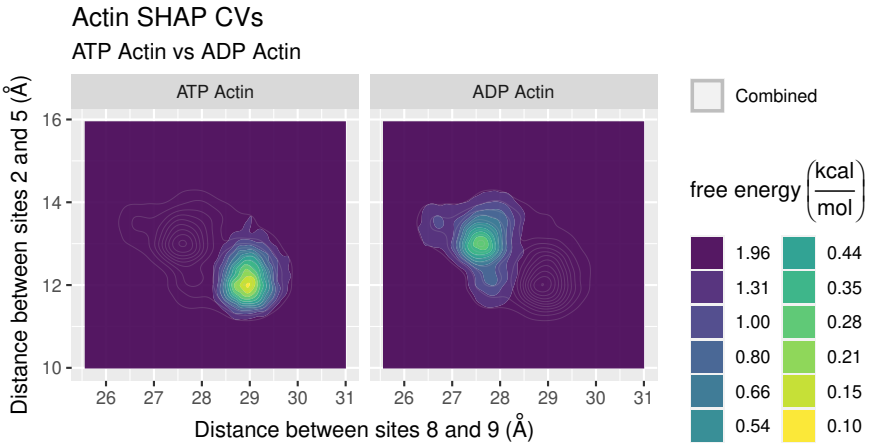

Figure A.4: Free energy surfaces produced along the 8,9 and 2,5 distances for the ATP and ADP actin ensembles.

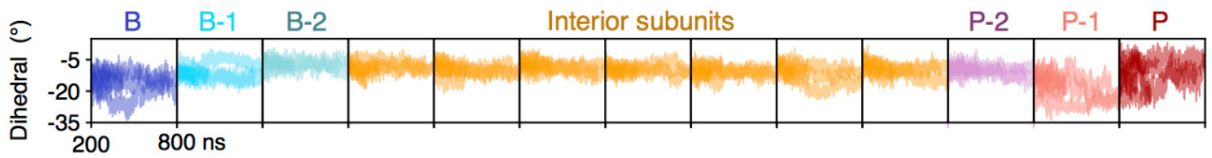

Figure A.5: Dihedral trajectories divided by unit observed over 400 *ns* of atomistic simulation of an actin filament. Note that the barbed end (denoted *B*) and the pointed end (denoted *P*) differ from the internal units in terms of dihedral distribution; however, the internal units seem to be homogeneous. Multiple actin filaments, including those in the ATP, ADP, and ADP-Pi state were used to create this figure. Reproduced from reference [150] with permission.

APPENDIX B

ATP ACTIN SHAP CVS

Section 3.5.2 presented SHAP based analysis on ADP actin. Here, we provide corresponding results for ATP actin, which contains a different nucleotide in site 9. Similar, but not identical results, are found. Beginning with results analogous to those presented in Fig. 3.4, Fig. B.1 provides a distributions projected along ΔW. Similar patterns are observed: the 12 site model exhibits a larger spread until being mapped to the 4 site resolution. We next analyze the 12 site ATP model similarly

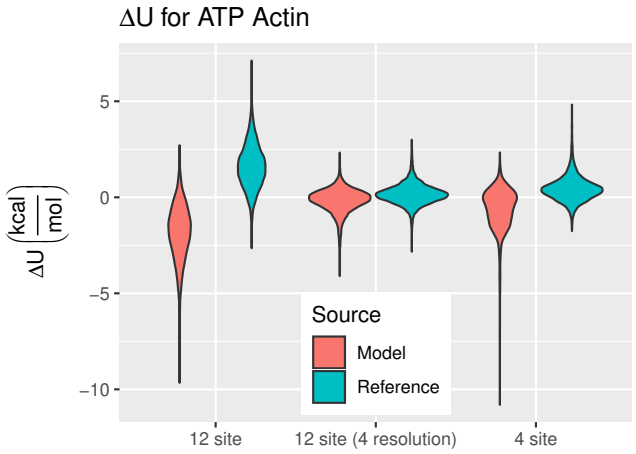

Figure B.1: Violin plots of ΔW for multiple models and resolutions of ATP actin, divided along the reference and model ensembles. The "12 site (4 resolution)" and "4 site" models are compared to the reference at the 4 site resolution, while "12 site" is compared at the 12 site resolution. Note that varying models/resolutions change the form of ΔW, which changes the shape of the projected reference distribution.

to Fig. A.1; this is seen in Fig. B.2. Again, the largest sources of error include the D-loop (site 5) and the nucleotide (site 9): this is seen in table B.1. The elongated vertical aspect of Fig. B.2 is due to the 2,5 distance; horizontal spread is primarily due to the 2,4 distance, with additional

effects from the 5,9 distance. The variety of isolated islands present are due to hard mismatches in distances such as the 5,8 and 2,10 distances. Current experience suggests that isolated islands are generally primarily due to hard mismatches in overlap at extreme distances; diffuse large scale features are often due to persistent but more complex manybody errors.

Fig. B.3 shows results analogous to Fig. 3.11. Horizontal spread in Fig B.3 can be attributed to the 1,4 distance, 2,4 distance, and 2,3 distance. We here provide figures that led to this conclusion: Figs. B.4, B.5, B.6, and B.7. These figures are generated by searching features with high MAS values; this strategy was used to find the variables of and create corresponding plots in this work. Note that these figures display SHAP values and not physical distances. The island in the top left of Fig. B.3 is attributable to error in the 1,3 distance (Fig. B.6). Similarly to the results seen in Fig. 3.12, Fig. B.8 visualizes the projection of the 12 site model onto the SHAP CVs generated when comparing the 4 site model to the reference ensemble. The improvement is even more stark compared to that observed in Fig. 3.12, with the 12 site model appearing to remove much of the error present in the 4 site model. Differences are still seen in the top left island, and errors are still present as implied by Fig. 3.10: ΔW's interpretation as pointwise error still implies that the tail distributions present differ in thickness between the model and reference. The large number of involved variables makes it difficult to efficiently discern the physical variables underlying the increase in accuracy. However, direct comparison between the mapped 12 site and 4 site models implies the importance of the 2,3 and 2,4 distances, as seen in Fig. B.9. Despite the noted patterns, the SHAP based analysis above is more opaque that than that in chapter 3. This seems to be due to more manybody error that is not as focused in a small number of pair distances. We note that this error is still reported in ΔW; as a result, other dimensional reduction techniques may be more applicable.

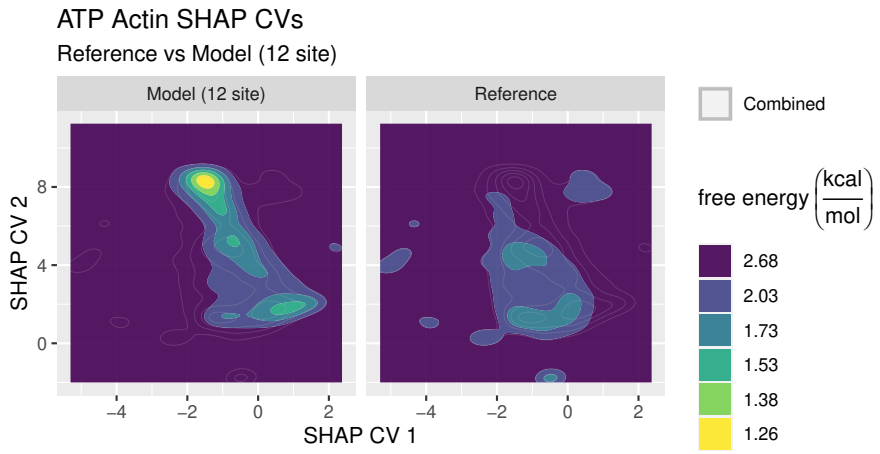

Figure B.2: The free energy surface given by the SHAP CVs calculated by comparing the 12 site elastic network to the reference ensemble at a 12 site resolution.

Distance	2,5	2,4	5,9	1,9	5,8	2,10	8,11
MAS	1.1	0.44	0.37	0.32	0.29	0.26	0.24

Table B.1: Top 6 mean absolute SHAP (MAS) values from comparing the 12 site actin model to reference data. Note that site 5 is the D-loop and site 9 is the nucleotide. Note that the MAS are here presented in units of $k_b T$.

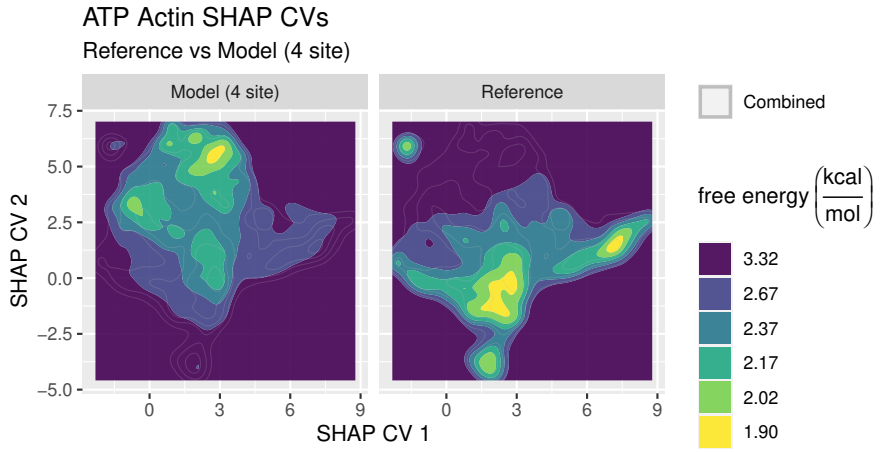

Figure B.3: Free energy surface for ATP actin produced along the SHAP variables generated by comparing a 4 site elastic network model to mapped reference data.

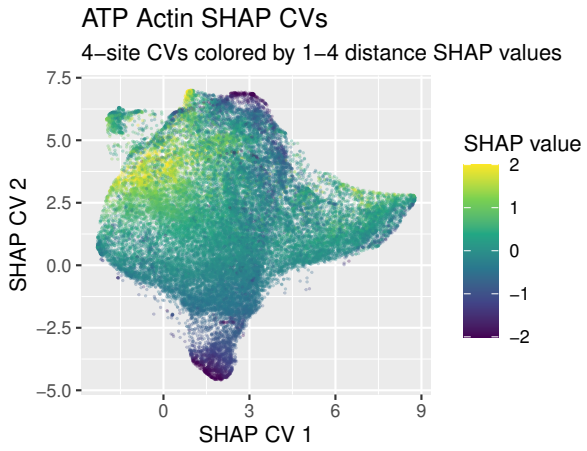

Figure B.4: Samples for ATP actin plotted along the SHAP variables in Fig. B.3 and color coded by SHAP value for the 1,4 distance. Note that the SHAP values have been capped to the range $[-2, 2]$ to increase contrast; larger absolute SHAP values are found in the extremes.

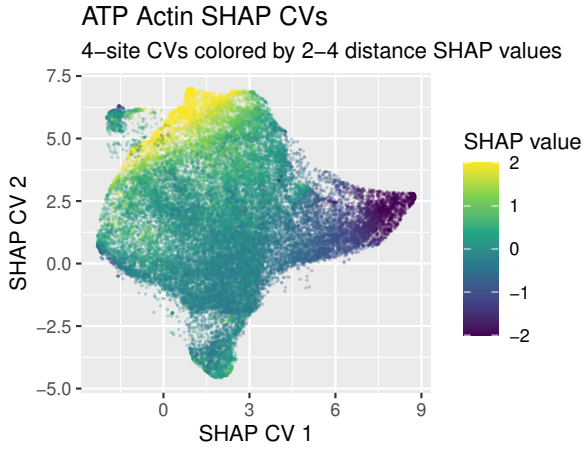

Figure B.5: Samples for ATP actin plotted along the SHAP variables in Fig. B.3 and color coded by SHAP value for the 2,4 distance. Note that the SHAP values have been capped to the range $[-2, 2]$ to increase contrast; larger absolute SHAP values are found in the extremes.

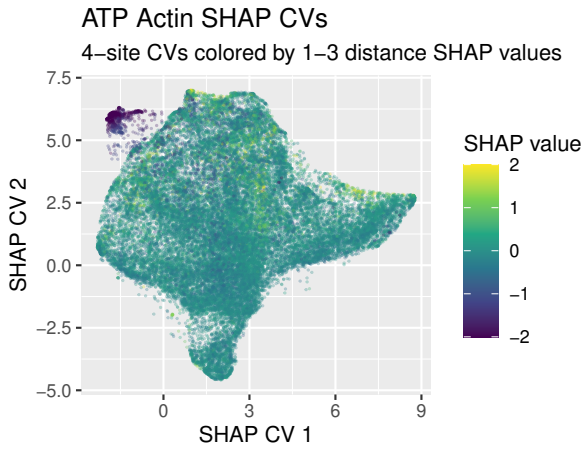

Figure B.6: Samples for ATP actin plotted along the SHAP variables in Fig. B.3 and color coded by SHAP value for the 1,3 distance. Note that the SHAP values have been capped to the range $[-2, 2]$ to increase contrast; larger absolute SHAP values are found in the extremes.

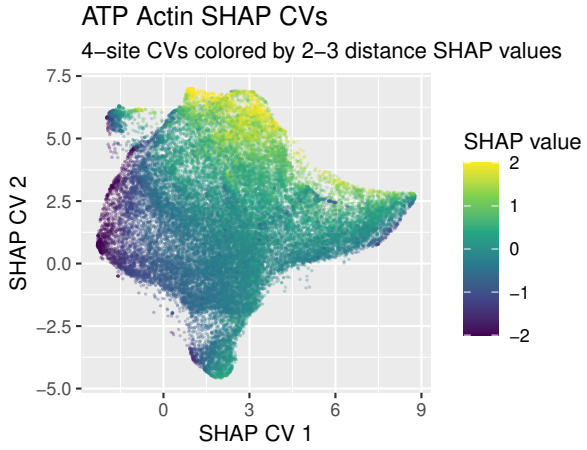

Figure B.7: Samples for ATP actin plotted along the SHAP variables in Fig. B.3 and color coded by SHAP value for the 2,3 distance. Note that the SHAP values have been capped to the range $[-2, 2]$ to increase contrast; larger absolute SHAP values are found in the extremes.

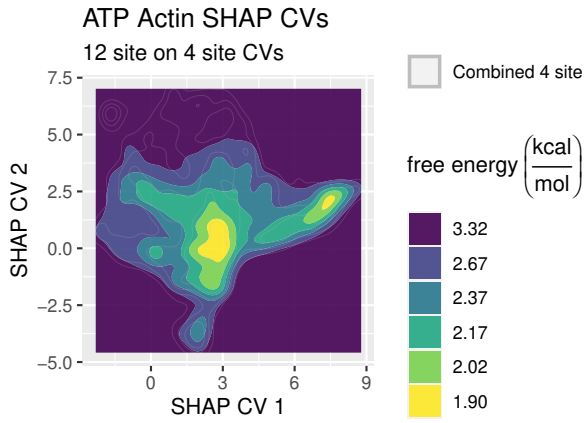

Figure B.8: 12 site free energy surface produced along the SHAP variables generated by comparing a 4 site elastic network model to mapped ATP actin reference data. Filled surfaces represent the 12 site model ensemble, while grey lines represent the combined 4 site model and reference ensembles presented in Fig. B.3.

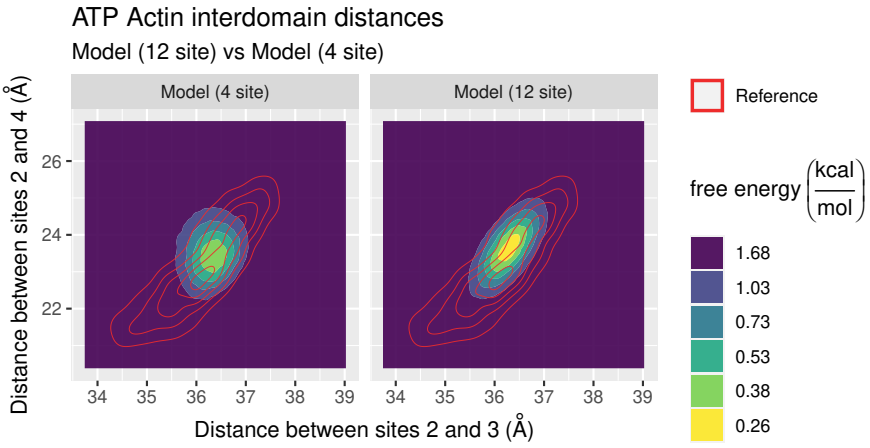

Figure B.9: Free energy surface for ATP actin produced the distance between sites 2 and 4 and the distance between sites 2 and 3. Filled contour panels represent the two model ensembles, while the red line contour overlay correspond to statistics from the reference ensemble.

APPENDIX C

ENVELOPE THEOREM

The envelope theorem is used to justify optimizing θ using only the partials calculated holding the optimal observable constant. A general statement of the envelope theorem is given by theorem 1 in Milgrom & Segal [194], which also notes that the theorem applies to directional derivatives in a normed vector space.

Let X denote the choice set and let the relevant parameter be $t \in [0,1]$. Let $f : X \times [0,1] \to \mathbf{R}$ denote the parameterized objective function. The value function V and the optimal choice correspondence X^* are given by:

$$V(t) := \sup_{x \in X} f(x,t) \tag{C.1}$$

$$X^*(t) := \{x \in X : f(x,t) = V(t)\} \tag{C.2}$$

Take $t \in (0,1)$ and $x^* \in X^*(t)$, and suppose that $\partial_t f(x^*,t)$ exists. If V is differentiable at t, then $V'(t) = \partial_t f(x^*,t)$.

This result puts no constraint on X, which corresponds to $\mathcal{Q}$ in the current work, except that its maximal member have a derivative at that point. Additionally, as noted in Milgrom & Segal [194], this result is only useful if V is known to be differentiable. This is compatible with our f-divergence variational statement when considered in the context of a complete $\mathcal{Q}$ and population averages, but must in general be confirmed for each choice of $\mathcal{Q}$. In situations where a closed form expression corresponding to the maximum is not known, constraints may be put on each member of $\mathcal{Q}$ to ensure applicability. Suitable constraints may be found in the remainder of Milgrom & Segal [194].

APPENDIX D

ARCG MOMENTUM CONSISTENCY

The described approach to achieve momentum consistency requires that we put more specific constraints on $\mathcal{G}$. This is needed due to our minimal strategy for providing sufficiency conditions for consistency: primarily, we utilize arguments in previous work to provide sufficient constraints. The resulting conditions given suffice for the case of virtual particles that are simply dropped from the system by $\mathcal{G}$. Generalizations to linear mappings that share particles between sites can additionally be inferred. First we discuss the approach of previous work on momentum consistency as is relevant to our work, and then concisely give a route to momentum consistency.

D.1 MS-CG

Generally, we will here assume that $\mathcal{M}_{\mathbf{r}}$ satisfies specific properties. Once $\mathcal{M}_{\mathbf{r}}$ is defined, we construct an appropriate $\mathcal{M}_{\mathbf{p}}$. First, $\mathcal{M}_{\mathbf{r}}$ must be expressible in the following linear form, where $\mathcal{M}_{\mathbf{r}I}$ denotes the Ith particle entry of the output of $\mathcal{M}_{\mathbf{r}}$, i iterates over the particles contribute to site I, and c denotes positive constants.

$$\mathcal{M}_{\mathbf{r}I}(r^n) := \sum_{i=1}^{n_I^{\mathcal{M}}} c_{Ii}^{\mathcal{M}_{\mathbf{r}}} r_i \tag{D.1}$$

As in MS-CG[45], we impose translational consistency.

$$\sum_{i=1}^{n_I^{\mathcal{M}}} c_{Ii}^{\mathcal{M}_{\mathbf{r}}} = 1 \tag{D.2}$$

From this we allow $\mathcal{M}_{\mathbf{r}}$ to imply $\mathcal{M}_{\mathbf{p}}$ up to the factor of the CG masses $\{M_I\}_I$ as stated in MS-CG.

$$\mathcal{M}_{\mathbf{p}_I}(\mathbf{p}^n) := M_I \sum_{i=1}^{n_I^{\mathcal{M}}} \frac{c_{Ii}\mathbf{p}_i}{m_i} \tag{D.3}$$

As before, this type of map transforms global consistency into constituent momentum and position space components, i.e.,

$$\begin{aligned} p_{\text{mod}}(\mathbf{R}^{3N},\mathbf{P}^{3N}) = \int \mathrm{d}\mathbf{r}^{3n} \int \mathrm{d}\mathbf{p}^{3n} p_{\text{mod}}^{\text{pre}}(\mathbf{r}^{3n},\mathbf{p}^{3n}) \delta(\mathcal{M}_{\mathbf{r}}(\mathbf{r}^{3n}) - \mathbf{R}^{3N}) \delta(\mathcal{M}_{\mathbf{p}}(\mathbf{p}^{3n}) - \mathbf{P}^{3N}) \\ = p_{\text{mod},\mathbf{R}}(\mathbf{R}^{3N}) p_{\text{mod},\mathbf{P}}(\mathbf{P}^{3N}) \end{aligned} \tag{D.4}$$

where the vector valued delta functions are understood to be products of scalar delta functions. If $\mathcal{M}$ does not associate any individual atoms to more than a single CG site, then

$$\exp\left(-\beta \sum_{I=1}^{N} \frac{\mathbf{P}_I^2}{2M_I}\right) \propto \int \mathrm{d}\mathbf{p}^{3n} \exp\left(-\beta \sum_{i=1}^{n} \frac{\mathbf{p}_i^2}{2m_i}\right) \times \delta(\mathcal{M}_{\mathbf{p}}(\mathbf{p}^{3n}) - \mathbf{P}^{3N}) \tag{D.5}$$

with

$$M_I^{\mathcal{M}^{-1}} := \sum_{i \in I_I} \frac{{c_{Ii}^{\mathcal{M}}}^2}{m_i} \tag{D.6}$$

We will additionally assume that analogous constraints are put on $\mathcal{G}_r$ when considering momentum consistency below.

D.2 MOMENTUM CONSISTENCY

Using these points we now move forward directly discussing momentum consistency. As stated previously, by constraining $\mathcal{G}$ and $\mathcal{M}$ as above, and assuming the underlying systems are characterized by separable probability densities, we find

$$p_{\text{mod}}(\mathbf{R}^{3N},\mathbf{P}^{3N}) = p_{\text{mod},\mathbf{R}}(\mathbf{R}^{3N}) p_{\text{mod},\mathbf{P}}(\mathbf{P}^{3N}) \tag{D.7}$$

$$p_{\text{ref}}(\mathbf{R}^{3N},\mathbf{P}^{3N}) = p_{\text{ref},\mathbf{R}}(\mathbf{R}^{3N}) p_{\text{ref},\mathbf{P}}(\mathbf{P}^{3N}) \tag{D.8}$$

As a result, we split up our consistency statement (omitting arguments for clarity)

$$(p_{\text{mod},\mathbf{R}} = p_{\text{ref},\mathbf{R}} \wedge p_{\text{mod},\mathbf{P}} = p_{\text{ref},\mathbf{P}}) \implies p_{\text{mod}} = p_{\text{ref}} \tag{D.9}$$

Configurational consistency is handled via divergence matching as described in the main article; we here consider momentum consistency algebraically.

$$\begin{aligned} p_{\text{mod},\mathbf{P}} = p_{\text{ref},\mathbf{P}} \iff & \int \mathrm{d}p^{3\nu} \exp\left(-\beta \sum_{i=1}^{\nu} \frac{p_i^2}{2m_i}\right) \delta(\mathcal{G}_p(p^{3\nu}) - \mathbf{P}^{3N}) \\ & \propto \int \mathrm{d}\mathbf{p}^{3n} \exp\left(-\beta \sum_{i=1}^{n} \frac{\mathbf{p}_i^2}{2m_i}\right) \delta(\mathcal{M}_{\mathbf{p}}(\mathbf{p}^{3n}) - \mathbf{P}^{3N}). \end{aligned} \tag{D.10}$$

We substitute these using two sets of properly designed CG masses, each set implied by a mapping operator and the masses at resolution it maps

$$\exp\left(-\beta \sum_{I=1}^{N} \frac{\mathbf{P}_I^2}{2M_I^{\mathcal{G}}}\right) \propto \exp\left(-\beta \sum_{I=1}^{N} \frac{\mathbf{P}_I^2}{2M_I^{\mathcal{M}}}\right) \tag{D.11}$$

$$M_I^{\mathcal{G}^{-1}} := \sum_{i \in I_I^{\mathcal{G}}} \frac{c_{Ii}^{\mathcal{G}^2}}{m_i} \tag{D.12}$$

$$M_I^{\mathcal{M}^{-1}} := \sum_{i \in I_I^{\mathcal{M}}} \frac{c_{Ii}^{\mathcal{M}^2}}{m_i} \tag{D.13}$$

The only solution is to set $M_I^{\mathcal{G}} = M_I^{\mathcal{M}}$ for each CG site I; in this case find a set of equations implying consistency.

$$\left[0 = \sum_{i \in I_I^{\mathcal{M}}} \frac{c_{Ii}^{\mathcal{M}^2}}{m_i} - \sum_{i \in I_I^{\mathcal{G}}} \frac{c_{Ii}^{\mathcal{G}^2}}{m_i}\right] \forall \text{ CG sites } I \tag{D.14}$$

Note that these equations are still subject to the aforementioned constraints (positivity, etc.). This provides a simple condition connecting our FG masses, pre-CG masses, $\mathcal{M}$, and $\mathcal{G}$, and allows one to check for momentum consistency.

When considering a CG model with no pre-CG resolution the FG mapping $\mathcal{M}$ must associate each atom with at most a single CG site in order for the mapped momentum distribution to factor-

ize with respect to each CG site. This is required for momentum consistency if the CG model is simulated using traditional molecular dynamics software as the momentum distribution produced by traditional molecular dynamics is necessarily factorizable. This same constraint to $\mathcal{M}$ and $\mathcal{G}$ is assumed in the preceding analysis, but this is not generally required for ARCG models as nonfactorizable momentum distributions may be produced by both $\mathcal{M}$ and $\mathcal{G}$. However, the analysis provided to illustrate momentum consistency is based on a generalizable strategy: previous approaches to momentum consistency which produced a closed form expression for a function proportional to the Boltzmann density of the mapped atomistic distribution may be extended to the current setting by simply calculating the density implied by both $\mathcal{M}$ and $\mathcal{G}$ and setting them to be equivalent. In this way, more sophisticated approaches such as the one in Han *et al.* [217] may be applied analogously to approach more complex mapping operators.

APPENDIX E

ARCG LOSS DERIVATIONS

The basis of the duality central to f-divergences is translated from theorem 9 in Reid & Williamson [81]. The equations relating loss functions l from the combined loss $\underline{L}$ may be confirmed via algebra after the two following identities are noted, both of which may be found in Reid & Williamson [81].

$$\left.\frac{\partial \underline{L}}{\partial x}\right|_h = l_{\text{ref}}(h) - l_{\text{mod}}(h) \tag{E.1}$$

$$\underline{L}(h) = (1-h)l_{\text{mod}}(h) + hl_{\text{ref}}(h) \tag{E.2}$$

The terms needed to define l_{ref} and l_{mod} are given as follows. First, note that the function generating the appropriate relative entropy is $x\log x$ (not $\log x$). From this we find (only in the case of relative entropy)

$$\underline{L}(h) = -2h\log\frac{h}{1-h} \tag{E.3}$$

and

$$\left.\frac{\partial \underline{L}}{\partial x}\right|_h = -2\left(\log\frac{h}{1-h} + \frac{1}{1-h}\right). \tag{E.4}$$

Through substitution we then arrive at Eq. (4.24). A similar procedure may be used to emulate other f-divergences.

APPENDIX F

RELATIVE ENTROPY VIRTUAL PARTICLE DERIVATIVE

In this section we derive the regression based approach for performing REM on systems with virtual particles. Starting from the Eq. (4.33), we assume we have found the ideal η.

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\theta_i}\mathcal{F}^{\mathrm{RE}}\left[p^{\mathrm{pre}}_{\mathrm{mod},r,\theta}, p_{\mathrm{ref},\mathbf{R}}; \mathcal{G}\right] = &-\beta\left\langle \frac{\eta\circ\mathcal{G}}{1-\eta\circ\mathcal{G}}\right\rangle_{p^{\mathrm{pre}}_{\mathrm{mod},r,\theta}}\left\langle \frac{\partial U^{\mathrm{pre}}_{\mathrm{mod}\theta}}{\partial\theta_i}\right\rangle_{p^{\mathrm{pre}}_{\mathrm{mod},r,\theta}}\\ &+\beta\left\langle\left(\frac{\eta\circ\mathcal{G}}{1-\eta\circ\mathcal{G}}\right)\frac{\partial U^{\mathrm{pre}}_{\mathrm{mod},\theta}}{\partial\theta_i}\right\rangle_{p^{\mathrm{pre}}_{\mathrm{mod},r,\theta}}\end{aligned} \tag{F.1}$$

We deal with these terms individually. First,

$$\left\langle \frac{\eta\circ\mathcal{G}}{1-\eta\circ\mathcal{G}}\right\rangle_{p^{\mathrm{pre}}_{\mathrm{mod},r,\theta}} \tag{F.2}$$

By law of the unconscious statistician this is 1. This is shown here assuming that $\mathcal{G}$ can be written as a subset of a valid change of coordinates, which we denote $\mathcal{G}^{\mathrm{full}}$. We denote the output of $\mathcal{G}$ g, the output of $\mathcal{G}'$ g', the output of $\mathcal{G}^{\mathrm{full}}$ $[g,g']$, and use J to denote the absolute value of the determinant of the appropriate Jacobian. Additionally, as these new variables have densities, we use f_{mod} to correspond to the joint density over g and g', and $f_{\mathrm{mod}}(\cdot|g')$ and $f_{\mathrm{mod}}(\cdot|g)$ to correspond to the conditional densities. We note that in the case of typical virtual particles, no changes of coordinates is necessary, and in the case of a linear change in coordinates with linearly independent variables,

$\mathcal{G}^{\text{full}}$ can be defined such that the absolute value of the determinant of its Jacobian is 1.

$$
\begin{aligned}
&\left\langle \frac{\eta \circ \mathcal{G}}{1-\eta \circ \mathcal{G}} \right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} \\
=&\int_{\chi} \frac{\eta \circ \mathcal{G}(x)}{1-\eta \circ \mathcal{G}(x)} p^{\text{pre}}_{\text{mod},r,\theta}(x)\mathrm{d}x \\
=&\int_{\mathcal{G}^{\text{full}}(\chi)} \frac{\eta(g)}{1-\eta(g)} p^{\text{pre}}_{\text{mod},r,\theta}(x(g,g'))J(g,g')\mathrm{d}g\mathrm{d}g' \\
=&\int_{\mathcal{G}(\chi)} \frac{\eta(g)}{1-\eta(g)} \left[\int_{\mathcal{G}'(\chi)} p^{\text{pre}}_{\text{mod},r,\theta}(x(g,g'))J(g,g')\mathrm{d}g'\right] \mathrm{d}g \\
=&\int_{\mathcal{G}(\chi)} \frac{p_{\text{ref},\mathbf{R}}(g)}{p_{\text{mod},\mathbf{R}}(g)} p_{\text{mod},\mathbf{R}}(g)\mathrm{d}g \\
=&1
\end{aligned}
$$

Next, we consider

$$
\left\langle \left(\frac{\eta \circ \mathcal{G}}{1-\eta \circ \mathcal{G}}\right) \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i} \right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} \tag{F.3}
$$

we similarly get

$$
\begin{aligned}
&\left\langle \left(\frac{\eta \circ \mathcal{G}}{1-\eta \circ \mathcal{G}}\right) \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i} \right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} \\
=&\int_{\chi} \frac{p_{\text{ref},\mathbf{R}}(g)}{p_{\text{mod},\mathbf{R}}(g)} \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i}(x(g,g'))J(g,g')p^{\text{pre}}_{\text{mod},r,\theta}(x(g,g'))\mathrm{d}g\mathrm{d}g' \\
=&\int_{\chi} \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i}(x(g,g')) \frac{p_{\text{ref},\mathbf{R}}(g)}{p_{\text{mod},\mathbf{R}}(g)} \frac{f_{\text{mod}}(g,g')}{p_{\text{mod},\mathbf{R}}(g)} p_{\text{mod},\mathbf{R}}(g)\mathrm{d}g\mathrm{d}g' \\
=&\int_{\chi} \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i}(x(g,g')) \frac{p_{\text{ref},\mathbf{R}}(g)}{p_{\text{mod},\mathbf{R}}(g)} f_{\text{mod}}(g'|g) p_{\text{mod},\mathbf{R}}(g)\mathrm{d}g\mathrm{d}g' \\
=&\int_{\chi} \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i}(x(g,g')) f_{\text{mod}}(g'|g) p_{\text{ref},\mathbf{R}}(g)\mathrm{d}g\mathrm{d}g' \qquad \text{(F.4)} \\
=&\int_{\mathcal{G}(\chi)} \left[\int_{\mathcal{G}'(\chi)} \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i}(x(g,g')) f_{\text{mod}}(g'|g)\mathrm{d}g'\right] p_{\text{ref},\mathbf{R}}(g)\mathrm{d}g \qquad \text{(F.5)}
\end{aligned}
$$

Eq. (F.4) shows that Eq. (F.3) is the average of the corresponding partial derivative across a mixed ensemble: the coarse-grained coordinates, here denoted g follow the distribution of the mapped

reference data, while the coordinates orthogonal to g, g', follow the distribution given by the CG force-field. Eq. (F.4) could be evaluated by using conditional simulation. However, a fruitful alternative is to consider Eq. (F.5), where the value in the square brackets is the conditional mean of the corresponding partial derivative. Using Eq. (2.21), this value can be estimated using $p^{\text{pre}}_{\text{mod},r}$ and least squares regression, as it is also the corresponding conditional mean over the pre-CG ensemble:

$$\int_{\mathcal{G}(\chi)} \left[\int_{\mathcal{G}'(\chi)} \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i}(x(g,g'))f(g'|g)\mathrm{d}g' \right] p_{\text{mod},\mathbf{R}}(g)\mathrm{d}g \tag{F.6}$$

Similarly, due to the law of conditional expectation (Eq. (2.22)), this regressed partial derivative can be used in the final term present as so

$$\left\langle \frac{\partial U^{\text{pre}}_{\text{mod}\theta}}{\partial \theta_i} \right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} = \left\langle \int_{\mathcal{G}'(\chi)} \frac{\partial U^{\text{pre}}_{\text{mod},\theta}}{\partial \theta_i}(x(g,g'))f(g'|g)\mathrm{d}g' \right\rangle_{p^{\text{pre}}_{\text{mod},r,\theta}} . \tag{F.7}$$

Collectively, this implies that the regressed partial derivative can be used as a drop in replacement in the first order REM update as described in the main text. We note that these equations are very similar to the expressions present in a expectation maximization (EM) update, as is used for Gaussian mixture modeling.

APPENDIX G

ARCG NUMERICAL SIMULATION DETAILS

This appendix contains details of the molecular potentials used, the features used as input to the variational estimator, and the noise used to smooth the optimization.

G.1 LIQUID SPLINE POTENTIALS

The *b*-splines describing the potential used to approximate water used knots that were not spaced evenly. Instead, various uniform regions of high and low knot density were used. This was due to computational constraints on the current implementation used to numerically optimize the potentials, not limitations of the methodology itself. It was found that a high knot density was needed to capture the inner well of the potential. The knots used for the water potential were 0., 0.417, 0.833, 1.25, 1.67, 2.08, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0, 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 4.0, 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 5.0, 5.4, 5.8, 6.2, 6.6, and 7.0 Angstroms. This corresponds to a higher density of knots near the inner well. In contrast, the methanol potential instead used uniform knots spaced from 0 to 10 Angstroms.

G.2 LIQUID VARIATIONAL FEATURES

This subsection describes the features used as input to the variational estimator. The single component LJ fluid and the integrated bonded particle used relatively simple feature sets, while the examples of the integrated binary LJ system, the approximated methanol system, and the approximated water system used a more complex feature set as input the variational estimator. We here

define the classes of features used, and then describe the set used for each of those examples. These features are calculated on each frame to produce the input for the variational estimator.

The first class of features is defined as the frame-wise average of a power of the distances between all the particles in the system.

$$H_{\text{moment}}(\mathbf{R}^{3N}, n) = \frac{1}{n_p} \sum_{i>j} r_{ij}^n \tag{G.1}$$

where n_p is the number of pairs in the system.

The second class characterizes the average local density of each frame. The local environment is characterized by passing the softened number of neighbors within a certain cutoff through a hyperbolic tangent function. Note that an offset and scaling factor is applied to this local density before the hyperbolic tangent is applied.

$$H_{\text{density}}(\mathbf{R}^{3N}, r_{\text{cut}}, a, b) = \frac{1}{n} \sum_i f\left(\frac{\sum_{j \neq i} -g(r_{ij} - r_{\text{cut}}) - a}{b} \right) \tag{G.2}$$

where f is the hyperbolic tangent, g is the logistic sigmoid, and n is the number of particles in the system.

The third class of features is given by calculating an RDF at each frame, i.e. given a radial bin it returns the number of particle pairs whose separating distance is in that bin.

$$H_{\text{RDF}}(\mathbf{R}^{3N}, B) := \frac{1}{n_p} \sum_{i>j} \mathbf{1}[r_{ij} \in B] \tag{G.3}$$

The single component LJ system used $H_{\text{moment}}(\cdot, -6)$ and $H_{\text{moment}}(\cdot, -12)$. This set of features is sufficient to create a complete Q as we are able to write the potential of both the reference and models systems as a function of it. The virtual bonded particle only used the distance between the two real particles as input; this is can be seen to be sufficient by considering the rotational and translational symmetry present in the system. The integrated binary LJ system and the approximated methanol system used the same set of features: this was composed of features from the 3 classes

described above. The H_{moment} features were parameterized with 2, 4, 6, and 12. The parameterization of the H_{density} features is given in table G.1. The H_{RDF} features were parameterized with 50 equally spaced bins from 2.5 Å to 10 Å. The featurization used the water example was identical except for the RDF features: in this case, they were parameterized 2 Å to 12 Å with 100 bins. The extended radial features were due to the higher resolution knot density.

$r_{\text{cut}}/\text{Å}$	a	b
4	0	2
4	1	2
7	7	2
7	9	2
10	9	2
10	11	2

Table G.1: Parameters used for the local density feature functions.

The neural network architectures used were simple feed-forward networks. Not including the input and output layers, the LJ example used to layers of 5 nodes, the virtual bonded site example used 3 layers of 10 nodes, and the binary LJ system used 4 layers of 15 nodes. The architecture did not have a noticeable effect as long as at least two layers were present.

G.3 NOISE

Noise was added to improve convergence of a variety of the numerical examples in this document (all except the case of the virtual bonded particle and dodecaalanine). This is helpful in the cases examined when the distributions being optimized are highly dissimilar. The procedure used to apply noise is summarized as follows. First, a data set composed of the combined samples from both the reference and model trajectories are whitened to have a mean of zero and a standard deviation of one in each dimension. Gaussian noise was applied of a specified variance with a mean of zero was applied to each dimension. This variance noise was geometrically decayed when the reported accuracy of the classifier (produced by optimization of the variational statement) was below a set

threshold for a set number of iterations. The decay factor was set to 0.95-0.97 for the examples presented. Additional details be found in the tests presented in the code base.

G.4 DODECAALANINE

This appendix first contains the SHAP CVs generated for the ARCG and REM 2 type dodecaalanine models. As seen in Figs G.1, G.2, and G.3, notably similar structure is seen relative to the REM no-nonbonded model presented in Fig. 3.5: the long tail structure is similarly attributable to terminal bond error. Differences are seen between the various ARCG and REM models in terms of overall number of islands; this appears to be due to capturing minima in the corresponding bond distributions, but seems to be exaggerated relative to the distributions present in Fig. 4.15. Collectively, the results suggest similar qualitative signatures of error for all three models. It is notable that the REM model seems to have higher levels of error in both Figs. G.1 and 4.16; although not discussed extensively in this document, relative entropy is known[189] to be willing to place model density where there is little reference density in order to avoid missing areas of reference density. This behavior may result in relatively large negative values of ΔW; in other words, relative entropy does not heavily penalize negative ΔW values. Nevertheless, it is difficult to be sure of the source of this discrepancy. These trends and similarities are seen for Figs. G.4 and G.5, as well as tables G.2 and G.3, which show similar results when using the Jensen-Shannon divergence and multiscale coarse-graining (MSCG). We note that due to technical constraints the MSCG model used a nonbonded cutoff of 25 angstroms and was created using bonded potentials derived from the corresponding REM model, which may strongly bias the results towards the errors seen in previous REM models. Nevertheless, the persistent similarities across the various parameterization strategies support the low importance of the model selection metric relative to the force-field basis used.

Distance	1,2	11,12	1,3	4,7	2,5	8,11
MAS	0.71	0.68	0.27	0.26	0.24	0.22

Table G.2: Top 6 mean absolute SHAP (MAS) values for classification performed between the dodecaalanine ARCG-Jensen-Shannon model and the reference data. Note that the MAS are here presented in units of $k_b T$.

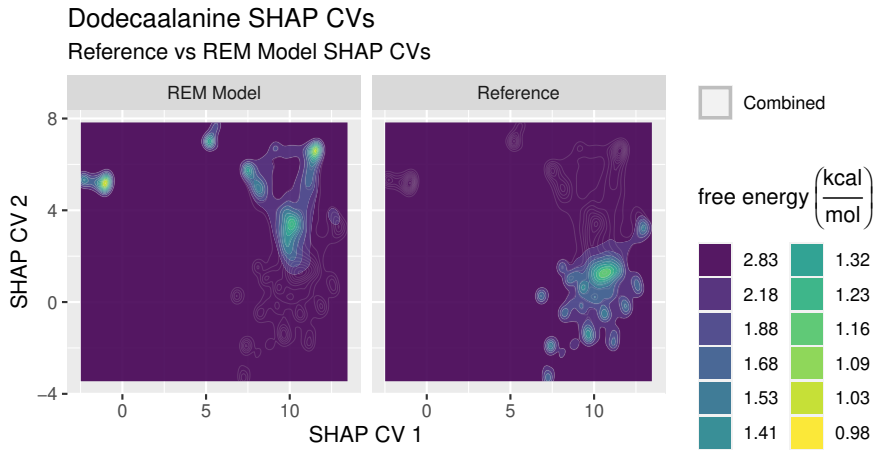

Figure G.1: Free energy surfaces produced along the SHAP CVs generated from the REM dodecaalanine model. Light grey lines characterize the combined density of the model and reference data and serve as a visual guide.

Distance	11,12	1,2	1,5	8,12	10,12	3,7
MAS	0.83	0.68	0.34	0.30	0.30	0.28

Table G.3: Top 6 mean absolute SHAP (MAS) values for classification performed between the dodecaalanine MSCG model and the reference data. Note that the this MSCG model used bond potentials derived through REM, and as such is not fully derived via MSCG; additionally, the nonbonded cutoff was set to 25 angstroms. MAS are here presented in units of $k_b T$.

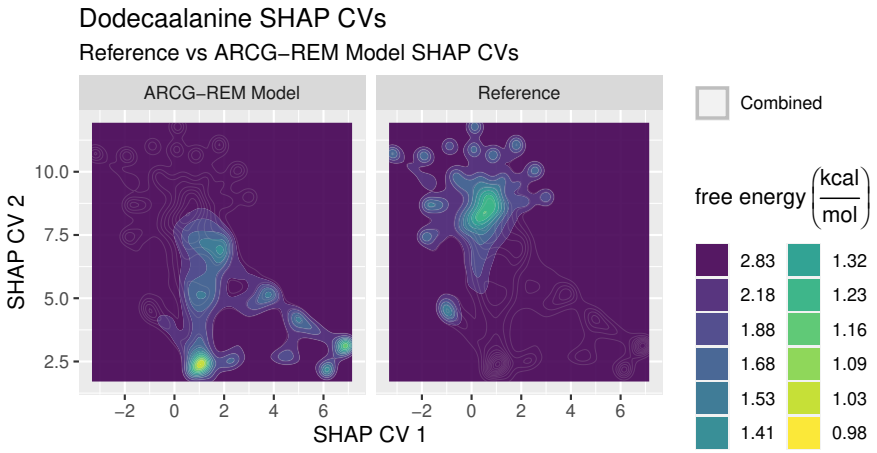

Figure G.2: Free energy surfaces produced along the SHAP CVs generated from the ARCG-REM dodecaalanine model. Light grey lines characterize the combined density of the model and reference data and serve as a visual guide.

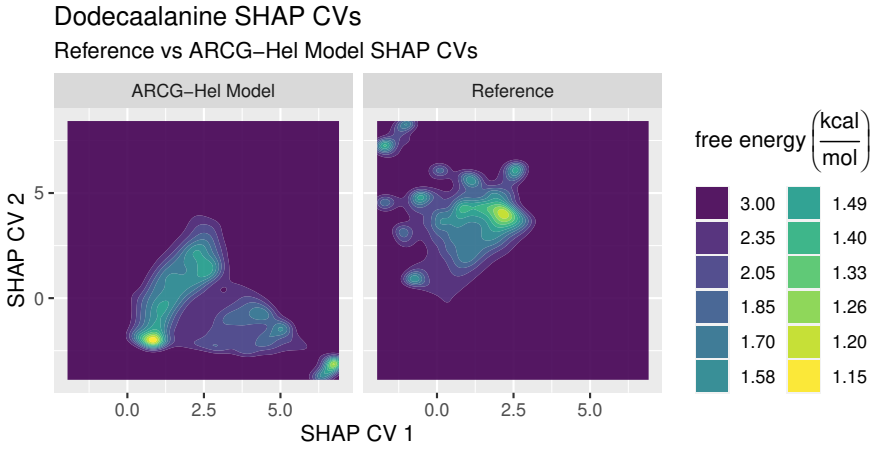

Figure G.3: Free energy surfaces produced along the SHAP CVs generated from the ARCG-Hel dodecaalanine model. Light grey lines characterize the combined density of the model and reference data and serve as a visual guide.

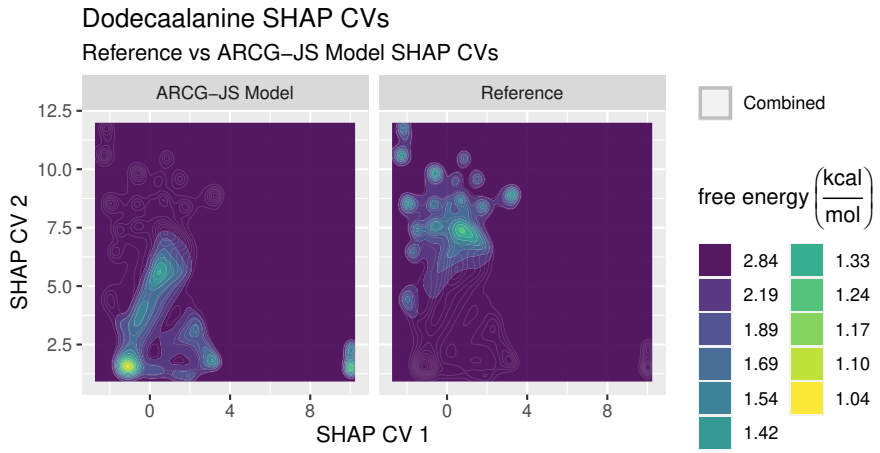

Figure G.4: Free energy surfaces produced along the SHAP CVs generated from the ARCG-Jensen-Shannon dodecaalanine model. Light grey lines characterize the combined density of the model and reference data and serve as a visual guide.

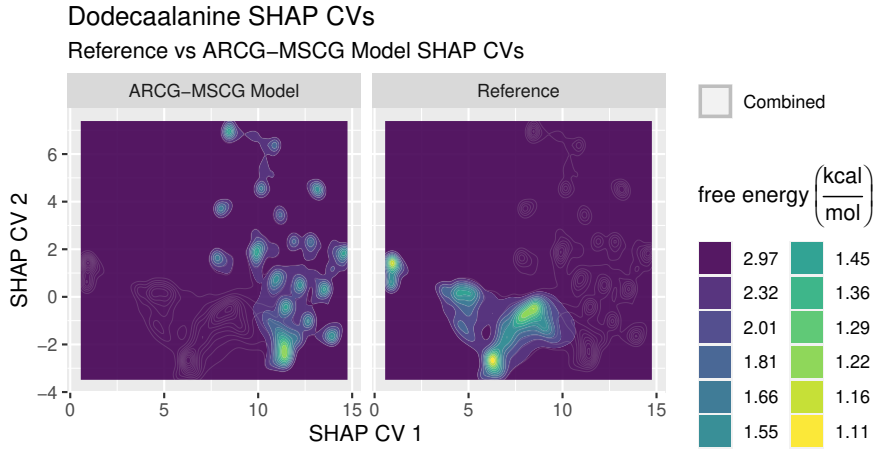

Figure G.5: Free energy surfaces produced along the SHAP CVs generated from the MSCG dodecaalanine model. See table G.3 for additional notes. Light grey lines characterize the combined density of the model and reference data and serve as a visual guide.

BIBLIOGRAPHY

1. Russakovsky, O. *et al.* Imagenet large scale visual recognition challenge. Int. J. Comput. Vision. **115,** 211–252 (2015).

2. Hu, J., Shen, L. & Sun, G. Squeeze-and-excitation networks. in Proceedings of the IEEE conference on computer vision and pattern recognition (2018), 7132–7141.

3. Karthika, S., Radhakrishnan, T. & Kalaichelvi, P. A review of classical and nonclassical nucleation theories. Cryst. Growth. Des. **16,** 6663–6681 (2016).

4. Vekilov, P. G. Nucleation. Cryst. Growth. Des. **10,** 5007–5019 (2010).

5. Rahman, A. Correlations in the motion of atoms in liquid argon. Phys. Rev. **136,** A405 (1964).

6. McCammon, J. A., Gelin, B. R. & Karplus, M. Dynamics of folded proteins. Nature **267,** 585–590 (1977).

7. Frenkel, D. & Smit, B. Understanding molecular simulation: From algorithms to applications. 2002.

8. Shaw, D. E. *et al.* Millisecond-scale molecular dynamics simulations on Anton. in Proceedings of the conference on high performance computing networking, storage and analysis (2009), 1–11.

9. Pollack, L. Fashioning NAMD, a history of risk and reward: Klaus Schulten Reminisces. Innovations in Biomolecular Modeling and Simulations **1,** 8 (2012).

10. Allen, M. P. & Tildesley, D. J. Computer simulation of liquids (Oxford university press, 2017).

11. Saunders, M. G. & Voth, G. A. Coarse-graining methods for computational biology. Ann. Rev. Biophys. **42,** 73–93 (2013).

12. Ferguson, A. L., Panagiotopoulos, A. Z., Kevrekidis, I. G. & Debenedetti, P. G. Nonlinear dimensionality reduction in molecular simulation: The diffusion map approach. Chem. Phys. Lett. **509,** 1–11 (2011).

13. Gkeka, P. *et al.* Machine Learning Force Fields and Coarse-Grained Variables in Molecular Dynamics: Application to Materials and Biological Systems. J. Chem. Theory. Comput. **16.** PMID: 32559068, 4757–4775. eprint: https://doi.org/10.1021/acs.jctc.0c00355. https://doi.org/10.1021/acs.jctc.0c00355 (2020).

14. Noé, F., Tkatchenko, A., Müller, K.-R. & Clementi, C. Machine learning for molecular simulation. Annu. Rev. Phys. Chem. **71,** 361–390 (2020).

15. Prinz, J.-H. *et al.* Markov models of molecular kinetics: Generation and validation. J. Chem. Phys. **134,** 174105 (2011).

16. Bowman, G. R., Pande, V. S. & Noé, F. An introduction to Markov state models and their application to long timescale molecular simulation (Springer Science & Business Media, 2013).

17. Mardt, A., Pasquali, L., Wu, H. & Noé, F. VAMPnets for deep learning of molecular kinetics. Nat. Commun. **9,** 1–11 (2018).

18. Noé, F. & Rosta, E. Markov models of molecular kinetics. 2019.

19. Plimpton, S. Fast parallel algorithms for short-range molecular dynamics. J. Comput. Phys. **117,** 1–19 (1995).

20. Adcock, S. A. & McCammon, J. A. Molecular dynamics: survey of methods for simulating the activity of proteins. Chem. Rev. **106,** 1589–1615 (2006).

21. Klein, M. L. & Shinoda, W. Large-scale molecular dynamics simulations of self-assembling systems. Science **321,** 798–800 (2008).

22. Shaw, D. E. *et al.* Anton, a special-purpose machine for molecular dynamics simulation. Commun. ACM **51,** 91–97 (2008).

23. Plimpton, S. J. & Thompson, A. P. Computational aspects of many-body potentials. MRS Bull. **37,** 513–521 (2012).

24. Tiwary, P. & van de Walle, A. in Multiscale Materials Modeling for Nanomechanics, 195–221 (Springer, 2016).

25. Phillips, J. C. *et al.* Scalable molecular dynamics on CPU and GPU architectures with NAMD. J. Chem. Phys. **153,** 044130 (2020).

26. Grime, J. M. *et al.* Coarse-grained simulation reveals key features of HIV-1 capsid self-assembly. Nat. Commun. **7,** 1–11 (2016).

27. Pak, A. J. *et al.* Immature HIV-1 lattice assembly dynamics are regulated by scaffolding from nucleic acid and the plasma membrane. Proc. Natl. Acad. Sci. U. S. A. **114,** E10056–E10065 (2017).

28. Pak, A. J. & Voth, G. A. Advances in coarse-grained modeling of macromolecular complexes. Curr. Opin. Struc. Biol. **52,** 119–126 (2018).

29. Tóth, G. Interactions from diffraction data: historical and comprehensive overview of simulation assisted methods. J. Phys-Condens. Mat. **19,** 335220 (2007).

30. Voth, G. A. Coarse-graining of condensed phase and biomolecular systems (CRC press, 2008).

31. Peter, C. & Kremer, K. Multiscale simulation of soft matter systems–from the atomistic to the coarse-grained level and back. Soft Matter **5,** 4357–4366 (2009).

32. Kamerlin, S. C. L. & Warshel, A. Multiscale modeling of biological functions. Phys. Chem. Chem. Phys. **13,** 10401–10411 (2011).

33. Noid, W. Perspective: Coarse-grained models for biomolecular systems. J. Chem. Phys. **139,** 09B201_1 (2013).

34. Marrink, S. J. & Tieleman, D. P. Perspective on the Martini model. Chem. Soc. Rev. **42,** 6801–6822 (2013).

35. Potestio, R., Peter, C. & Kremer, K. Computer simulations of soft matter: Linking the scales. Entropy **16,** 4199–4245 (2014).

36. Noid, W. G. in Biomolecular Simulations, 487–531 (Springer, 2013).

37. Brini, E. *et al.* Systematic coarse-graining methods for soft matter simulations–a review. Soft Matter **9,** 2108–2119 (2013).

38. Stillinger, F. H. Effective pair interactions in liquids. Water. J. Phys. Chem **74,** 3677–3687 (1970).

39. Schommers, W. A pair potential for liquid rubidium from the pair correlation function. Phys. Lett. A **43,** 157–158 (1973).

40. Lyubartsev, A. P. & Laaksonen, A. Calculation of effective interaction potentials from radial distribution functions: A reverse Monte Carlo approach. Phys. Rev. E **52,** 3730 (1995).

41. Müller-Plathe, F. Coarse-graining in polymer simulation: from the atomistic to the mesoscopic scale and back. ChemPhysChem **3,** 754–769 (2002).

42. Reith, D., Pütz, M. & Müller-Plathe, F. Deriving effective mesoscale potentials from atomistic simulations. J. Comput. Chem. **24,** 1624–1636 (2003).

43. Izvekov, S. & Voth, G. A. A multiscale coarse-graining method for biomolecular systems. J. Phys. Chem. B **109,** 2469–2473 (2005).

44. Noid, W., Chu, J.-W., Ayton, G. S. & Voth, G. A. Multiscale coarse-graining and structural correlations: Connections to liquid-state theory. The Journal of Physical Chemistry B **111,** 4116–4127 (2007).

45. Noid, W. *et al.* The multiscale coarse-graining method. I. A rigorous bridge between atomistic and coarse-grained models. J. Chem. Phys. **128,** 244114 (2008).

46. Noid, W. *et al.* The multiscale coarse-graining method. II. Numerical implementation for coarse-grained molecular models. J. Chem. Phys. **128,** 244115 (2008).

47. Shell, M. S. The relative entropy is fundamental to multiscale and inverse thermodynamic problems. J. Chem. Phys. **129,** 144108 (2008).

48. Mullinax, J. & Noid, W. Generalized Yvon-Born-Green theory for molecular systems. Phys. Rev. Lett. **103,** 198104 (2009).

49. Savelyev, A. & Papoian, G. A. Molecular renormalization group coarse-graining of polymer chains: application to double-stranded DNA. Biophys. J. **96,** 4044–4052 (2009).

50. Karimi-Varzaneh, H. A., Qian, H.-J., Chen, X., Carbone, P. & Müller-Plathe, F. IBIsCO: A molecular dynamics simulation package for coarse-grained simulation. J. Comput. Chem. **32,** 1475–1487 (2011).

51. Carmichael, S. P. & Shell, M. S. A new multiscale algorithm and its application to coarse-grained peptide models for self-assembly. J. Phys. Chem. B **116,** 8383–8393 (2012).

52. Dama, J. F. *et al.* The theory of ultra-coarse-graining. 1. General principles. J. Chem. Theory Comput. **9,** 2466–2480 (2013).

53. Rudzinski, J. F. & Noid, W. G. Bottom-up coarse-graining of peptide ensembles and helix–coil transitions. J. Chem. Theory Comput. **11,** 1278–1291 (2015).

54. Lyubartsev, A. P., Naômé, A., Vercauteren, D. P. & Laaksonen, A. Systematic hierarchical coarse-graining with the inverse Monte Carlo method. J. Chem. Phys. **143,** 243120 (2015).

55. Vlcek, L. & Chialvo, A. A. Rigorous force field optimization principles based on statistical distance minimization. J. Chem. Phys. **143,** 144110 (2015).

56. De Oliveira, T. E., Netz, P. A., Kremer, K., Junghans, C. & Mukherji, D. C–IBI: Targeting cumulative coordination within an iterative protocol to derive coarse-grained models of (multi-component) complex fluids. J. Chem. Phys. **144,** 174106 (2016).

57. Sanyal, T. & Shell, M. S. Coarse-grained models using local-density potentials optimized with the relative entropy: Application to implicit solvation. J. Chem. Phys. **145,** 034109 (2016).

58. Dunn, N. J. & Noid, W. Bottom-up coarse-grained models with predictive accuracy and transferability for both structural and thermodynamic properties of heptane-toluene mixtures. J. Chem. Phys. **144,** 204124 (2016).

59. Lemke, T. & Peter, C. Neural network based prediction of conformational free energies-a new route towards coarse-grained simulation models. J. Chem. Theory Comput. (2017).

60. John, S. T. & Csanyi, G. Many-Body Coarse-Grained Interactions Using Gaussian Approximation Potentials. J. Phys. Chem. B **121,** 10934–10949 (2017).

61. Wagner, J. W., Dannenhoffer-Lafage, T., Jin, J. & Voth, G. A. Extending the range and physical accuracy of coarse-grained models: Order parameter dependent interactions. J. Chem. Phys. **147,** 044113 (2017).

62. Tsourtis, A., Harmandaris, V. & Tsagkarogiannis, D. Parameterization of coarse-grained molecular interactions through potential of mean force calculations and cluster expansion techniques. Entropy **19,** 395 (2017).

63. Schöberl, M., Zabaras, N. & Koutsourelakis, P.-S. Predictive coarse-graining. J. Comput. Phys. **333,** 49–77 (2017).

64. Zhang, L., Han, J., Wang, H., Car, R. & Weinan, E. DeePCG: constructing coarse-grained models via deep neural networks. J. Chem. Phys. **149,** 09B201_1 (2013).

65. Wang, J. *et al.* Machine learning of coarse-grained molecular dynamics force fields. ACS Cent. Sci. **5,** 755–767 (2019).

66. Jin, J., Han, Y. & Voth, G. A. Coarse-graining involving virtual sites: Centers of symmetry coarse-graining. J. Chem. Phys. **150,** 154103 (2019).

67. Durumeric, A. E. & Voth, G. A. Adversarial-residual-coarse-graining: Applying machine learning theory to systematic molecular coarse-graining. J. Chem. Phys. **151,** 124110 (2019).

68. Husic, B. E. *et al.* Coarse Graining Molecular Dynamics with Graph Neural Networks. arXiv preprint arXiv:2007.11412 (2020).

69. Rudzinski, J. F. & Noid, W. Coarse-graining entropy, forces, and structures. J. Chem. Phys. **135,** 214101 (2011).

70. Chaimovich, A. & Shell, M. S. Anomalous waterlike behavior in spherically-symmetric water models optimized with the relative entropy. Phys. Chem. Chem. Phys. **11,** 1901–1915 (2009).

71. Jarin, Z. *et al.* Unusual Organization of I-BAR Proteins on Tubular and Vesicular Membranes. Biophys. J. **117,** 553–562 (2019).

72. Wagner, J. W., Dama, J. F., Durumeric, A. E. & Voth, G. A. On the representability problem and the physical meaning of coarse-grained models. J. Chem. Phys. **145,** 044108 (2016).

73. Jin, J., Pak, A. J. & Voth, G. A. Understanding missing entropy in coarse-grained systems: Addressing issues of representability and transferability. J. Phys. Chem. Lett. **10,** 4549–4557 (2019).

74. Mitchell, T. M. Machine learning and data mining. Commun. ACM **42,** 30–36 (1999).

75. Friedman, J., Hastie, T. & Tibshirani, R. The elements of statistical learning. **10** (Springer series in statistics New York, 2001).

76. Mohri, M., Rostamizadeh, A. & Talwalkar, A. Foundations of machine learning (MIT press, 2018).

77. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat. Mach. Intell. **1,** 206–215 (2019).

78. Silver, D. *et al.* Mastering the game of Go with deep neural networks and tree search. Nature **529,** 484–489 (2016).

79. Ceriotti, M. Unsupervised machine learning in atomistic simulations, between predictions and understanding. J. Chem. Phys. **150,** 150901 (2019).

80. Song, L., Reid, M. D., Williamson, R. C. & Smola, A. J. Discriminative Estimation of f-Divergence.

81. Reid, M. D. & Williamson, R. C. Information, divergence and risk for binary experiments. J. Mach. Learn. Res. **12,** 731–817 (2011).

82. Goodfellow, I. *et al.* Generative adversarial nets. in Adv. Neur. In. (2014), 2672–2680.

83. Mohamed, S. & Lakshminarayanan, B. Learning in implicit generative models. arXiv preprint arXiv:1610.03483 (2016).

84. Karpathy, A. Generative Models. 2020. https://openai.com/blog/generative-models/.

85. Alqahtani, H., Kavakli-Thorne, M. & Kumar, G. Applications of generative adversarial networks (gans): An updated review. Arch. Comput. Method. E., 1–28 (2019).

86. Curtó, J. D., Zarza, H. & Kim, T. High-resolution deep convolutional generative adversarial networks. arXiv preprint arXiv:1711.06491 (2017).

87. Wang, X. *et al.* Esrgan: Enhanced super-resolution generative adversarial networks. in Proceedings of the European Conference on Computer Vision (ECCV) (2018).

88. Dai, B., Fidler, S., Urtasun, R. & Lin, D. Towards diverse and natural image descriptions via a conditional gan. in Proceedings of the IEEE International Conference on Computer Vision (2017), 2970–2979.

89. Noé, F., Olsson, S., Köhler, J. & Wu, H. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. Science **365,** eaaw1147 (2019).

90. Li, W., Burkhart, C., Polińska, P., Harmandaris, V. & Doxastakis, M. Backmapping coarse-grained macromolecules: An efficient and versatile machine learning approach. J. Chem. Phys. **153,** 041101 (2020).

91. Stieffenhofer, M., Wand, M. & Bereau, T. Adversarial Reverse Mapping of Equilibrated Condensed-Phase Molecular Structures. arXiv preprint arXiv:2003.07753 (2020).

92. LeCun, Y., Cortes, C. & Burges, C. MNIST handwritten digit database. ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist **2** (2010).

93. Steppan, J. MNIST database. 2020. https://en.wikipedia.org/wiki/MNIST_database.

94. Buja, A., Stuetzle, W. & Shen, Y. Loss functions for binary class probability estimation and classification: Structure and applications. Working draft, November **3** (2005).

95. Gneiting, T. & Raftery, A. E. Strictly proper scoring rules, prediction, and estimation. J. Am. Stat. Assoc. **102,** 359–378 (2007).

96. Christensen, R. Plane answers to complex questions. **1** (Springer, 2002).

97. Dannenhoffer-Lafage, T., Wagner, J. W., Durumeric, A. E. & Voth, G. A. Compatible observable decompositions for coarse-grained representations of real molecular systems. J. Chem. Phys. **151,** 134115 (2019).

98. Pak, A. J., Dannenhoffer-Lafage, T., Madsen, J. J. & Voth, G. A. Systematic coarse-grained lipid force fields with semiexplicit solvation via virtual sites. J. Chem. Theory. Comput. **15,** 2087–2100 (2019).

99. Karplus, M. & McCammon, J. A. Molecular dynamics simulations of biomolecules. Nat Struct. Mol. Bio. **9,** 646 (2002).

100. Shaw, D. E. *et al.* Atomic-level characterization of the structural dynamics of proteins. Science **330,** 341–346 (2010).

101. Buch, I., Harvey, M. J., Giorgino, T., Anderson, D. P. & De Fabritiis, G. High-throughput all-atom molecular dynamics simulations using distributed computing. J. Chem. Inf. Model. **50,** 397–403 (2010).

102. Lindorff-Larsen, K., Piana, S., Dror, R. O. & Shaw, D. E. How fast-folding proteins fold. Science **334,** 517–520 (2011).

103. Karplus, M. & Lavery, R. Significance of molecular dynamics simulations for life sciences. Isr. J. Chem. **54,** 1042–1051 (2014).

104. Rudzinski, J. F. Recent progress towards chemically-specific coarse-grained simulation models with consistent dynamical properties. Computation **7,** 42 (2019).

105. Humphrey, W., Dalke, A., Schulten, K., *et al.* VMD: visual molecular dynamics. J. Mol. Graphics **14,** 33–38 (1996).

106. Bartók, A. P., Kermode, J., Bernstein, N. & Csányi, G. Machine learning a general-purpose interatomic potential for silicon. Phys. Rev. X **8,** 041048 (2018).

107. Stoltz, G., Rousset, M., *et al.* Free energy computations: A mathematical perspective (World Scientific, 2010).

108. Molnar, C. Interpretable Machine Learning (Lulu.com, 2020).

109. Arrieta, A. B. *et al.* Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. Inform. Fusion **58,** 82–115 (2020).

110. Arya, V. *et al.* One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques. arXiv preprint arXiv:1909.03012 (2019).

111. Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R. & Yu, B. Definitions, methods, and applications in interpretable machine learning. Proc. Natl. Acad. Sci. U. S. A. **116,** 22071–22080 (2019).

112. Molnar, C., Casalicchio, G. & Bischl, B. Interpretable Machine Learning–A Brief History, State-of-the-Art and Challenges. arXiv preprint arXiv:2010.09337 (2020).

113. Holzinger, A., Kieseberg, P., Weippl, E. & Tjoa, A. M. Current Advances, Trends and Challenges of Machine Learning and Knowledge Extraction: From Machine Learning to Explainable AI. in Machine Learning and Knowledge Extraction (eds Holzinger, A., Kieseberg, P., Tjoa, A. M. & Weippl, E.) (Springer International Publishing, Cham, 2018), 1–8. ISBN: 978-3-319-99740-7.

114. Kodratoff, Y. The comprehensibility manifesto. 1994. https://www.kdnuggets.com/news/94/n9.txt.

115. Rüping, S. *et al.* Learning interpretable models (2006).

116. Mittelstadt, B., Russell, C. & Wachter, S. Explaining explanations in AI. in Proceedings of the conference on fairness, accountability, and transparency (2019), 279–288.

117. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. in Adv. Neur. In. (2017), 4765–4774.

118. Lundberg, S. M. *et al.* From local explanations to global understanding with explainable AI for trees. Nat. Mach. Intell. **2,** 2522–5839 (2020).

119. Lundberg, S. M. *et al.* Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. Nat. Biomed. Eng. **2,** 749 (2018).

120. Shapley, L. S. A value for n-person games. Contributions to the Theory of Games **2,** 307–317 (1953).

121. Young, H. P. Monotonic solutions of cooperative games. Int. J. Game. Theory **14,** 65–72 (1985).

122. Sundararajan, M. & Najmi, A. The many Shapley values for model explanation. arXiv preprint arXiv:1908.08474 (2019).

123. Kumar, I. E., Venkatasubramanian, S., Scheidegger, C. & Friedler, S. Problems with Shapley-value-based explanations as feature importance measures. arXiv preprint arXiv:2002.11097 (2020).

124. Vinayak, R. K. & Gilad-Bachrach, R. Dart: Dropouts meet multiple additive regression trees. in Artificial Intelligence and Statistics (2015), 489–497.

125. Ke, G. *et al.* Lightgbm: A highly efficient gradient boosting decision tree. in Adv. Neur. In. (2017), 3146–3154.

126. McInnes, L., Healy, J. & Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018).

127. Maaten, L. v. d. & Hinton, G. Visualizing data using t-SNE. J. Mach. Learn. Res. **9,** 2579–2605 (2008).

128. Case, D. *et al.* AMBER 2018. University of California, San Francisco (2018).

129. Huang, J. *et al.* CHARMM36m: an improved force field for folded and intrinsically disordered proteins. Nat. Methods **14,** 71–73 (2017).

130. Schneider, T. & Stoll, E. Molecular-dynamics study of a three-dimensional one-component model for distortive phase transitions. Phys. Rev. B **17,** 1302 (1978).

131. Shinoda, W., Shiga, M. & Mikami, M. Rapid estimation of elastic constants by molecular dynamics simulation under constant stress. Phys. Rev. B **69,** 134103 (2004).

132. Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. arXiv e-prints **abs/1605.02688.** http://arxiv.org/abs/1605.02688 (May 2016).

133. Harris, C. R. *et al.* Array programming with NumPy. Nature **585,** 357–362. https://doi.org/10.1038/s41586-020-2649-2 (Sept. 2020).

134. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res. **12,** 2825–2830 (2011).

135. Lundberg, S. M. *et al.* From local explanations to global understanding with explainable AI for trees. Nat. Mach. Intell. **2,** 2522–5839 (2020).

136. Pandas development team, T. pandas-dev/pandas: Pandas 1.1.3. Version v1.1.3 (Zenodo, Oct. 2020). https://doi.org/10.5281/zenodo.4067057.

137. Wickham, H. ggplot2: Elegant Graphics for Data Analysis. ISBN: 978-3-319-24277-4. https://ggplot2.tidyverse.org (Springer-Verlag New York, 2016).

138. Dowle, M. & Srinivasan, A. data.table: Extension of 'data.frame'. R package version 1.12.8 (2019). https://CRAN.R-project.org/package=data.table.

139. Borchers, H. W. pracma: Practical Numerical Math Functions. R package version 2.2.9 (2019). https://CRAN.R-project.org/package=pracma.

140. Hinton, G., Srivastava, N. & Swersky, K. Neural networks for machine learning. Coursera, video lectures **264** (2012).

141. Piana, S. & Laio, A. A bias-exchange approach to protein folding. J. Phys. Chem. B **111,** 4553–4559 (2007).

142. Prakash, A., Baer, M. D., Mundy, C. J. & Pfaendtner, J. Peptoid backbone flexibilility dictates its interaction with water and surfaces: a molecular dynamics investigation. Biomacromolecules **19,** 1006–1015 (2018).

143. Abraham, M. J. *et al.* GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. SoftwareX **1,** 19–25 (2015).

144. Hocky, G. M. *et al.* Cations stiffen actin filaments by adhering a key structural element to adjacent subunits. J. Phys. Chem. B **120,** 4558–4567 (2016).

145. Mackerell Jr, A. D., Feig, M. & Brooks III, C. L. Extending the treatment of backbone energetics in protein force fields: Limitations of gas-phase quantum mechanics in reproducing protein conformational distributions in molecular dynamics simulations. J. Comput. Chem. **25,** 1400–1415 (2004).

146. Bussi, G., Donadio, D. & Parrinello, M. Canonical sampling through velocity rescaling. J. Chem. Phys. **126,** 014101 (2007).

147. Parrinello, M. & Rahman, A. Polymorphic transitions in single crystals: A new molecular dynamics method. J. Appl. Phys. **52,** 7182–7190 (1981).

148. Saunders, M. G. & Voth, G. A. Comparison between actin filament models: coarse-graining reveals essential differences. Structure **20,** 641–653 (2012).

149. Lyman, E., Pfaendtner, J. & Voth, G. A. Systematic multiscale parameterization of heterogeneous elastic network models of proteins. Biophys. J. **95,** 4183–4192 (2008).

150. Zsolnay, V., Katkar, H. H., Chou, S. Z., Pollard, T. D. & Voth, G. A. Structural basis for polarized elongation of actin filaments. bioRxiv (2020).

151. Sultan, M. M. & Pande, V. S. Automated design of collective variables using supervised machine learning. J. Chem. Phys. **149,** 094106 (2018).

152. Pfaendtner, J., Branduardi, D., Parrinello, M., Pollard, T. D. & Voth, G. A. Nucleotide-dependent conformational states of actin. Proc. Natl. Acad. Sci. U. S. A. **106,** 12723–12728 (2009).

153. Katkar, H. H. *et al.* Insights into the cooperative nature of ATP hydrolysis in actin filaments. Biophy. J. **115,** 1589–1602 (2018).

154. Galkin, V. E., Orlova, A., Schröder, G. F. & Egelman, E. H. Structural polymorphism in F-actin. Nat. Struct. Mol. Biol. **17,** 1318 (2010).

155. Galkin, V. E., Orlova, A., Vos, M. R., Schröder, G. F. & Egelman, E. H. Near-atomic resolution for one state of F-actin. Structure **23,** 173–182 (2015).

156. Huehn, A. R. *et al.* Structures of cofilin-induced structural changes reveal local and asymmetric perturbations of actin filaments. Proc. Natl. Acad. Sci. U. S. A. **117,** 1478–1484 (2020).

157. Bennett, C. H. Efficient estimation of free energy differences from Monte Carlo data. J. Comput. Phys. **22,** 245–268 (1976).

158. Goscinski, A., Fraux, G. & Ceriotti, M. The role of feature space in atomistic learning. arXiv preprint arXiv:2009.02741 (2020).

159. Anderson, B., Hy, T.-S. & Kondor, R. Cormorant: Covariant Molecular Neural Networks. arXiv preprint arXiv:1906.04015 (2019).

160. Schütt, K. T., Sauceda, H. E., Kindermans, P.-J., Tkatchenko, A. & Müller, K.-R. SchNet–A deep learning architecture for molecules and materials. J. Chem. Phys. **148,** 241722 (2018).

161. Hindupur, A. The GAN zoo. Retrieved from Deep Hunt website: https://deephunt. in/the-gan-zoo-79597dc8c347 (2017).

162. Jin, J., Han, Y. & Voth, G. A. Coarse-graining involving virtual sites: Centers of symmetry coarse-graining. J. Chem. Phys. **150,** 154103 (2019).

163. Baaden, M. & Marrink, S. J. Coarse-grain modelling of protein–protein interactions. Curr. Opin. Struc. Bio. **23,** 878–886 (2013).

164. Fan, J., Saunders, M. G. & Voth, G. A. Coarse-graining provides insights on the essential nature of heterogeneity in actin filaments. Biophys. J. **103,** 1334–1342 (2012).

165. Srivastava, A. & Voth, G. A. Hybrid approach for highly coarse-grained lipid bilayer models. J. Chem. Theory Comput. **9,** 750–765 (2012).

166. Cao, Z., Dama, J. F., Lu, L. & Voth, G. A. Solvent free ionic solution models from multiscale coarse-graining. J. Chem. Theory Comput. **9,** 172–178 (2012).

167. Abraham, M. J. *et al.* GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. SoftwareX **1,** 19–25 (2015).

168. Brooks, B. R. *et al.* CHARMM: a program for macromolecular energy, minimization, and dynamics calculations. J. Comput. Chem. **4,** 187–217 (1983).

169. Anderson, J. A., Lorenz, C. D. & Travesset, A. General purpose molecular dynamics simulations fully implemented on graphics processing units. J. Comput. Phys. **227,** 5342–5359 (2008).

170. Case, D. A. *et al.* The Amber biomolecular simulation programs. J. Comput. Chem. **26,** 1668–1688 (2005).

171. Phillips, J. C. *et al.* Scalable molecular dynamics with NAMD. J. Comput. Chem. **26,** 1781–1802 (2005).

172. Bowers, K. J. *et al.* Scalable algorithms for molecular dynamics simulations on commodity clusters. in Proceedings of the 2006 ACM/IEEE conference on Supercomputing (2006), 84.

173. Markidis, S. & Laure, E. Solving Software Challenges for Exascale (Springer, 2015).

174. Chaimovich, A. & Shell, M. S. Coarse-graining errors and numerical optimization using a relative entropy framework. J. Chem. Phys. **134,** 094112 (2011).

175. Goodfellow, I. NIPS 2016 tutorial: Generative adversarial networks. arXiv preprint arXiv:1701.00160 (2016).

176. Doersch, C. Tutorial on variational autoencoders. arXiv preprint arXiv:1606.05908 (2016).

177. Alain, G. *et al.* GSNs: generative stochastic networks. Information and Inference: A Journal of the IMA **5,** 210–249 (2016).

178. Radford, A., Metz, L. & Chintala, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. ArXiv e-prints. arXiv: 1511.06434 [cs.LG] (Nov. 2015).

179. Creswell, A. & Bharath, A. A. Adversarial Training For Sketch Retrieval. ArXiv e-prints. arXiv: 1607.02748 [cs.CV] (July 2016).

180. Sanchez-Lengeling, B., Outeiral, C., Guimaraes, G. L. & Aspuru-Guzik, A. Optimizing distributions over molecular space. An Objective-Reinforced Generative Adversarial Network for Inverse-design Chemistry (ORGANIC) (2017).

181. Ertl, P., Lewis, R., Martin, E. & Polyakov, V. In silico generation of novel, drug-like chemical matter using the LSTM neural network. ArXiv e-prints. arXiv: 1712.07449 [cs.LG] (Dec. 2017).

182. Shafaei, A., Little, J. J. & Schmidt, M. Play and learn: Using video games to train computer vision models. arXiv preprint arXiv:1608.01745 (2016).

183. Wood, E., Baltrušaitis, T., Morency, L.-P., Robinson, P. & Bulling, A. Learning an appearance-based gaze estimator from one million synthesised images. in Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications (2016), 131–138.

184. Salakhutdinov, R. & Larochelle, H. Efficient learning of deep Boltzmann machines. in Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (2010), 693–700.

185. Kingma, D. P. & Welling, M. Auto-Encoding Variational Bayes. ArXiv e-prints. arXiv: 1312.6114 [stat.ML] (Dec. 2013).

186. Nowozin, S., Cseke, B. & Tomioka, R. f-gan: Training generative neural samplers using variational divergence minimization. in Adv. Neur. In. (2016), 271–279.

187. Nguyen, X., Wainwright, M. J. & Jordan, M. I. Estimating divergence functionals and the likelihood ratio by convex risk minimization. IEEE T. Inform. Theory. **56,** 5847–5861 (2010).

188. Bouchacourt, D., Mudigonda, P. K. & Nowozin, S. DISCO Nets: DISsimilarity COefficients Networks. Advances in Neural Information Processing Systems **29,** 352–360 (2016).

189. Bottou, L., Arjovsky, M., Lopez-Paz, D. & Oquab, M. Geometrical Insights for Implicit Generative Modeling. arXiv preprint arXiv:1712.07822 (2017).

190. Arjovsky, M., Chintala, S. & Bottou, L. Wasserstein gan. arXiv preprint arXiv:1701.07875 (2017).

191. James, G., Witten, D., Hastie, T. & Tibshirani, R. An introduction to statistical learning (Springer, 2013).

192. L'Ecuyer, P. Note: On the interchange of derivative and expectation for likelihood ratio derivative estimators. Manage. Sci. **41,** 738–747 (1995).

193. Kleijnen, J. P. & Rubinstein, R. Y. Optimization and sensitivity analysis of computer simulation models by the score function method. Eur. J. Oper. Res. **88,** 413–427 (1996).

194. Milgrom, P. & Segal, I. Envelope theorems for arbitrary choice sets. Econometrica **70,** 583–601 (2002).

195. Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B. & Smola, A. A kernel two-sample test. J. Mach. Learn. Res. **13,** 723–773 (2012).

196. Dziugaite, G. K., Roy, D. M. & Ghahramani, Z. Training generative neural networks via maximum mean discrepancy optimization. arXiv preprint arXiv:1505.03906 (2015).

197. Csiszár, I. Information-type measures of difference of probability distributions and indirect observation. Stud. Sci. Math. Hung. **2,** 229–318 (1967).

198. Ali, S. M. & Silvey, S. D. A general class of coefficients of divergence of one distribution from another. J. Roy. Stat. Soc. B. Met. **28,** 131–142 (1966).

199. Lennard-Jones, J. E. Cohesion. P. Phys. Soc. **43,** 461 (1931).

200. Bilionis, I. & Zabaras, N. A stochastic optimization approach to coarse-graining using a relative-entropy framework. J. Chem. Phys. **138,** 044313 (2013).

201. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B. & Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. in Adv. Neur. In. (2017), 6626–6637.

202. Abrams, C. F. & Vanden-Eijnden, E. On-the-fly free energy parameterization via temperature accelerated molecular dynamics. Chem. Phys. Lett. **547,** 114–119 (2012).

203. Transtrum, M. K. *et al.* Perspective: Sloppiness and emergent theories in physics, biology, and beyond. J. Chem. Phys. **143,** 07B201_1 (2015).

204. Zwanzig, R. Nonlinear generalized Langevin equations. J. Stat. Phys. **9,** 215–220 (1973).

205. Zwanzig, R. Nonequilibrium statistical mechanics (Oxford University Press, 2001).

206. Davtyan, A., Dama, J. F., Voth, G. A. & Andersen, H. C. Dynamic force matching: A method for constructing dynamical coarse-grained models with realistic time dependence. J. Chem. Phys **142,** 154104 (2015).

207. Jorgensen, W. L. & Tirado-Rives, J. Potential energy functions for atomic-level simulations of water and organic and biomolecular systems. J. Phys. Chem.Proc. Natl. Acad. Sci. U. S. A. **102,** 6665–6670 (2005).

208. Dodda, L. S., Vilseck, J. Z., Tirado-Rives, J. & Jorgensen, W. L. 1.14* CM1A-LBCC: localized bond-charge corrected CM1A charges for condensed-phase simulations. J. Phys. Chem. B **121,** 3864–3870 (2017).

209. Dodda, L. S., Cabeza de Vaca, I., Tirado-Rives, J. & Jorgensen, W. L. LigParGen web server: an automatic OPLS-AA parameter generator for organic ligands. Nucleic Acids Res. **45,** W331–W336 (2017).

210. Mroueh, Y., Li, C.-L., Sercu, T., Raj, A. & Cheng, Y. Sobolev gan. arXiv preprint arXiv:1711.04894 (2017).

211. Pérez-Hernández, G., Paul, F., Giorgino, T., De Fabritiis, G. & Noé, F. Identification of slow molecular order parameters for Markov model construction. J. Chem. Phys. **139,** 07B604_1 (2013).

212. Noé, F. & Clementi, C. Kinetic distance and kinetic maps from molecular dynamics simulation. J. Chem. Theory Comput. **11,** 5002–5011 (2015).

213. Noé, F., Banisch, R. & Clementi, C. Commute maps: separating slowly mixing molecular configurations for kinetic modeling. J. Chem. Theory Comput. **12,** 5620–5630 (2016).

214. Liu, S. & Chaudhuri, K. The inductive bias of restricted f-gans. arXiv preprint arXiv:1809.04542 (2018).

215. Grime, J. & Madsen, J. J. Efficient simulation of tunable lipid assemblies across scales and resolutions. arXiv preprint arXiv:1910.05362 (2019).

216. Song, Y. & Kingma, D. P. How to Train Your Energy-Based Models. arXiv preprint arXiv:2101.03288 (2021).

217. Han, Y., Dama, J. F. & Voth, G. A. Mesoscopic coarse-grained representations of fluids rigorously derived from atomistic models. J. Chem. Phys. **149,** 044104 (2018).